Schema Free Querying of Semantic Data

by

Lushan Han

Thesis submitted to the Faculty of the Graduate School
of the University of Maryland in partial fulfillment
of the requirements for the degree of
PhD Computer Science
2014

To my wife Jie and my parents

ABSTRACT

Title of Thesis: Schema Free Querying of Semantic Data

Lushan Han, PhD Computer Science, 2014

Thesis directed by: Dr. Tim Finin, Professor
Department of Computer Science and
Electrical Engineering

Developing interfaces to enable casual, non-expert users to query complex structured data has been the subject of much research over the past forty years. We refer to them as schema-free query interfaces, since they allow users to freely query data without understanding its schema, knowing how to refer to objects, or mastering the appropriate formal query language. Schema-free query interfaces address fundamental problems in natural language processing, databases and AI to connect users' conceptual models and machine representations.

However, schema-free query interface systems are faced with three hard problems. First, we still lack a practical interface. Natural Language Interfaces (NLIs) are easy for users but hard for machines. Current NLP techniques are still unreliable in extracting the relational structure from natural language questions. Keyword query interfaces, on the other hand, have limited expressiveness and inherit ambiguity from the natural language terms used as keywords. Second, people express or model the same meaning in many different ways, which can result in the vocabulary and structure mismatches between users' queries and the machines' representation. We still rely on ad hoc and labor-intensive approaches to deal with this "semantic heterogeneity problem". Third, the Web has seen increasing amounts of open domain semantic data with heterogeneous or unknown schemas, which challenges traditional NLI systems that require a well-defined schema. Some modern systems gave up the approach of translating the user query into a formal query at the schema level and chose to directly search into the entity network (ABox) for the matchings of the user query. This approach, however, is computationally expensive and has an ad hoc nature.

In this thesis, we develop a novel approach to address the three hard problems. We introduce a new schema-free query interface, SFQ interface, in which users explicitly specify the relational structure of the query as a graphical "skeleton" and annotate it with freely chosen words, phrases and entity names. This circumvents the unreliable step of extracting complete relations from natural language queries.

We describe a framework for interpreting these SFQ queries over open domain semantic data that automatically translates them to formal queries. First, we learn a schema statistically from the entity network and represent it as a graph, which we call the schema network. Our mapping algorithms run on the schema network rather than the entity network, enhancing scalability. We define the probability of "observing" a path on the schema network. Following it, we create two statistical association models that will be used to carry out disambiguation. Novel mapping algorithms are developed that exploit semantic similarity measures and association measures to address the structure and vocabulary mismatch problems. Our approach is fully computational and requires no special lexicons, mapping rules, domain-specific syntactic or semantic grammars, thesauri or hard-coded semantics.

We evaluate our approach on two large datasets, DBLP+ and DBpedia. We developed DBLP+ by augmenting the DBLP dataset with additional data from CiteSeerX and ArnetMiner. We created 220 SFQ queries on the DBLP+ dataset. For DBpedia, we had three human subjects (who were unfamiliar with DBpedia) translate 33 natural language questions from the 2011 QALD workshop into SFQ queries. We carried out cross-validation on the 220 DBLP+ queries and cross-domain validation on the 99 DBpedia queries in which the parameters tuned for the DBLP+ queries are applied to the DBpedia queries. The evaluation results on the two datasets show that our system has very good efficacy and efficiency.

ACKNOWLEDGMENTS

I would like to give my most thanks to my advisor, Prof. Tim Finin. Without his continuous support, I don't think I could finish this thesis. There was a period during the days I studied at UMBC when I had been harassed by insomnia. This was probably the darkest time of my life. It is my advisor's belief in me that keeps me going forward on pursuing my PhD degree. Dr. Finin had very good insights to research problems and gave me constantly advices and inspirations. He also allowed me the freedom in pursuing new ideas and approaches, which I liked it very much. I would say I owe a lot to Dr. Finin.

I am also grateful to the members of my dissertation committee, Anupam Joshi, Yelena Yesha, Li Ding and Paul McNamee. They are always available when I need their help. I thank them for their time and valuable feedbacks on my thesis. Prof. Anupam Joshi is a good teacher and I learned a lot in his class and from his words of suggestion. Prof. Yelena Yesha gave me many useful advices in the project in which we collaborated with SAP Business Objects. Dr. Li Ding gave me a lot of help on the Swoogle project that I participated in during my early days at UMBC. Discussions with him had a long-lasting influence on this research. Dr. Paul McNamee always gave me detailed and important comments on my research work, which I find very helpful to improve my work.

Pursuing PhD degree in Computer Science had been my dream for a long time and I would like to thank UMBC for giving me the opportunity to fulfill it. I met a lot of talented people and made many friends here. I would like to thank Prof. Yu Peng for guidance on my research work. I would like to thank my colleagues, Wenjia Li, Patti Ordonez, Justin Martineau, Yang Yu, Xianshu Zhu, Yi Sun, Palani Kodeswaren, Varish Mulwad, Yichuan Gui, Abhay Lokesh Kashyap, Yan Kang and Akshay Java for helping and encouraging me and all the happy time we spent together. I am truly indebted to them.

Finally but most importantly, thanks to my parents and my wife. My parents financially supported me with the money that they saved for many years, despite of their retirements. My wife has been accompanied me in all the difficult time, encouraging me and trusting me. Their love and faith in me is the most precious asset in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 Problem Definition

Developing interfaces to enable casual, non-expert users to query complex structured data has been the subject of much research over the past forty years. Such interfaces allow users to ad hoc query data without understanding its schema, knowing how to refer to objects, or mastering the appropriate formal query language. We call such interfaces as *schema-free query interfaces*.

Natural Language Interface (NLI) has been the most studied schema-free query interface. Allowing people to query a database or knowledge base in natural language has been a long standing goal in the fields of NLP, AI and database. NLI has seen much work since the 1970s [109, 44, 5, 37, 4, 102]. More recently there has been interest in developing NLI for XML data [60] and collections of semantic data encoded in RDF [66, 18, 106, 67, 24].

There are two major obstacles for NLI systems to be widely adopted, however. First, current NLP techniques are still brittle in addressing the ambiguity and complexity of natural language in general [4, 52]. For example, the user does not know which wordings of a question will succeed and which are beyond the system's linguistic coverage. Second, it requires extensive domain knowledge for interpreting natural language questions. Some systems need develop a syntactic or semantic grammar that is specific to a domain. Many systems use a mapping lexicon or a set of mapping rules that maps a user's vocabulary or expressions to an ontology vocabulary or logical expressions in NLI systems. A world model is also often required that specifies the relationships between the vocabulary terms (e.g., subclass relationships) and the constraints on the types of arguments of properties. All these domain knowledge is very expensive in terms of human labor involved.

The keyword interface can be viewed as another schema-free query interface. Research in querying structured data with keywords and phrases has gained popularity recently [47, 111, 100, 95]. Keyword query systems are more robust than NLI systems because they typically employ a much simpler mapping strategy: map the keywords to the set of elements in the knowledge base that are structurally or associationally close, such as the most specific sub-tree for XML databases [111] and the smallest sub-graph for RDF databases [100]. However, keyword queries have limited expressiveness and inherit ambiguity from the natural language terms used as keywords. For example, the keyword query "president children spouse" can be interpreted either as "give me children and spouses of presidents" or "who are the spouses of the children of presidents". Li et al. [60] compared the performance of a NLI system to a keyword system on a set of complex queries and showed that the keyword system performed poorly against the NLI system.

The problems become much more severe in the scenario of Semantic Web or Big Data where data is not restricted to a narrow domain and often has heterogeneous schemas. Some examples include Linked Open Data (LOD) [8], Freebase, DBpedia and Google Knowledge Graph. The traditional NLI approaches that heavily depend on manually created domain knowledge are no longer applicable. General-purpose and fully automatic approaches are required in order to query the large and rapidly growing semantic/structured data that we are facing today. In order to precisely query structured data, these approaches should provide such interfaces that enable the relational structure between the query's entities to be specified.

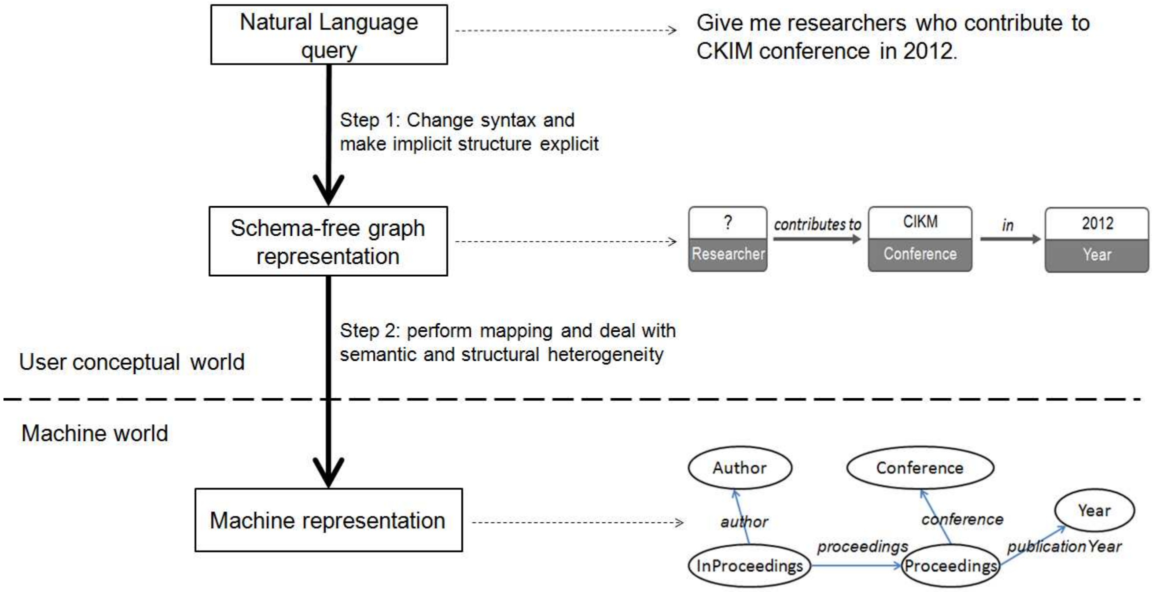

Natural Language query
Give me researchers who contribute to CKIM conference in 2012.
Step 1: Change syntax and make implicit structure explicit
Schema-free graph representation
?
Researcher
contributes to
CIKM
Conference
in
2012
Year
Step 2: perform mapping and deal with semantic and structural heterogeneity
User conceptual world
Machine world
Machine representation
Author
Conference
Year
author
conference
proceedings
publicationYear
InProceedings
Proceedings

FIG. 1.1. General-purpose NLI approach

1.2 Motivation

In this section, we will discuss the nature of general-purpose NLI, why the problem is hard, where the current research gets stuck. We will begin by motivating our approach and describing how we will tackle the problem.

NLI addresses a fundamental problem in NLP and AI because it bridges the user's conceptual world and the machine's representation, allowing people to seamlessly store, query and share their knowledge. Figure 1.1 presents the high-level general-purpose NLI approach that one normally takes. Though represented as a sequence of words, natural language queries have implicit structure. Therefore, the first step of interpreting a natural language query is to make the implicit structure explicit. The resulting intermediate representation is referred by us as schema-free graph representation. One example is shown in Figure 1.1. In this step, we only change the syntax of the query but keep the structure and content, namely the modeling of the query, unchanged. Currently, a NLP parser is required to carry out the first step. The second step is to map the schema-free graph query into the underlying machine representation. Because people have many different ways to express (model) the same meaning, semantic and structural heterogeneity exists between the user's modeling of the query and the machine's modeling, which we have to address in the second step. Consider the example in Figure 1.1. The relation "researcher contributes to conference" in the user query needs to be mapped to a schema path with three hops that connects the *Author* class and the *Conference* class. Neither string similarity nor synonyms exist between the user term "contributes to" and any of the terms in the schema path.

State of art NLP parsers still cannot perform the first step reliably. Syntactically correct parsing may not be semantically correct. General-purpose NLP parsers know little about semantics and they cannot tell how much sense a particular parsing result, a schema-free graph query, makes. Actually, this is the job of the second step. By mapping the schema-free graph query into the machine representation, we can come up with a score that describes how semantically reasonable the mapping is. However, up to date no proper work has been seen on the step two probably because this is a challenging work without user interface. Thus we fall into a kind of chicken-egg problem.

There is an interesting solution to break the circle. The intermediate schema-free graph representation can actually be used as a user interface. Because this representation is still in the user's conceptual world, it is also a *schema free query interface*. We conducted a preliminary user study about the interface on a small group of people. We trained them for half an hour and the asked them to create scheme-free graph queries for 33 natural language questions in Table 9.11. No one had difficulty in creating them[1]. The resulting queries are used to form an evaluation dataset for our system. Hereafter, we will call the Scheme-Free graph Query simply as SFQ and the new interface as SFQ interface. By using

[1] The reader is referred to Section 9.4.2 for more details

SFQ interface, we work around the step one at which current research is stuck. We can then build systems on the step two directly, which also reach the goal of allowing people to access and share knowledge. These systems, once mature, can in turn help with the first step.

This thesis is focused on step two. We describe a framework for interpreting SFQs over open domain semantic data and automatically translating them to formal queries, for example, SPARQL[2]. Semantic data in this thesis refers to structured data that are organized as a network of entities. The entities are typed with classes and interlinked with properties. Classes and properties are named with human-readable words or short phrases. Semantic Web RDF data is a typical example. Our approach does not require an ontology (schema) to be defined for semantic data. Instead, we introduce a method to learn ontological knowledge statistically from data itself.

Interpreting a SFQ means *semantically* mapping a SFQ to its corresponding schema graph in the knowledge base. This inevitably involves a problem that how semantics is captured. Previous systems use domain-specific grammars, mapping lexicons and mapping rules to capture semantics, but all these are hard-coded semantics. A thesaurus like WordNet allows certain flexibility by providing synonyms and hypernyms, but it is not enough to address the semantic heterogeneity problem. For example, “marry” has very similar meaning to “wife”, but none of the existing WordNet relations capture or even suggest this association. In contrast, we employ statistical and computational semantic similarity measures to capture lexical semantics, which enables our system to have a much broader linguistic coverage. Our semantic similarity measures are automatically learned from a three-billion words general text corpus, but they can also be learned from a domain-dependent corpus if domain-specific semantics is required.

[2] SPARQL is the standard query language for RDF

Lexical semantic similarity between words has been well studied and developed, but research in structural semantic similarity is still in its early age. For example, the best text semantic similarity measure from the 2013 STS task [41] only uses the content of the text (i.e. bag of words) and totally ignore structural features. In this thesis, we utilize lexical semantic similarity measures to develop a structural semantic similarity metric between a SFQ graph and a schema graph, which takes into account both the content and structure of the graphs. This structural semantic similarity metric can be used to evaluate how well a mapping is.

There is an important element that we need to complement lexical semantic similarity in carrying out the mapping as well as constituting the structural semantic similarity metric. This element is the association degree between schema terms (i.e. classes and properties). Some schema terms are more likely to co-occur and others less likely. This information is very important in performing disambiguation when multiple choices exist. Actually, our human brains also depend on this kind of information to understand language. For example, given the phrase "database table", a person can quickly pick the right meaning of "table" from several conceptual choices because of the "association model" she learned from her life experience. In this thesis, we introduce two statistical models for measuring association degree of schema terms, which are used in different phases of the mapping algorithm. We build the models automatically from data we are going to query against.

To better deal with semantic and structural heterogeneity problems, which are often intertwined, we perform context-sensitive mapping at relation-level and joint disambiguation at graph-level. Three simple examples are shown in Figure 1.2 to demonstrate why context-sensitive mapping is necessary. In each example, a single-relation SFQ is mapped to its corresponding schema path in a knowledge base. In the three examples, the same predicate "publish" is mapped to three different properties *author*, *institution* and *publisher*. A simple mapping-lexicon approach will fail because correctly mapping "publish"

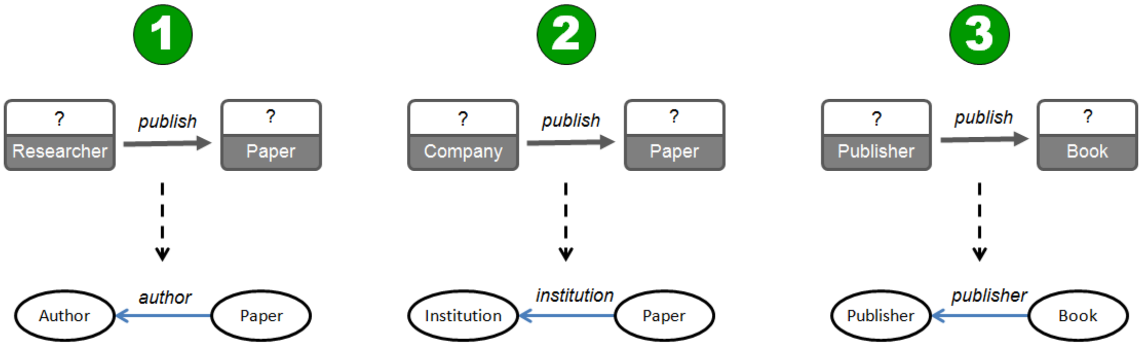


FIG. 1.2. Three simple examples illustrating context-sensitive mapping

requires understanding the context. Our context-sensitive mapping algorithm can successfully handle all these cases even though the semantic similarity between “publish” and *institution* is very low in the second example because we map the relation as a whole and the predicate “publish” is semantically aligned with the class *Paper* in the schema path, but not its syntactic counterpart, *institution*. We need carry out joint disambiguation at graph-level because the locally optimal mapping at one relation in a SFQ is not necessarily the best mapping when the whole graph is considered and whether it is depends on how well the other relations in the SFQ are mapped.

Efficiency is something we have to consider in querying large collection of semantic data. Our mapping algorithm operates at the concept-level (i.e. schema-level) and relies on statistical information that are automatically learned from instance data. This makes the approach much more scalable than those that directly search into both instance and concept data for possible matches since concept space is much smaller than instance space. However, dealing with the structure mismatch problem, even in concept space, still needs to go through a large number of mapping choices. We develop a novel algorithm that decompose the mapping problem into two ordered sub-problems and dramatically reduce the number of mapping choices we need to evaluate.

We evaluate our prototype system on two datasets, DBLP+ and DBpedia. DBLP+

is a dataset in Computer Science publication domain while DBpedia is a open domain dataset, representing structured data from Wikipedia. Both datasets are large, containing tens of millions of facts or triples each, but they have very different natures. DBpedia has a broad and relatively shallow domain whereas DBLP+ has a narrow but deeper domain. We carried out cross-validation on the DBLP+ dataset and cross-domain validation on the DBpedia dataset, in which the parameters tuned for a collection of the DBLP+ queries are applied to a collection of the DBpedia queries. Our prototype system showed very good performance on both of the datasets.

1.3 Thesis Statement

We can develop an effective and efficient algorithm to map a casual user's schema-free query into a formal knowledge base query language that overcomes vocabulary and structure mismatch problems by exploiting lexical semantic similarity measures, association degree measures and structural features.

1.4 Contributions

- An intuitive SFQ interface that avoids the problem of extracting relations structure from natural language queries.

- Being the first general-purpose open domain automatic approach that addresses both vocabulary and structure mismatch problems in mapping schema-free queries to formal queries.

- A novel algorithm that decomposes the mapping problem into two sub-problems: *concept mapping* and *relation mapping*, which greatly improves efficiency by dramatically reducing the number of mapping choices to be considered.

- A novel joint-disambiguation *concept mapping* algorithm that maps all the concepts in a SFQ to their corresponding classes in the knowledge base, which makes use of both semantic similarity and association degree measures.
- A novel context-sensitive *relation mapping* algorithm that maps relations in a SFQ to schema paths in the knowledge base, which takes into account both content and structure features.
- A novel approach to handle heterogeneous or unknown schemas by building a schema network from an entity network.
- An improved PMI (Pointwise Mutual Information) metric that offsets the bias of standard PMI [17] toward low frequency terms.
- Two novel statistical models that are automatically learned from instance data ("ABox") to represent ontological association knowledge and provide two metrics, (1) schema path probability and (2) the improved PMI, for measuring association degree.
- Novel and state of the art lexical semantic similarity and textual semantic similarity metrics.
- A 3-billion-words text corpus of English paragraphs.
- A 2-billion-words text corpus of Project Gutenberg English books.

1.5 Thesis Outline

The rest of this thesis proceeds as follows.

In Chapter 2, we discuss related work and background of this research.

In Chapter 3, we describe the concept of SFQ interface and its implementation, including an envisioned graphical interface and a text interface.

In Chapter 4, we describe the association models used by our approach and elaborate how we count co-occurrence statistics and build the models from instance data.

In Chapter 5, we describe how to interpret a SFQ, namely, the mapping algorithm. We start by presenting the approach outline and explain why we need decompose the mapping problem into two sub-problems. Then we elaborate the concept mapping algorithm and the relation mapping algorithm in order.

In Chapter 6, we show how we deal with entity matching and generating formal queries from the output of the mapping algorithm.

In Chapter 7, we describe how we derive an improved PMI metric that offsets the bias of standard PMI toward low frequency terms. The improved PMI metric is used to measure association degree by one of the two association models.

In Chapter 8, we discuss how we develop the semantic similarity measures used by the mapping algorithm and evaluate them using standard datasets/tasks.

In Chapter 9, we evaluate our prototype system on two large datasets, DBLP+ and DBpedia.

In Chapter 10, we conclude our paper and plans for future work.

Chapter 2

BACKGROUND AND RELATED WORK

2.1 Relation Extraction

A line of research closely related to our motivation for using SFQ interface is relation extraction. Parsing the full relational structure between the entities in natural language sentences remains a problem far from being solved, although we have made some progress over the years.

Modern dependency parsers [63, 26] can achieve about 90% precision and 80% recall, but what they generate are grammatical relations between individual words rather than semantic relations between entities. Some simple semantic relations, but not all the relations, can be obtained by analyzing the output of the parsers. If a SFQ query contains multiple relations and/or multiple answer types (variables) and/or entity names of less common types, its corresponding natural language queries would inevitably involve some linguistic phenomena hard to deal with, such as modifier attachment, anaphora and fine-grained named entity recognition. It is also likely that the query has to be described in more than one sentence and then the parsers need to deal with coreference resolution across sentences. Reliably solving these problems require semantics [4], which current parsers know little about.

The best relation extraction systems often rely on machine learning models to extract relations and use dependency parsers to produce features [13, 51]. The systems focus on extracting a small set of predefined relations, for example, LOCATED and FOUNDED, but not all kinds of the relations. Even so, their performance is still far from reliable when evaluated on the Automatic Content Extraction (ACE) tasks conducted by NIST.

Relation Extraction shares much similarity with Information Extraction (IE) since it is the major task of IE. A variety of techniques have been applied in IE, such as linguistic analysis, machine learning, pattern recognition and hand crafted rules. Traditional IE often begins with a given ontology that defines a set of target concepts and relations focusing on a particular domain and aims to collect as many instances as possible to populate the target ontology. The earliest IE methods require hand-crafted rules or a lot of human annotated training instances, either of which is both labor-intensive and heavily tuned to a particular domain. DIPRE [10] and Snowball [1] significantly reduce the human effort by requiring only a small set of seed instances. The seed instances are used to find new patterns statistically and in turn the new patterns can be used to find more instances. In the ontological IE, the extracted patterns (relations) and facts need to be mapped to the corresponding entities in the ontology, in which a disambiguation process is necessary.

Recently, there emerged a new IE paradigm, called Open IE (OIE) [91, 6, 15]. OIE is unsupervised extraction without any human intervention. OIE takes as input a corpus of text documents and is supposed to generate all the instances of all the relations in the corpus. A good example is TextRunner [6]. TextRunner is a generic and scalable OIE, which runs on the corpus of millions of web pages without discriminating the topics of the web pages and extracts all the instances with the triple form <*entities A, relation X, entities B*>. Finally the instances are organized to different clusters according to their relation name. Although current OIE systems are claimed to extract all kinds of relations, in fact they still selectively extract relations with certain patterns that their parsers can recognize. Moreover, the extracted facts in current OIE systems still stay at their surface representation– completely separate triples which are not interlinked to one another.

2.2 NLI Systems

NLIDB (NLI to Database) systems have been extensively studied since the 1970s [4] and typically took NL sentences as queries and used syntactic, semantic and pragmatic knowledge to produce corresponding SQL queries. Early systems like LUNAR [109] and LADDER [44] were heavily customized to a particular application and difficult to port to other application domains. These systems required developing syntactic or semantic grammars that are specific to an application domain. Later systems, including TEAM [37], ASK [97] and MASQUE [5], were designed to be portable. They used a more general parser but still required human-crafted semantic rules and domain knowledge to interpret the parse tree. The later systems also allowed knowledge engineers to reconfigure the system before moving to a new domain and/or letting end users add unknown words or expressions through user interaction. A common problem of the NLIDB systems in 70s and 80s is that they had a restricted linguistic coverage since they depended on hard-coded semantics. The domain-specific parsers and the semantic rules can fail to tolerate even a slight change in the wording of a question. In sum, these systems can obtain good performance but only for narrow application domains.

PRECISE [80], a more recent NLIDB system, reduced question interpretation to a maximum bipartite matching problem between the tokens in a NL query and database elements. PRECISE only used a tokenizer to get all the tokens in a natural language question and totally ignore the syntactic relations between the tokens. This makes it more like a keyword-based approach. PRECISE used a lexicon that stores possible matches between the tokens and the database elements. Therefore, it still relied on hard-coded semantics.

Learning semantic parsers are systems that use machine learning methods to map the

NL questions into logical forms. These systems, such as SCISSOR [35], have shown good performance but require manually annotated training data, which is expensive to obtain. Moreover, the effectiveness of these systems has only been evaluated on very restricted domains.

Recently, a number of portable NLI systems have been developed for ontologies [66, 18, 106, 24] and XML databases [60]. They are the follow-up of the work in NLIDB but in different problem domains.

NaLIX [60] translates NL questions to XML queries by mapping the adjacent NL tokens in the parse tree to the closely connected XML elements in the database. NaLIX addressed the structure mismatch problem by allowing the XML elements to be connected by paths. However, it left the vocabulary mismatch problem untouched because it assumed that the NL tokens can be matched exactly or at most by synonym extension.

ORAKEL [18] constructs a logical lambda-calculus query from a NL question using a recursive computation guided by the question's syntactic structure. However, it still requires human-crafted lexicons to interpret the logical query. ORAKEL has an interactive graphical interface that allows a lexicon engineer to add new mapping rules to the lexicons.

FREyA [24] generates a parse tree from a NL question, maps linguistic terms in the tree to ontology concepts, and formulates a SPARQL query from them by exploiting their associated domain and range restrictions. FREyA uses dialogs to interact with the user, where the user can specify the mappings from linguistic terms to ontology concepts.

Aqualog [66] and PANTO [106] translate the NL query to linguistic or query triples and then lexically match linguistic triples to ontology triples. Aqualog performs shallow parsing to obtain an ordered list of pos-tagged tokens from NL questions and then use a set of manually-made pattern rules to generate question types and linguistic triples. In contrast, PANTO applies a head-driven algorithm to collect linguistic triples from the syntactic strucutre of the parse tree. Both Aqualog and PANTO disregard the structure mismatch problem and assume a linguistic triple will be mapped to only one ontology triple. PANTO use a synonym lexicon to partially deal with the vocabulary mismatch problem. Aqualog uses a learning mechanism to obtain a domain-dependent lexicon from user interaction. The disambiguation mechanism of Aqualog is also interactive. It will ask the user for disambiguation whenever it fails to interpret a query.

The last few years have seen a growing interest in open domain NLI systems. True Knowledge[1] [102] and PowerAqua [67] are two good examples. Both of them choose pragmatic approaches to turn NL questions into relations. True Knowledge creates 1,200 translation templates to match NL questions. True Knowledge supports user interaction and exploits a repository storing user rephrasing of the questions it cannot understand. True Knowledge can carry out inference once the user question has been represented in logical forms but the step of translating NL questions to logical forms still heavily depends on hard-coded semantics. It lacks a computational approach to address the structure mismatch problem, although the thousand templates can help reduce it.

PowerAqua is an entension of Aqualog and it uses the same method as Aqualog but with a larger number of hand-created pattern rules to produce question types and relations. PowerAqua added components for merging facts from different ontologies and ranking the results using confidence measures. Similar to Aqualog, PowerAqua maintains mapping lexicons to deal with the vocabulary mismatch problem. In order to improve recall, PowerAqua adopts partial match. It will match a linguistic triple to an ontology triple as long as the subjects and objects match, regardless of the predicates. PowerAqua further extends this partial match approach to deal with the structure mismatch problem. It will match a linguistic triple to an ontology path of two triples if the two endings of the path can be matched with the subject and object of the linguistic triple. PowerAqua runs a po-

[1] Now, it has a new name, Evi.

tentially expensive graph matching algorithm that searches in the RDF knowledge base for the matchings of the query graph at both entity and schema levels.

Treo [33] is the only system we have seen which also uses semantic similarity measures to address the vocabulary mismatch problem. Treo [33] reduce a NL query to a list of ordered terms guided by the query's dependency structure and matches the terms to an RDF path using semantic similarity measures. It requires recognizing a named entity as the pivot starting a spreading activation process based on a semantic similarity threshold. However, Treo has many limitations. First, Treo can only work for queries in the form of a path. It cannot handle graphs or trees. Second, Treo cannot produce exact answers or SPARQL queries. Instead, it outputs a ranked list of triple paths that may contain the answers. Third, Treo does not address the structure mismatch problem. Fourth, it requires a named entity in the query.

2.3 Keywords Interface System

Substantial research has been done on applying keyword search on structured data, including relational database [47], XML [111, 95] and RDF [100]. Such keyword-based approaches cannot express complex queries and often mix textual content from meta-data and data. A number of approaches [20, 56] extend keyword queries with limited structure information, allowing to specify entity types and attribute-value pairs. However, they are still unable to support querying complex semantics.

A research effort closely related to ours is Schema-Free XQuery [61], as it also seeks a middle ground between XQuery (the formal XML query language) and keywords query. The approach allows casual users to create a XQuery without specifying the path expressions between meaningfully related nodes. A new XQuery operator MLCAS is introduced for the user to manually group or associate the nodes. On the other hand, MLCAS also stands for a XML structure that defines the most specific query context for the related nodes. MLCAS extends the notion LCA (Lowest Common Ancestor) that is commonly used in the XML keyword-based approaches. The Schema-Free XQuery approach focuses on addressing the structure mismatch problem while assuming there is no vocabulary mismatch problem or it can simply be solved by synonym expansion. Although it spares the user from specifying the exact path expressions, the Schema-Free XQuery approach still requires the user to understand and use the XQuery syntax. Moreover, the approach is not applicable to graphs because it adheres to the LCA structure that is only available for trees.

Chapter 3

SFQ INTERFACE

In this chapter, we first talk about the concept and rules of the SFQ query language. We then use one example to illustrate our graphical web interface that is still under development and introduce the important concepts of our approach. Finally we show a way to serialize the SFQ query, which is used to communicate with our system for the present.

3.1 SFQ Concept and Rules

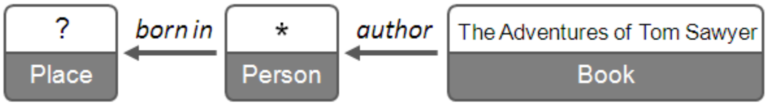


FIG. 3.1. A Schema-Free Query for "Where was the author of the Adventures of Tom Sawyer born?".

A SFQ query is represented as a graph with nodes denoting entities and links representing semantic relations between them. Each entity is described by two unrestricted terms: its name or value and its concept (i.e., class or type). Figure 3.1 shows an example of a SFQ with three entities (a place, person and book) linked by two relations (*born in* and *author*). Users flag entities they want to see in the results with a '?' and those they do not with a '*'. Terms for concepts can be nouns (*book*) or simple noun phrases (*soc-*

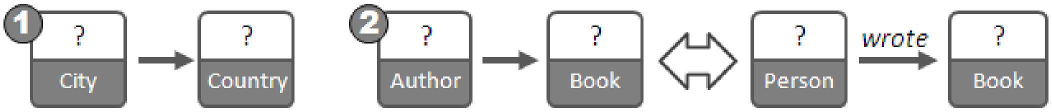


FIG. 3.2. Two examples of default relation.

cer club) and relations can be verbs (*wrote*), prepositions (*in*), nouns (*author*), or simple phrases (*born in*). Users are free to reference concepts and relations in their own words as in composing a NL question.

Relation names are optional and can be omitted when there is a single "apparent" relation between two concepts that corresponds to the user's intended one. The "apparent" relation, which we call the *default relation*, is typically a *has-relation* or *in-relation*, as shown in the examples in Figure 3.2. In the first example, a *has-* or *in-relation* exists between *City* and *Country* and in the second, a *has-relation* also exists between *Author* and *Book*. Our system uses a stop word list for filtering relation names with words like *in*, *has*, *from*, *belong*, *of* and *locate*. In this way, a *has-* or *in-relation* is automatically turned into a default relation.

A default relation can be represented in other ways. Consider the second example in Figure 3.2. An author is a person who writes. Since the semantics about the relation "wrote" is embedded in the concept "Author", the relation can be left out when the concept "Author" is used. However, if the concept "Person" is used, we do need explicitly name the relation "wrote". This example illustrates an interesting phenomenon: semantics can stretch across syntactically neighboring terms. We refer to this phenomenon as *semantic stretch*. We are going to revisit it when we discuss the relation mapping algorithm in Section 5.3.

We currently expect concept names from users, enabling our system to resolve mappings in concept space rather than instance space. The requirement stems from the observation that people find it easy to explicitly tag the types but it is much harder for machines to infer them. However, we are developing techniques to relax this, as described in the Section 5.2.

The value of entities can be something other than a name, such as a number or date. If the value of an entity is a number, “Number” is used as the entity’s concept. Numerical attributes such as population, area, height, and revenue can be thought of as either relations or concepts, but since *Number* is already used as the concept, we require them to be relations. Following this simple rule helps our system better understand the user’s query. However, we have also developed a backup approach in the case that the user is unaware of this rule, which is described in Section 4.1.

Like a typical database query language, SFQ can express factual queries but not *why* or *how* questions. We currently support neither numerical restrictions on entity value nor aggregation functions working on the entity in question. We plan to implement these features using form-based fields and drop-lists beside the graph area for creating SFQ. Alternatively, we may implement them as a set of buttons that can be dragged and applied to the entity variables.

3.2 Envisioned Graphical Web Interface

Figure 3.3 shows the start page of our our envisioned graphical web interface. The left panel is the drawing area for the user to create a SFQ query. The right panel is the place where the user can apply numerical restrictions, *order by* clause or aggregation functions to the entity being selected in the left panel using form-based fields and drop-lists. One SFQ example is given in the left panel, which asks about the actresses worked with the director Woody Allen.

There are two buttons beneath the panels, which resemble the buttons in the start page

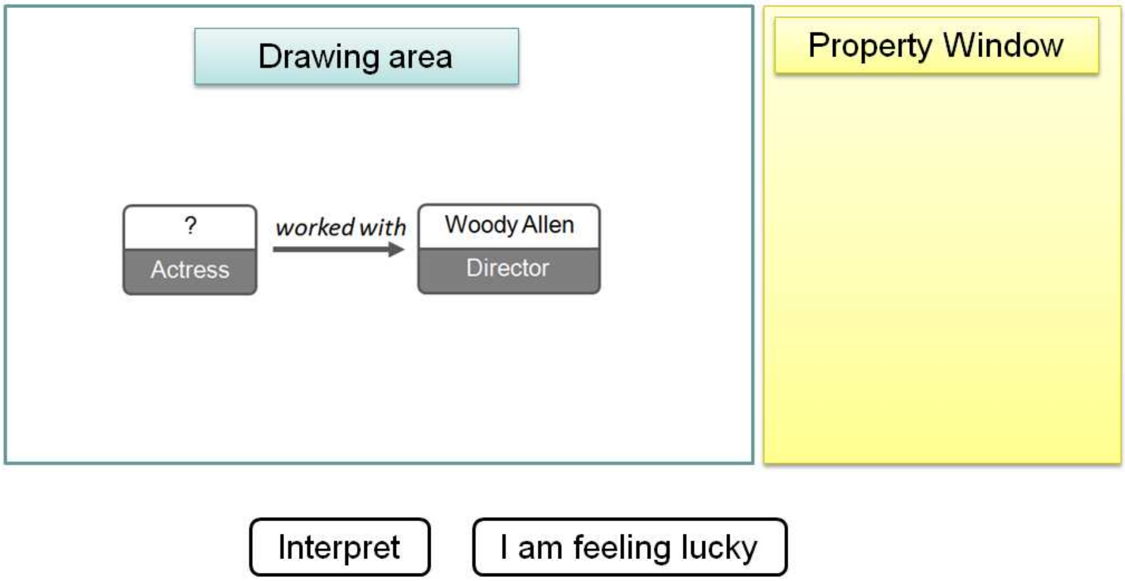


FIG. 3.3. The start page of our envisioned web interface

of Google. If the "I am feeling lucky" button is clicked, the user is brought to the possible answers of the SFQ query in the form of a table. However, there is a chance that the answers are wrong. Actually, this is a problem of all NLI systems. The user cannot trust the answers of a NLI system because no system up to date can return fully correct answers all the times. To address this problem, we provide the "Interpret" button, which brings the user a new page with the top 10 interpretations of the SFQ query. An interpretation refers to a mapping of the SFQ query.

Figure 3.4 shows the page with the top 10 interpretations for the example query in Figure 3.3. This is the real output that is produced by our system on the DBpedia dataset. Each interpretation is associated with a score that indicates the computed semantic similarity between the interpretation and the user's query. The user can go through the interpretations and select the one that best fits her query. In this way, the user can know not just answers but also how the answers come from. Regarding to this example, the first interpretation

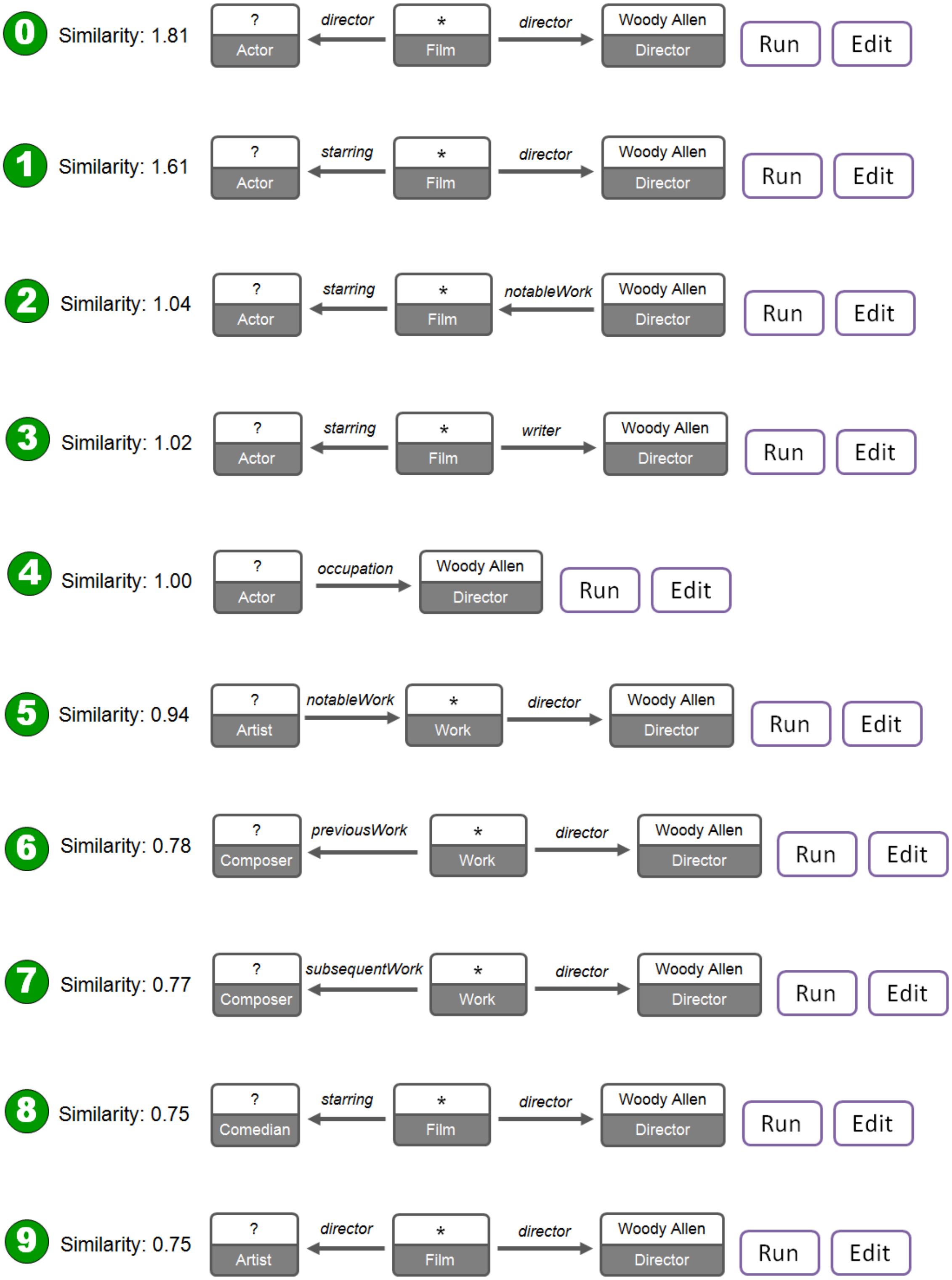


FIG. 3.4. The page with top 10 interpretations after clicking the *Interpret* button

generated by our system is actually wrong. Our statistical models currently are not able to tell that a film only has a single director and the co-director relation does not exist, but it is pretty easy for a human to know it. The second interpretation is the one that most fits the user's query, which is also the best one among all possible DBpedia interpretations because there is no "Actress" class in DBpedia. As you can see, the goal of our system is to return the most semantically similar interpretation, which is not necessarily a completely correct interpretation.

The interpretations are also represented in the SFQ query format so that the user can further update the interpretation if necessary. The "Edit" button is made for this purpose. After clicking it, the user will go back to the start page with the new SFQ query.

3.3 Serializing a SFQ query

A SFQ query can be serialized into a text of multiple lines, with each line representing a relation or an edge of the query graph. Figure 3.5 illustrates a serialization example for the SFQ in Figure 3.1.

```
The Adventure of Tom Sawyer/Book, author, *x/Person
*x, born in, ?y/Place
```

FIG. 3.5. An example of SFQ serialization.

A relation consists of three elements in order, the subject entity, the predicate, and the object entity, which are separated by comma. An entity is described by two terms: its name or value and its concept (type), which are seperated by '/'. The entities starting with '?' are variables, whose values will be shown in the query result. You can name a variable as '?x', '?y', or anything meaningful to you. '*x', like '?x', is also a variable. However, it is a hidden variable, whose value will not be shown in the query result. So you can controll what entities to be presented in the query result by using ’?’ or ’*’. In the second relation of the example, ’*x’ refers to the ’*x/Person’ in the first relation. After an entity has been assigned a type, it can be referenced later with its name only.

Chapter 4

STATISTICAL ASSOCIATION MODELS

This chapter describes two statistical association models that our query system depends on. The first model, we call schema path model, defines a metric that measures the degree to which schema terms can keep company in the form of a *path*. This metric can be uniformly applied to paths of different length. The second model, we call CAK (Concept Association Model), provides *pairwise* association degree between schema terms.

Without loss of generality, this chapter uses Semantic Web RDF data to explain the process of building the statistical models from instance data. Meanwhile, we provide definitions of some fundamental concepts that will be used throughout this thesis.

4.1 Data and Preprocessing

We store instance data in two files. The first file contains all relations (triples) between instances, which is called the *relation file*. The second file provides all type definitions for the instances, which is called the *type file*. The instance data is corresponding to the ABox[1] data in a knowledge base. Our system takes as input the two data files and generates the association models automatically.

Five common sense data types are pre-defined in order to integrate all data types oc-

[1] See `http://en.wikipedia.org/wiki/Abox` for the definition

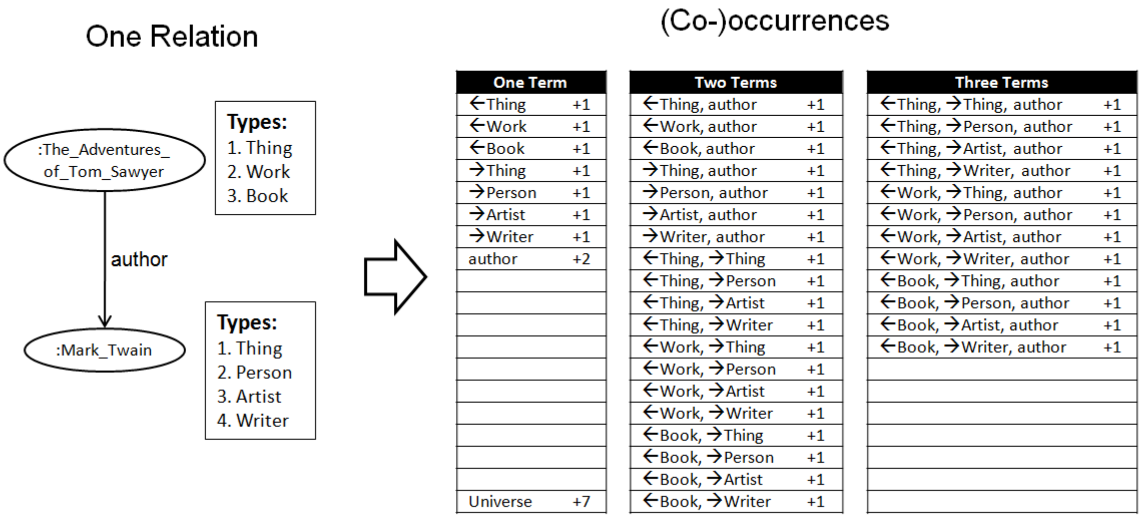
curring in RDF data. They are $\hat{N}umber$, $\hat{D}ate$, $\hat{Y}ear$, $\hat{T}ext$ and $\hat{L}iteral$, of which $\hat{L}iteral$ is the super type of the other four[2]. $\hat{N}umber$ is used for representing all numerical data types (e.g., *xsd:integer*, *xsd:float*). These five pre-defined data types are created for the purpose of facilitating the alignment on attribute types between the user's conceptual world and the machine's representation.

As we discussed in Section 3.1, we expect the user to use the pre-defined data type, "Number", as the concept for describing numerical attributes. However, the user may not be aware of this rule. For example, instead of creating $\frac{Beijing}{City} \xrightarrow{area} \frac{?}{Number}$, she may still compose $\frac{Beijing}{City} \xrightarrow{has} \frac{?}{Area}$. Some numerical attributes, such as "Area", have low computed semantic similarity with "Number", making it difficult for our system to align them. To deal with this problem, we developed an approach that automatically deduce attribute types from datatype properties used in the RDF data. Datatype properties are typically referenced as nouns or noun phrases, which can actually work as attribute types. If they are noun phrases (e.g. "urbanArea"), we use their head noun as the attribute type (e.g. $\hat{A}rea$).

For Semantic Web data that does not have a properly defined ontology or has an incomplete type system (e.g. DBpedia), it is also helpful to learn types from object properties used in the RDF data. We call the learned types as *inferred classes*. Many property names are nouns or noun phrases, which can be used to infer the type of the object instance. For example, the object of the *religion* property should be a religion and therefore is an instance of $\tilde{R}eligion$[3]. The same method as for deriving attribute types is applied to learning inferred classes.

[2] Data types belong to attribute types. We cap the first letter of attribute types with $\wedge$ to distinguish them from other types.

[3] We cap the first letter of inferred classes with $\sim$ to distinguish them from other types.



One Term	
←Thing	+1
←Work	+1
←Book	+1
→Thing	+1
→Person	+1
→Artist	+1
→Writer	+1
author	+2
Universe	+7

Two Terms	
←Thing, author	+1
←Work, author	+1
←Book, author	+1
→Thing, author	+1
→Person, author	+1
→Artist, author	+1
→Writer, author	+1
←Thing, →Thing	+1
←Thing, →Person	+1
←Thing, →Artist	+1
←Thing, →Writer	+1
←Work, →Thing	+1
←Work, →Person	+1
←Work, →Artist	+1
←Work, →Writer	+1
←Book, →Thing	+1
←Book, →Person	+1
←Book, →Artist	+1
←Book, →Writer	+1

Three Terms	
←Thing, →Thing, author	+1
←Thing, →Person, author	+1
←Thing, →Artist, author	+1
←Thing, →Writer, author	+1
←Work, →Thing, author	+1
←Work, →Person, author	+1
←Work, →Artist, author	+1
←Work, →Writer, author	+1
←Book, →Thing, author	+1
←Book, →Person, author	+1
←Book, →Artist, author	+1
←Book, →Writer, author	+1

FIG. 4.1. This example shows how we count term occurrences and co-occurrences in an RDF knowledge base.

4.2 Counting Co-occurrence

All statistical metrics in our models are based on co-occurrence counts of schema terms. Therefore, we first discuss how we count them. The *relation file* contains all relations between instances, from which we can obtain co-occurrences between entities. Although entity co-occurrences are not what we are interested in, we can actually count co-occurrences between schema terms indirectly from co-occurrences between entities because entities are associated with types.

Figure 4.1 shows how we count term occurrences and co-occurrences for one relation. On the figure's left is an RDF triple describing a relation and the type definitions for its subject and object and on the right are the resulting occurrences and co-occurrences of terms. We record two-term and three-term co-occurrences, both of which will be used in deriving our association models.

We consider direction in counting co-occurrences between classes and properties. To provide a consistent notation, we must decide whether to associate the direction with a class or a property. We choose to associate the direction with classes and call such classes as *directed classes*.

Definition 4.2.1. DIRECTED CLASS. $\rightarrow c$ is defined as the class c when it is used with incoming properties and $\leftarrow c$ is defined as the class c when it is used with outgoing properties. An inverse operation is defined on directed classes such that we have $(\rightarrow c)^{-1} = \leftarrow c$ and $(\leftarrow c)^{-1} = \rightarrow c$. To complement it, we also define $(\rightarrow c)^{1} = \rightarrow c$ and $(\leftarrow c)^{1} = \leftarrow c$.

Whenever a directed class and a property are given together, we can tell the direction regardless of the order they appear. However, if we associate the direction with properties, knowing the order of the class and the property is then a requisite for resolving the direction.

Because an instance can have multiple types, the fact that *Mark_Twain* is the object of the property *author* results in four directed co-occurrences between the property *author* and each of the types of *Mark_Twain*. Similarly, by observing that *The_Adventures_of_Tom_Sawyer* and *Mark_Twain* are the subject and object of the relation we can produce twelve pairwise directed co-occurrences between their types as well as twelve directed three-term co-occurrences. Although many term co-occurrences are generated, the frequency count of directed classes only increase by one. We can make an analogy between this and the counting of word co-occurrences in a text corpus where sliding the context window forward one word can result in many co-occurrences between the target word and all other words in the window.

We use $|a|$, $|a, b|$ and $|a, b, c|$ to denote the total number of term occurrences or co-occurrences. For example, $|\leftarrow Book,\ author|$ represents the number of times that *author* is used as an outgoing property of a *Book* instance in the RDF data. For another example, $|Writer|$ represents the number of times that *Writer* instances occur in the data and we can also deduce that $|Writer| = |\leftarrow Writer| + |\rightarrow Writer|$. The order of the items is irrelevant to their co-occurrence count. For example, $|\leftarrow Book, \rightarrow Writer,\ author| = |\rightarrow$

$Writer, \leftarrow Book,\ author|$.

4.3 Schema Path Model

In this section, we introduce two important concepts: (1) schema network and (2) schema path and then we define schema path probability on the schema network. Furthermore, we describe optimization technique to improve the space efficiency of the schema path model.

4.3.1 Defining Schema Path Probability

The statistical information that we learn from RDF data actually forms a meta description of the underlying entity network, which itself is also a network. We call such a meta network as *schema network*. The schema network is the target space which our mapping algorithm will search into.

Definition 4.3.1. SCHEMA NETWORK. The schema network is defined as a both node and edge labeled, weighted and directed graph $G_s = (C, R, P)$ with the node set C and the edge set R, where (1) C contains all the classes; (2) P is a set that contains all the properties; and (3) $R \subseteq C \times P \times C$ is the set of edges, in which (c, p, c') represents a relation directed from class c to c' and has a weight of $|\ \leftarrow c, \rightarrow c',\ p|$.

A path on the schema network is called a schema path. Consider two examples: (1) $Actor \overset{starring}{\leftarrow} Film \overset{director}{\rightarrow} Director$; and (2) $Author \overset{author}{\leftarrow} Paper \overset{cites}{\rightarrow} Paper \overset{author}{\rightarrow} Author$. The first example represents the indirect *worked with* relation between a actor and a director and the second one denotes the indirect *cites* relation between two authors. Before we formalize the definition of a schema path $\mathcal{P}$, we first define an inverse operation on a relation (c, p, c') such that we have $(c, p, c')^{-1} = (c', p, c)$. To complement it, we also define $(c, p, c')^1 = (c, p, c')$.

Definition 4.3.2. SCHEMA PATH. A schema path $\mathcal{P} = (\mathbf{C}, \mathbf{P}, \mathbf{D}, l)$ is defined as a path on the schema network $G_s = (C, R, P)$, where (1) l denotes the length of the path; (2) $\mathbf{C}$ is a vector of length $l+1$ that represents the sequence of classes $\langle c_0, c_1, c_2, \cdots, c_l \rangle$ in the path; (3) $\mathbf{P}$ is a vector of length l that represents the sequence of properties $\langle p_0, p_1, p_2, \cdots, p_{l-1} \rangle$ in the path; and (4) $\mathbf{D} \in \{-1, 1\}^l$ is a vector of directions $\langle d_0, d_1, d_2, \cdots, d_{l-1} \rangle$ such that for every $i \in \{0..\, l-1\}$, $(c_i, p_i, c_{i+1})^{d_i} \in R$ represents an edge in the path.

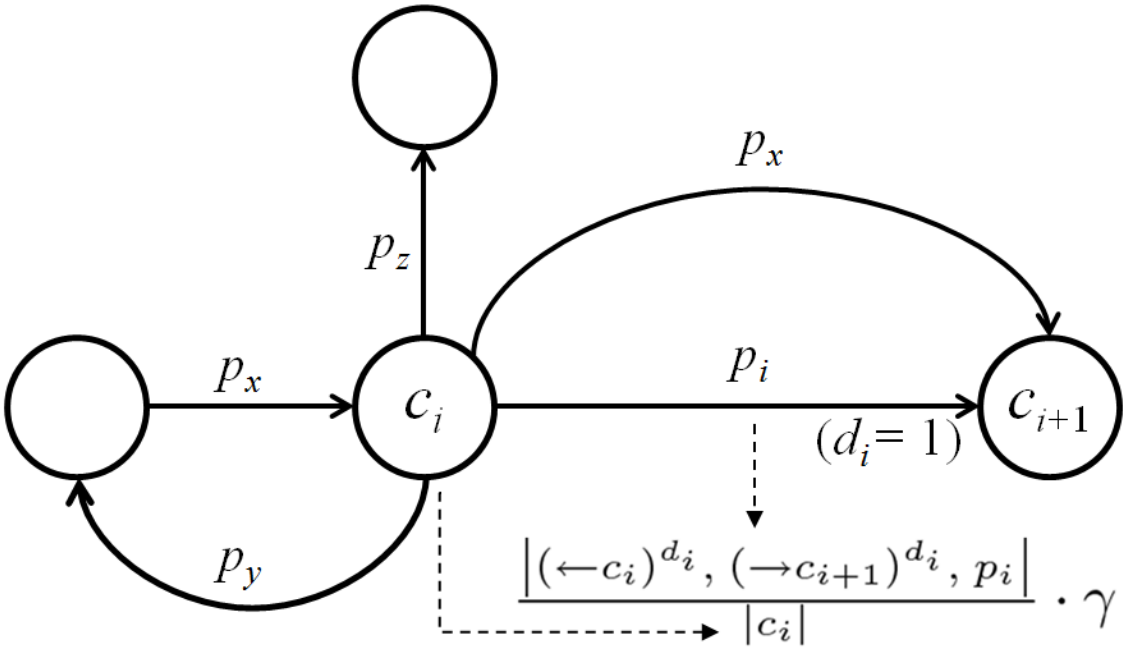
A schema path $\mathcal{P}$ represents a composite relation. Some schema paths make more reasonable relations than others do. We need develop a way to measure the reasonableness of a schema path. Our intuition is that the more likely a schema path is to occur in the schema network, the more reasonable relation the path makes. Therefore, the problem is reduced to acquiring or estimating the probability of “observing” a schema path $\mathcal{P}$ in the schema network. We compute it as the probability of the joint event that (A_1) we select the starting node c_0 of the path randomly from all the nodes in the schema network and then (A_2) observe the path in a random walk starting with c_0. More specifically,

$$P(\mathcal{P}) = P(A_1) \cdot P(A_2) \tag{4.1}$$

The probability of A_1 is the ratio of the occurrence count of c_0 to the size of the universe, which is simply the sum of the occurrence counts of all the nodes in the schema network.

$$P(A_1) = \frac{|c_0|}{|Universe|} \tag{4.2}$$

We use the notation $c_i \frac{p_i}{d_i} c_{i+1}$ to refer to the transition from the node c_i to the node c_{i+1}



p_z
p_x
p_x
c_i
p_i
c_{i+1}
$(d_i = 1)$
p_y
$\frac{\left|(\leftarrow c_i)^{d_i},\,(\rightarrow c_{i+1})^{d_i},\,p_i\right|}{|c_i|}\cdot\gamma$

FIG. 4.2. Computing transition probability in a random walk.

in a random walk following the path $\mathcal{P}$. Then we have,

$$P(A_2) = \prod_{i=0}^{l-1} P(c_i \frac{p_i}{d_i} c_{i+1}) \tag{4.3}$$

Figure 4.2 illustrates how we compute the transition probability $P(c_i \frac{p_i}{d_i} c_{i+1})$. The weight of the edge $(c_i, p_i, c_{i+1})^{d_i}$ is $\left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i},\ p_i\right|$. The total occurrence counts of the node c_i is $|c_i|$. The standard transition probability is simply the ratio of the two. However, we extend the standard one by multiplying it with a parameter γ, a real number between 0 and 1. The purpose of introducing γ is to enable us to have control over the relative importance of short paths against long paths. Equation 4.4 shows our transition probability function.

$$P(c_i \frac{p_i}{d_i} c_{i+1}) = \begin{cases} \dfrac{\left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i},\ p_i\right|}{|c_i|} \cdot \gamma & \text{if } 0 < i \leqslant l-1 \\ \dfrac{\left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i},\ p_i\right|}{|c_i|} & \text{if } i = 0 \end{cases} \tag{4.4}$$

We do not apply γ when $i = 0$ because c_i is then the starting node of the path and γ is useless at this occasion. If we interpret this γ in the famous "drunken sailor's walk" paradigm, it means that the drunk can sleep at any node with a probability of $1 - \gamma$ after starting his walk.

From Equations 4.1, 4.2, 4.3 and 4.4, it follows that

$$\begin{aligned} P(\mathcal{P}) &= P(A_1) \cdot P(A_2) \\ &= \frac{|c_0|}{|Universe|} \cdot \prod_{i=0}^{l-1} P(c_i \frac{p_i}{d_i} c_{i+1}) \\ &= \gamma^{l-1} \cdot \frac{|c_0|}{|Universe|} \cdot \prod_{i=0}^{l-1} \frac{\left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i},\ p_i\right|}{|c_i|} \\ &= \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_i|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i},\ p_i\right| \end{aligned} \tag{4.5}$$

If $\gamma = 1$, then $P(\mathcal{P})$ is the same as the one computed from using the standard random walk. If $\gamma = 0$, only paths with a length of one have non-zero probability, which means only direct relations are considered in this case.

A path $\mathcal{P}$ and its return path $\mathcal{P}'$ represent the same relation. For example, $Director \overset{director}{\leftarrow} Film \overset{starring}{\rightarrow} Actor$ describes the same indirect relation as $Actor \overset{starring}{\leftarrow} Film \overset{director}{\rightarrow} Director$ does. Therefore, they should have the same probability to be "observed" in the schema network. Before we prove that $P(\mathcal{P}) = P(\mathcal{P}')$, we first formalize the definition of the return path $\mathcal{P}'$.

Definition 4.3.3. RETURN PATH. A schema path $\mathcal{P}' = (\mathbf{C}', \mathbf{P}', \mathbf{D}', l')$ is said to be the return path of $\mathcal{P} = (\mathbf{C}, \mathbf{P}, \mathbf{D}, l)$ if and only if (1) $l' = l$ and (2) $\forall i \in \{0..\,l\}\ \ c'_i = c_{l-i}$ and (3) $\forall i \in \{0..\,l-1\}\ \ p'_i = p_{l-1-i}$ and (4) $\forall i \in \{0..\,l-1\}\ \ d'_i = (-1) \cdot d_{l-1-i}$

Property 4.3.1. *Given a schema path* $\mathcal{P} = (\mathbf{C}, \mathbf{P}, \mathbf{D}, l)$ *and its return path* $\mathcal{P}' = (\mathbf{C}', \mathbf{P}', \mathbf{D}', l')$*, we have* $P(\mathcal{P}) = P(\mathcal{P}')$.

Proof.

$$\begin{aligned}
P(\mathcal{P}') &= \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{i=1}^{l-1} \frac{1}{|c'_i|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c'_i)^{d'_i},\ (\rightarrow c'_{i+1})^{d'_i},\ p'_i\right| \\
&= \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_{l-i}|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c_{l-i})^{(-1)\cdot d_{l-1-i}},\ (\rightarrow c_{l-i-1})^{(-1)\cdot d_{l-1-i}},\ p_{l-1-i}\right| \\
&= \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_{l-i}|} \cdot \prod_{i=0}^{l-1} \left|(\rightarrow c_{l-i})^{d_{l-i-1}},\ (\leftarrow c_{l-i-1})^{d_{l-i-1}},\ p_{l-i-1}\right| \\
&= \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_{l-i}|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c_{l-i-1})^{d_{l-i-1}},\ (\rightarrow c_{l-i})^{d_{l-i-1}},\ p_{l-i-1}\right|
\end{aligned}$$

Letting $j = l - i$ in the first $\prod$ and $j = l - i - 1$ in the second $\prod$,

$$\begin{aligned}
&= \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{j=1}^{l-1} \frac{1}{|c_j|} \cdot \prod_{j=0}^{l-1} \left|(\leftarrow c_j)^{d_j},\ (\rightarrow c_{j+1})^{d_j},\ p_j\right| \\
&= P(\mathcal{P})
\end{aligned}$$

□

By applying the random walk on a graph, a special case of a Markov chain, we actually make the *Markov property* assumption – the conditional probability of being in a future state depends only on the present state, not on all past states. Although this assumption does not hold for many real world problems, the Markov chain has been shown to be a useful simplification in practice, which is also true for our problem.

On the basis of schema path probability, we define schema path frequency $f(\mathcal{P})$ as

$$\begin{aligned} f(\mathcal{P}) &= P(\mathcal{P}) \cdot |Universe| \\ &= \gamma^{l-1} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_i|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i},\ p_i\right| \end{aligned} \tag{4.6}$$

When $l = 1$ we have $f(\mathcal{P}) = \left|(\leftarrow c_0)^{d_0},\ (\rightarrow c_1)^{d_0},\ p_0\right|$, which is simply the weight of the edge $(c_0, p_0, c_1)^{d_0}$.

Our schema path model is intended to store and index all the schema paths with a length no larger than a given threshold and their frequencies. The entire model will be held in memory for fast computation of our querying system. The only operation that the model needs to support at present is to return all the schema paths and their frequencies between two given classes. The length threshold is typically a small number, such as 2 or 3. One reason is that we expect a relation in a SFQ query would only be mapped to a short schema path because otherwise the user is expected to decompose the “long” relation into multiple relations in the SFQ query. The other reason is that if long paths are included, a very large number of paths will be produced for a knowledge base having many classes and properties (e.g. DBpedia). This will result in a very large model, which is difficult to store in memory.

4.3.2 Model Optimization

Even after applying the length threshold, large and diverse knowledge bases can give rise to too many schema paths to materialize in a model. For example, there is about one million paths between two classes *Person* and *Place* in DBpedia even when the length threshold is only two. On the other hand, dynamically finding all the schema paths between two given nodes from the schema network and compute their frequencies would be too slow. To tackle this problem, we develop an optimization technique that drastically reduces the required space of the model and still maintain fast computation.

First of all, for the schema network, we group all the edges with the same direction between two nodes into a single edge. We call the resulting network as *concept network*.

Definition 4.3.4. CONCEPT NETWORK. On the basis of the schema network $G_s = (C, R, P)$, the concept network is defined as a node-labeled, weighted and directed graph $G_c = (C, \ddot{R})$ with the same node set C as in G_s and the edge set $\ddot{R} \subseteq C \times C$ such that $(c, c') \in \ddot{R}$ if and only if $\exists p \in P,\ (c, p, c') \in R$. Each edge $(c, c') \in \ddot{R}$ has the weight of $|\leftarrow c, \rightarrow c'|$.

By analogy to schema paths on the schema network, we also have *concept paths* on the concept network. Two examples of concept paths are $Actor \leftarrow Film \rightarrow Director$ and $Author \leftarrow Paper \rightarrow Paper \rightarrow Author$. We formally define concept path as below.

Definition 4.3.5. CONCEPT PATH. A concept path $\ddot{\mathcal{P}} = (\mathbf{C}, \mathbf{D}, l)$ is defined as a path on the concept network $G_c = (C, \ddot{R})$, where (1) l denotes the length of the path; (2) $\mathbf{C}$ is a vector of length $l+1$ that represents the sequence of classes $\langle c_0, c_1, c_2, \cdots, c_l \rangle$ in the path; and (3) $\mathbf{D} \in \{-1, 1\}^l$ is a vector of directions $\langle d_0, d_1, d_2, \cdots, d_{l-1} \rangle$ such that for every $i \in \{0..\, l-1\},\ (c_i, c_{i+1})^{d_i} \in \ddot{R}$ represents an edge in the path.

Similarly, we define concept path probability $P(\ddot{\mathcal{P}})$ on the concept network. Following the same inference in the previous section, we can prove that

$$P(\ddot{\mathcal{P}}) = \frac{\gamma^{l-1}}{|Universe|} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_i|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i}\right| \tag{4.7}$$

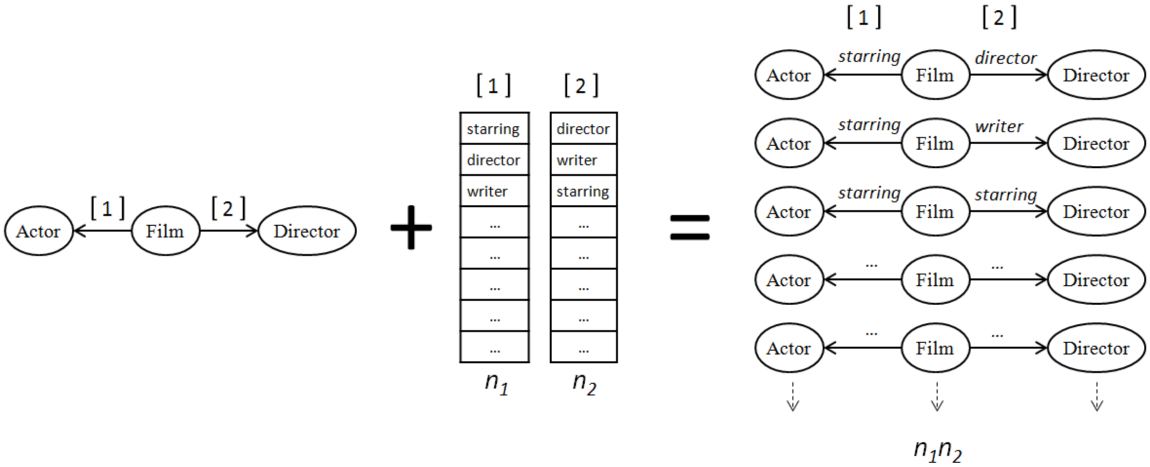


FIG. 4.3. schema path model optimization

We also define concept path frequency $f(\ddot{\mathcal{P}})$ on the basis of $P(\ddot{\mathcal{P}})$.

$$f(\ddot{\mathcal{P}}) = P(\ddot{\mathcal{P}}) \cdot |Universe| \\ = \gamma^{l-1} \cdot \prod_{i=1}^{l-1} \frac{1}{|c_i|} \cdot \prod_{i=0}^{l-1} \left|(\leftarrow c_i)^{d_i},\ (\rightarrow c_{i+1})^{d_i}\right| \tag{4.8}$$

Concept paths can be seen as templates or patterns for schema paths. Figure 4.3 gives an illustration. The concept path $Actor \overset{[1]}{\leftarrow} Film \overset{[2]}{\rightarrow} Director$ can be seen as a schema path template with two placeholders. On the other hand, as part of data structure of the schema network, we maintain arrays which store all the properties that can go between two directed classes. Hence, we can collect all the schema paths having the pattern of the concept path at almost no cost by filling the placeholders with all the combinations of the elements in their corresponding arrays. It follows that our schema path model only needs to store and index concept paths rather than schema paths themselves. This greatly reduces the space required by the model. As shown in the example, instead of storing $n_1 \cdot n_2$ schema paths, we only need store a single concept path.

We can also compute the schema path frequency $f(\mathcal{P})$ indirectly from the concept path frequency $f(\ddot{\mathcal{P}})$, as shown in Equation 4.9

$$f(\mathcal{P}) = f(\ddot{\mathcal{P}}) \prod_{i=0}^{l-1} P(p_i | (c_i \xrightarrow{*} c_{i+1})^{d_i}) \tag{4.9}$$

where $P(p_i | (c_i \xrightarrow{*} c_{i+1})^{d_i})$ represents the conditional probability that p_i appears between two given classes c_i and c_{i+1} with the direction d_i on the schema network.

4.4 Concept Association Knowledge Model

Knowing the strength of association between concepts and properties is an important kind of domain knowledge that is very useful for disambiguation. The term ‘Titanic’ in the query “Who are the actors of Titanic” could refer to a ship or a film, but the latter is more likely because films commonly have actors but other potential types (e.g., ship, book, game, place, album, etc.) do not. We know birds can fly but trees cannot, a database table is not kitchen table, etc. Such knowledge is essential for human language understanding. We refer to this as *Concept Association Knowledge* (CAK). Domain and range definitions for properties in ontologies, argument constraint definitions of predicates in logic systems and database schemata all belong to this knowledge. However, manually defining this knowledge for broad or open domains is tedious and expensive.

Instead, we learn concept association knowledge statistically from instance data and thus avoid the expensive human labor. However, instead of producing “tight” assertions such as those used in RDF property domain and range constraints, we generate the *degree of associations*. Classical logics that make either true or false assertions are less suited in an open-domain scenario, especially those created from heterogeneous data sources. For example, what is the range of the property *author* in DBpedia? Both *Writer* and *Artist* are not appropriate because the object of *author* could be something other than *Writer* or *Artist*, for example *Scientist*. Having *Person* as the range would be too general to be useful in disambiguation. Thus in our case there is no a fixed range for the property *author* but different classes do have varied association strengths with the property *author*.

We employ Pointwise Mutual Information (PMI) [17] to compute two types of associations: (i) *direct* association between classes and properties and (ii) *indirect* association between two classes. Equation 7.1 gives the PMI formula where f_{t_1} and f_{t_2} are the marginal occurrence counts of the two terms t_1 and t_2 and $f(t_1, t_2)$ is the co-occurrence count of t_1 and t_2 in the universe. N is a constant for the size of the universe.

$$\text{PMI}(t_1, t_2) \approx \log(\frac{f(t_1, t_2) \cdot N}{f_{t_1} \cdot f_{t_2}}) \tag{4.10}$$

Because the associations that we measure are direction-sensitive, we use directed classes in computing PMI. Hereafter, we use the notation $\vec{c}$ to generally refer to a directed class, regardless of what the direction is.

By applying Equation 4.10, the association between a class and a property is computed as below.

$$\text{PMI}(\vec{c},\ p) = log(\frac{|\vec{c},\ p| \cdot |Universe|}{|\vec{c}| \cdot |p|}) \tag{4.11}$$

We also need to measure association between two classes which may not be directly connected. This involves a problem about how to measure the “co-occurrence” count of two classes which are connected by paths. It is not clear what is the absolute co-occurrence count in this circumstance, but we can still develop some measures which reflect the relative co-occurrence degree of two classes. We achieve this by making use of concept path frequency $f(\ddot{\mathcal{P}})$. Let $\left\{\vec{c_x} \overset{*}{\rightsquigarrow} \vec{c_y}\right\}$ represents the set of concept paths that connect two directed classes $\vec{c_x}$ and $\vec{c_y}$. The indirect co-occurrence count between $\vec{c_x}$ and $\vec{c_y}$ is defined as the maximum path frequency of all the paths connecting them. More specifically,

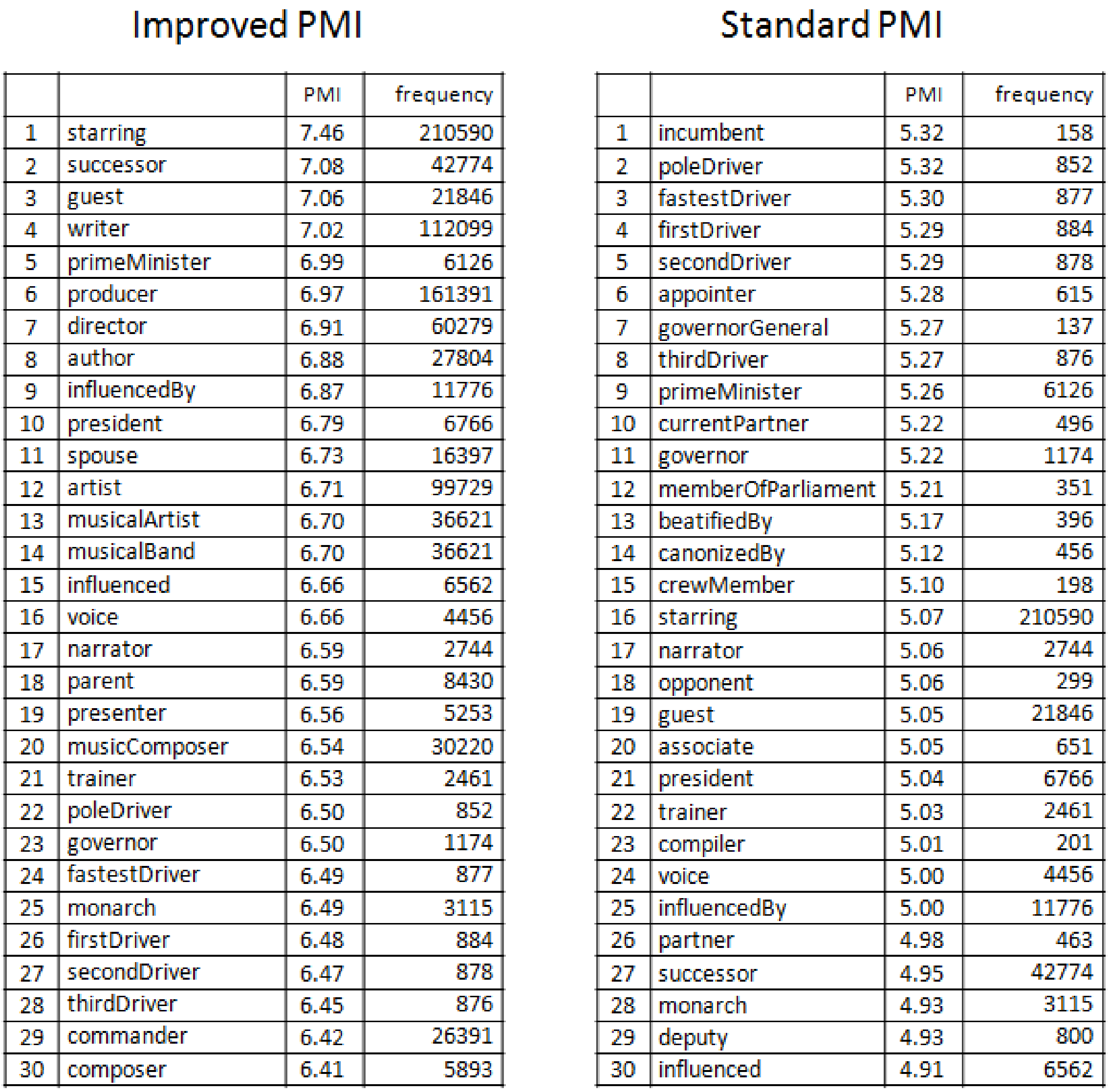
$$f(\vec{c_x}, \vec{c_y}) = \max_{\ddot{\mathcal{P}} \in \left\{\vec{c_x} \overset{*}{\rightsquigarrow} \vec{c_y}\right\}} f(\ddot{\mathcal{P}}) \tag{4.12}$$

There are other reasonable ways to define $f(\vec{c_x}, \vec{c_y})$. For example, instead of using the maximum path frequency, we can take the sum of all the path frequencies. Moreover, we can use schema path frequency rather than concept path frequency. We chose the current one because it is relatively easy to implement. Our initial experiments also show that the current measure works fairly well. We plan to experiment with other options in our future work.

Using Equation 4.10 and 4.12, the association between two directed classes is computed as the following.

$$\begin{aligned} \text{PMI}(\vec{c_x}, \vec{c_y}) &= log(\frac{f(\vec{c_x}, \vec{c_y}) \cdot |Universe|}{|\vec{c_x}\,| \cdot |\vec{c_y}|}) \\ &= log(\frac{\max_{\ddot{\mathcal{P}} \in \left\{\vec{c_x} \overset{*}{\rightsquigarrow} \vec{c_y}\right\}} f(\ddot{\mathcal{P}}) \cdot |Universe|}{|\vec{c_x}\,| \cdot |\vec{c_y}|}) \end{aligned} \tag{4.13}$$

PMI is a popular statistical association measure, being conceptually simple, efficient and effective. It also shows the best overall performance in measuring word association [96], compared with other measures including χ^2-test, likelihood ratio and average mu-



Improved PMI

		PMI	frequency
1	starring	7.46	210590
2	successor	7.08	42774
3	guest	7.06	21846
4	writer	7.02	112099
5	primeMinister	6.99	6126
6	producer	6.97	161391
7	director	6.91	60279
8	author	6.88	27804
9	influencedBy	6.87	11776
10	president	6.79	6766
11	spouse	6.73	16397
12	artist	6.71	99729
13	musicalArtist	6.70	36621
14	musicalBand	6.70	36621
15	influenced	6.66	6562
16	voice	6.66	4456
17	narrator	6.59	2744
18	parent	6.59	8430
19	presenter	6.56	5253
20	musicComposer	6.54	30220
21	trainer	6.53	2461
22	poleDriver	6.50	852
23	governor	6.50	1174
24	fastestDriver	6.49	877
25	monarch	6.49	3115
26	firstDriver	6.48	884
27	secondDriver	6.47	878
28	thirdDriver	6.45	876
29	commander	6.42	26391
30	composer	6.41	5893

Standard PMI

		PMI	frequency
1	incumbent	5.32	158
2	poleDriver	5.32	852
3	fastestDriver	5.30	877
4	firstDriver	5.29	884
5	secondDriver	5.29	878
6	appointer	5.28	615
7	governorGeneral	5.27	137
8	thirdDriver	5.27	876
9	primeMinister	5.26	6126
10	currentPartner	5.22	496
11	governor	5.22	1174
12	memberOfParliament	5.21	351
13	beatifiedBy	5.17	396
14	canonizedBy	5.12	456
15	crewMember	5.10	198
16	starring	5.07	210590
17	narrator	5.06	2744
18	opponent	5.06	299
19	guest	5.05	21846
20	associate	5.05	651
21	president	5.04	6766
22	trainer	5.03	2461
23	compiler	5.01	201
24	voice	5.00	4456
25	influencedBy	5.00	11776
26	partner	4.98	463
27	successor	4.95	42774
28	monarch	4.93	3115
29	deputy	4.93	800
30	influenced	4.91	6562

FIG. 4.4. Improved PMI vs. standard PMI in generating the most associated incoming properties of *Person* in DBpedia

tual information. However, PMI has a well-known problem of being biased towards low-frequency terms. To address this problem, we develop an improved PMI metric, described in Chapter 7, which offsets the bias of standard PMI. Figure 4.4 shows an example that compares the improved PMI with standard PMI in populating the top-30 list of the most associated properties with the directed class $\rightarrow Person$ in DBpedia. Although many of properties in the two lists overlap, standard PMI tends to favor low-frequency terms while the improved PMI does not exhibit this tendency. Hereafter, all the PMI values in this thesis are produced by our improved PMI metric. For simplicity, we will use the term PMI to refer to our modified PMI metric.

Figure 4.5 shows some examples that are taken from our DBpedia CAK model where the maximum path length is set to two and γ is 0.4. Examples 1 to 4 present, in order, top-25 outgoing and incoming properties that are most associated with two classes *Film* and *Actor*. Note that datatype properties are indicated by an initial "@" character to distinguish them from object properties. Examples 5 and 6 show the top-50 most associated classes to the class *Actor*. Classes that are not directly connected to *Actor* can also be strongly associated. For example, →*D̃irector* has a PMI value 6.1 with →*Actor*, which is ranked at 7th place.

In the first four examples, the top properties are very informative for telling the classes, such as *starring* and *director* for *Film*. Lower ranked properties tend to be less related to the classes. Example 4 shows that both *author* and *writer* can be incoming properties of *Actor*, but *writer* is more related. On the other hand, the first example shows that only *writer*, not *author*, can describe *Film*. In the DBpedia ontology, *author* and *writer* are used for different contexts with *writer* used for films. The class *Actor* has both *writer* and *author* as incoming properties because actors can write things other than films (e.g., books). Manually creating this level of ontological knowledge would take many hours of human labor, which we can save by using the CAK model. Noisy data in DBpedia can result in some abnormal associations, as shown in example 1, but their association strength is typically low.

For readers who are interested in seeing more examples, the entire CAK model learned from the DBLP+ dataset can be referenced in Appendix A.

1) ←**Film**: starring 7.6, musicComposer 7.4, cinematography 7.4, director 7.4, distributor 7.4, editing 7.3, writer 7.1, @budget 7.1, @gross 7.1, language 6.8, producer 6.5, @runtime 6.4, @releaseDate 6.3, narrator 6.2, country 5.0, @name 4.7, subsequentWork 4.2, previousWork 4.1, mediaType -1.3, @facilityId -1.3, @airDate -1.4, @frequency -1.6, @pages -1.6, license -1.6, author -1.7

2) →**Film**: academyAward 8.7, baftaAward 8.2, basedOn 7.7, previousWork 7.7, subsequentWork 7.7, filmFareAward 7.5, notableWork 7.5, award 6.8, emmyAward 6.0, album 5.8, knownFor 5.4, related 5.4, associatedAct 5.2, product 4.9, format 4.7, significantProject 4.6, genre 4.4, series 4.3, deathCause 3.9, tonyAward 3.9, derivative 3.8, resolution 3.7, associatedBand 3.7, associatedMusicalArtist 3.7, openingTheme 3.5

3) ←**Actor**: spouse 7.2, tonyAward 7.2, academyAward 7.1, @alias 7.0, occupation 7.0, emmyAward 7.0, birthPlace 6.8, ethnicity 6.8, baftaAward 6.7, @birthDate 6.6, @numberOfFilms 6.6, partner 6.6, filmFareAward 6.5, @activeYearsStartYear 6.2, deathPlace 6.2, associatedAct 6.0, @deathDate 6.0, @activeYearsEndYear 5.6, award 5.4, @name 5.2, homepage 4.8, @weight 4.6, @birthName 4.5, instrument 3.1, child 2.4

4) →**Actor**: starring 8.2, guest 7.9, director 7.5, voice 7.5, narrator 7.2, spouse 6.6, writer 6.6, presenter 6.5, executiveProducer 6.4, cinematography 6.4, editing 6.2, creator 6.2, producer 6.2, partner 6.1, openingTheme 6.0, endingTheme 5.9, musicComposer 5.9, showJudge 5.2, lyrics 5.1, composer 5.0, author 5.0, musicBy 4.7, child 4.4, relative 4.3, foundationPerson 4.2

5) ←**Actor**: →Âlias 7.0, →P̃rofession 6.6, →S̃pouse 6.6, →Musical 6.5, →F̃ield 6.4, →FilmFestival 6.2, →D̂ate 6.2, →M̃unicipality 6.0, →R̃esidence 5.9, →H̃eadquarters 5.9, →H̃ometown 5.9, →Õccupation 5.9, →C̃apital 5.9, →C̃ounty 5.9, →P̃rovince 5.9, →R̃egion 5.9, →Ãdvisor 5.9, →City 5.9, →G̃arrison 5.8, →Ãrea 5.8, →Ãssembly 5.8, →L̃ocation 5.8, →T̃raining 5.8, →H̃ighschool 5.8, →B̃uilder 5.8, →G̃round 5.8, →Ẽducation 5.8, →C̃ampus 5.8, →Ãlma mater 5.8, →Settlement 5.8, →P̃roduct 5.8, →S̃ubsidiary 5.7, →S̃ource 5.7, →C̃ause 5.7, →M̃outh 5.7, →S̃tate 5.7, →PopulatedPlace 5.7, →M̃anufacturer 5.7, →D̃esigner 5.7, →Place 5.7, →Award 5.7, →P̃artner 5.7, →S̃hrine 5.7, →P̃ublisher 5.7, →G̃enre 5.7, →Film 5.7, →C̃itizenship 5.6, →Ẽditor 5.6, →Ẽthnicity 5.6, →Ŷear 5.6

6) →**Actor**: ←Film 7.5, ←TelevisionShow 6.7, ←S̃eries 6.6, ←Work 6.4, ←TelevisionEpisode 6.4, →Actor 6.1, →D̃irector 6.1, →Ẽditing 6.0, →C̃inematography 6.0, →B̂udget 5.7, →Ñarrator 5.7, →Artist 5.6, →Person 5.5, →Film 5.4, ←Musical 5.4, →Language 5.3, →G̃uest 5.2, ←S̃pouse 5.2, →P̃roducer 5.1, ←Thing 5.1, →Work 5.1, →MusicalWork 5.1, →R̂untime 5.0, →Album 5.0, →R̂eview 4.9, →TelevisionEpisode 4.9, →MusicalArtist 4.9, →C̃omposer 4.9, →Broadcast 4.9, →AdultActor 4.8, →Single 4.8, →S̃pouse 4.8, ←P̃roduct 4.7, →M̃edia 4.7, →Company 4.7, →Îsbn 4.7, →Comedian 4.6, →MusicGenre 4.6, →RecordLabel 4.6, →C̃reator 4.6, →Ŝide 4.6, →P̃latform 4.6, ←Comedian 4.6, →Ôclc 4.6, →Ṽoice 4.6, →Ĩnput 4.6, →Band 4.5, ←Ñarrator 4.5, →F̃ormat 4.5, →M̃usic 4.5

FIG. 4.5. Six examples from DBpedia's CAK. The examples 1-4 show the most associated properties for four directed classes and the examples 5-6 show the the most associated classes for two directed classes.

Chapter 5

QUERY INTERPRETATION

In this chapter, we first formalize our problem and give an overall picture of our approach. Next, we describe the two phases in the mapping algorithm, namely, concept mapping and relation mapping.

5.1 Approach Outline

First of all, we give the formal definition of SFQ.

Definition 5.1.1. SFQ. A SFQ is defined as a node-and-edge labeled, directed graph $G_q = (\hat{E}, \hat{R})$ with the node set $\hat{E}$ and the edge set $\hat{R}$, where (1) $\hat{E}$ contains all the entities in the query and each entity $\hat{e} \in \hat{E}$ is described by two parts: its name or value or variable $\hat{e}.v$ and its concept $\hat{e}.c$; (2) $\hat{R}$ is the set of edges, in which $(\hat{e}, \hat{p}, \hat{e}')$ represents a relation directed from the subject entity $\hat{e}$ to the object entity $\hat{e}'$ with the predicate $\hat{p}$.

Next, we define SFQ interpretation on the schema network.

Definition 5.1.2. SFQ INTERPRETATION. An interpretation of a SFQ $G_q = (\hat{E}, \hat{R})$ on the schema network $G_s = (C, R, P)$ is a mapping σ such that $\forall\, \hat{e} \in \hat{E},\ \sigma(\hat{e}.c) \in C$ and $\forall\, (\hat{e}, \hat{p}, \hat{e}') \in \hat{R},\ \sigma((\hat{e}, \hat{p}, \hat{e}'))$ is a schema path $\mathcal{P}$ that starts with $\sigma(\hat{e}.c)$ and ends with $\sigma(\hat{e}'.c)$.

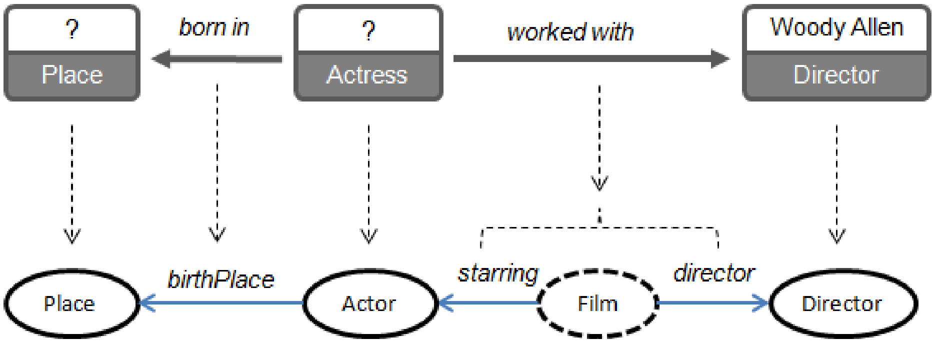


FIG. 5.1. A SFQ mapping example

The result of a mapping σ is also a graph and we denote it as G_σ.

We have two problems to solve in this chapter: (1) developing an objective function $\phi(\sigma)$ that measures the semantic similarity between G_q and G_σ and (2) searching in the space $\mathbb{M}$ of possible mappings for the ones that produce the highest $\phi(\sigma)$ scores.

In contrast to many graph matching approaches which directly search into the entity network, our approach deals with a much smaller network, the schema network. However, the mapping space $\mathbb{M}$ is still huge and it is intractable to go through each possible mapping for finding the optimal ones.

Instead of blindly trying all mapping possibilities, we can rely on lexical semantic similarity measures to find mapping candidates. Consider the example in Figure 5.1, which gives a SFQ and its best mapping in DBpedia. All the concepts in the SFQ are semantically similar to their corresponding classes. This allows us to generate class candidates for each concept in the SFQ by exploiting lexical semantic similarity. However, we cannot successfully generate candidates for all the properties because some of them need to be mapped to paths, for example, “worked with”. Determining how well a path is mapped by a SFQ relation requires knowing, in advance, the classes to which the concepts connected by the relation are mapped. In other words, the computation is context-dependent.

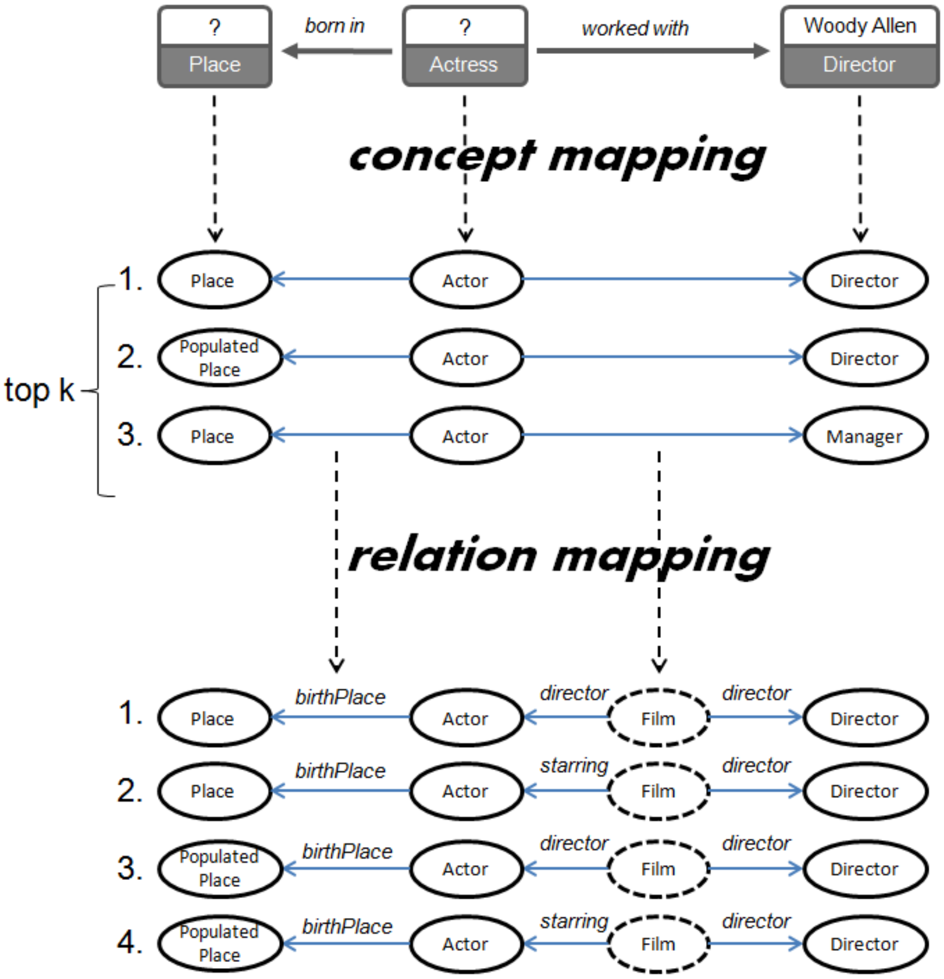


FIG. 5.2. two-phase mapping algorithm

One way to deal with this is to iterate through all combinations of the class candidates and find the best mappings for the relations for each combination. However, this approach is very costly because mapping SFQ relations to paths is a computationally expensive process and we cannot afford to iterate the process many times.

We chose to use local information at hand to jointly disambiguate and resolve the SFQ concepts first, in spite of missing information on indirect relations. We call this step as *concept mapping*. After the mappings of the concepts are known, we then disambiguate and resolve the relations. We refer to the second step as *relation mapping*. An illustration of the two-phase mapping procedure is given in Figure 5.2. Rather than just producing the top interpretation of the SFQ concepts, we generate the top-k interpretations as the result of concept mapping. We do so because our concept mapping algorithm is not perfect and the highest ranked interpretation is not always correct. The efficacy of the concept mapping algorithm determines the selecting of the k value. The more effective the concept mapping algorithm is, the smaller k we can choose. The k value in turn has great impact on the efficiency of the relation mapping phase because the relation mapping algorithm will be applied to each of the k concept mapping hypotheses.

In the rest of this chapter, interpreting the SFQ in Figure 5.1 on the schema network of DBpedia will be used as the running example to illustrate the two-phase mapping algorithm.

5.2 Phase 1: Concept Mapping

5.2.1 Generating Candidates via Lexical Semantic Similarity

We develop two different lexical semantic similarity models for generating candidates because we find that semantic similarity for concepts differs from that for relations. For example, “doctor” and “patient” are different concepts, but they can work as the same relation. Some other examples include “parent” vs. “child”, “wife” vs. “husband”, “book” vs. “writer”, and “wife” vs. “marry”. To distinguish them, we coin two terms *concept similarity* and *relation similarity*[1]. We refer to the two models as concept similarity model and relation similarity model.

The candidate lists for SFQ concepts and relations have the size of k_1 and k_2, respectively. For each concept or relation in a SFQ, we populate its candidate list with the most

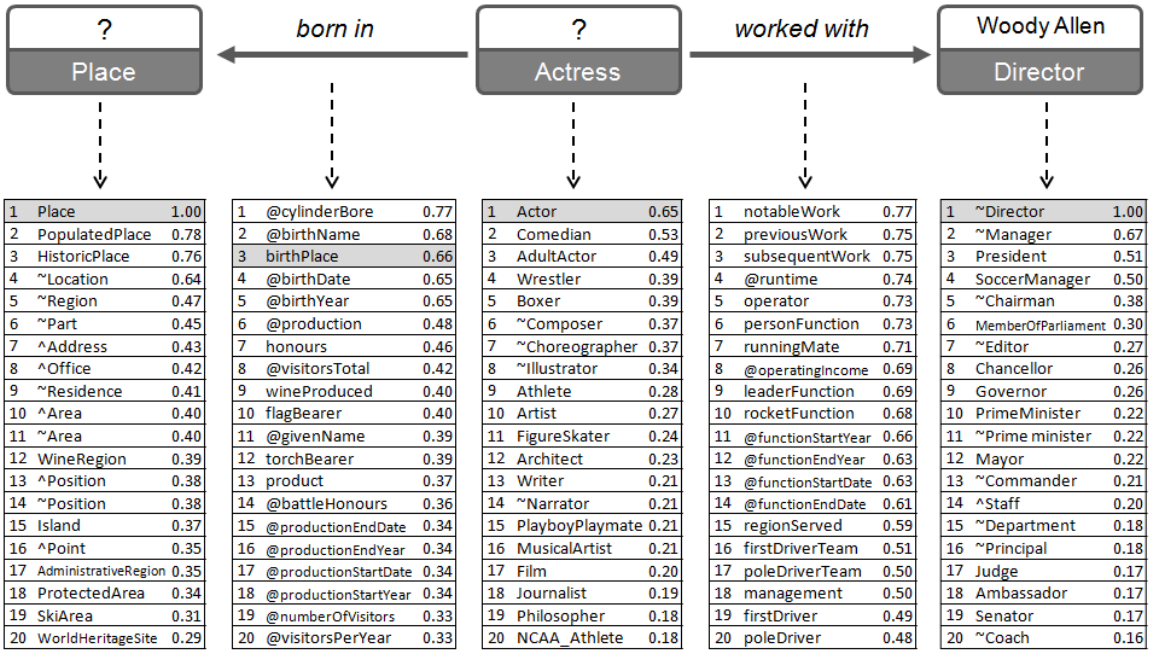
[1] Relation similarity is very close to the notion of *semantic relatedness*.

semantically similar ontology classes or properties[2] (See Chapter 8 for semantic similarity computation). Minimum similarity thresholds, 0.075 and 0.05, are experimentally applied to concepts and relations, respectively, for guaranteeing that all the terms have at least some similarity. For a *default relation*, we generate the $\frac{3}{4}k_2$ ontology properties most semantically similar to each of its connected concepts because the semantics of a *default relation* is often conveyed in one of its connected concepts. Then we assemble these into a list of $\frac{3}{2}k_2$ ontology properties. The values for k_1 and k_2 are the results of the compromise between mapping performance and computation time, which depend on the degree of heterogeneity in the underlying ontologies and the quality of the semantic similarity measure. For a dataset as diverse as DBpedia, we experimentally found that setting $k_1 = 10$ and $k_2 = 20$ is sufficient to produce good results.

Figure 5.3 shows the candidate lists generated from our DBpedia ontology for the five user terms in the SFQ query, with candidates ranked by their similarity score. Both k_1 and k_2 are set to 20. Classes starting with $\wedge$ are attribute types and starting with $\sim$ are inferred classes (See Section 4.1 for their definitions). We use the Stanford part of speech (POS) tagger and morphology package [99] to get word lemmas with their POS and then compute their semantic similarity. While our similarity measure is effective and works well, it is not perfect. For example, “born in” is mistaken as highly similar to “@cylinderBore”. It is worth mentioning that we manually increased the similarity score between “Actress” and “Film” from 0.11 to 0.20 for the purpose of creating an interesting running example that requires joint disambiguation.

We are developing techniques to deal with the case when entities are only described by their names, without being given their concepts. For example, there could be no “Director” associated with “Woody Allen” in the SFQ in Figure 5.3. Our solution is to find

[2] The candidate properties of a relation are semantically similar to the predicate of the relation only.



1	Place	1.00	1	@cylinderBore	0.77	1	Actor	0.65	1	notableWork	0.77	1	~Director	1.00
2	PopulatedPlace	0.78	2	@birthName	0.68	2	Comedian	0.53	2	previousWork	0.75	2	~Manager	0.67
3	HistoricPlace	0.76	3	birthPlace	0.66	3	AdultActor	0.49	3	subsequentWork	0.75	3	President	0.51
4	~Location	0.64	4	@birthDate	0.65	4	Wrestler	0.39	4	@runtime	0.74	4	SoccerManager	0.50
5	~Region	0.47	5	@birthYear	0.65	5	Boxer	0.39	5	operator	0.73	5	~Chairman	0.38
6	~Part	0.45	6	@production	0.48	6	~Composer	0.37	6	personFunction	0.73	6	MemberOfParliament	0.30
7	^Address	0.43	7	honours	0.46	7	~Choreographer	0.37	7	runningMate	0.71	7	~Editor	0.27
8	^Office	0.42	8	@visitorsTotal	0.42	8	~Illustrator	0.34	8	@operatingIncome	0.69	8	Chancellor	0.26
9	~Residence	0.41	9	wineProduced	0.40	9	Athlete	0.28	9	leaderFunction	0.69	9	Governor	0.26
10	^Area	0.40	10	flagBearer	0.40	10	Artist	0.27	10	rocketFunction	0.68	10	PrimeMinister	0.22
11	~Area	0.40	11	@givenName	0.39	11	FigureSkater	0.24	11	@functionStartYear	0.66	11	~Prime minister	0.22
12	WineRegion	0.39	12	torchBearer	0.39	12	Architect	0.23	12	@functionEndYear	0.63	12	Mayor	0.22
13	^Position	0.38	13	product	0.37	13	Writer	0.21	13	@functionStartDate	0.63	13	~Commander	0.21
14	~Position	0.38	14	@battleHonours	0.36	14	~Narrator	0.21	14	@functionEndDate	0.61	14	^Staff	0.20
15	Island	0.37	15	@productionEndDate	0.34	15	PlayboyPlaymate	0.21	15	regionServed	0.59	15	~Department	0.18
16	^Point	0.35	16	@productionEndYear	0.34	16	MusicalArtist	0.21	16	firstDriverTeam	0.51	16	~Principal	0.18
17	AdministrativeRegion	0.35	17	@productionStartDate	0.34	17	Film	0.20	17	poleDriverTeam	0.50	17	Judge	0.17
18	ProtectedArea	0.34	18	@productionStartYear	0.34	18	Journalist	0.19	18	management	0.50	18	Ambassador	0.17
19	SkiArea	0.31	19	@numberOfVisitors	0.33	19	Philosopher	0.18	19	firstDriver	0.49	19	Senator	0.17
20	WorldHeritageSite	0.29	20	@visitorsPerYear	0.33	20	NCAA_Athlete	0.18	20	poleDriver	0.48	20	~Coach	0.16

FIG. 5.3. A ranked list of terms from our DBpedia ontology is generated for each term in the SFQ, “Which actresses worked with Woody Allen and where were they born?”.

all the entities that lexically match “Woody Allen” and compute their string similarities with “Woody Allen”. Then we collect all the types of the entities, which might include *Director*, *Person*, *Book* and etc., and put them into the candidate list of Woody Allen along with their string similarities. Our disambiguation algorithm, which is discussed next, can automatically locate the most appropriate type for the entity Woody Allen.

Our DBpedia ontology consists of two sets, namely the class set C and the property set P in the DBpedia’s schema network $G_s = (C, R, P)$. The cardinality of C and P are 523 and $1,099$ respectively. Generating candidates requires comparing each concept or relation in a given SFQ $G_q = (\hat{E}, \hat{R})$ to every class or property in the ontology, resulting in a running time of $\Theta(\hat{E}C + \hat{R}P)$. In fact, we extend this to the process of computing semantic similarity between every user term in a given SFQ and every ontology term and further cache the values in a hash map for future access. Our semantic similarity computation is very efficient. The whole process took only 27 milliseconds per question by average for the set of questions in Table 9.11.

5.2.2 Disambiguation via Optimization

In Figure 5.3, each combination of ontology terms, with one term coming from each candidate list, is a possible mapping or “mapping hypothesis”, but some are reasonable and others not. Disambiguation here means choosing the most reasonable hypotheses from all possible combinations. Our disambiguation problem differs from traditional ones (e.g. word-sense disambiguation) in that we do not have predefined senses or categories. In contrast, each term is associated with a similarity value that indicates how likely it is to be the right term by itself. The values are automatically produced and therefore require no human input or involvement.

An intuitive measure of reasonableness for a given hypothesis is the degree to which its ontology terms associate in the way that their corresponding user terms connect in the SFQ. For example, since “Actress” and “Director” are connected in the SFQ in Figure 5.3, we can expect that their corresponding classes should have good pairwise statistical association. Therefore, according to our CAK model we can know that *Actor* and *D̃irector* makes a more reasonable combination than that of *Actor* and *President*. For another instance, since “Place” is connected by “born in”, their corresponding ontology terms should also have good statistical association. Thus the combination of *Place* and *birthPlace* makes much more sense than that of *Place* and *@cylinderBore* or that of *Place* and *@birthName* because the CAK model tells us that a strong association holds between *Place* and *birthPlace* but not *@cylinderBore* or *@birthName*.

We use two types of connections in a SFQ for computing the overall association of a hypothesis: connections between concepts and their relations (e.g., “Actress” and “born in”) and between two connected concepts (e.g., “Actress” and “Director”). We exclude indirect connections (e.g., between “Place” and “worked with”) because they do not necessarily entail good associations. This distinguishes from the coarse-grained disambiguation methods [113] where context is simply a bag of words without compositional structure.

While some of the SFQ relations (e.g. “worked with”) need to be mapped to indirect relations, the others are still mapped to direct relations (e.g. “born in”). Hence, the association degrees between class candidates and property candidates are still helpful in better disambiguating the query, which we should not ignore.

The association degree of ontology terms in a hypothesis is just one of the two elements we use to evaluate the fitness of the hypothesis. The other element is, of course, lexical semantic similarity between the user terms and their corresponding ontology terms in the hypothesis. We need combine the two to constitute the fitness function in the concept mapping phase.

We start formalizing our approach. Suppose the query graph $G_q = (\hat{E}, \hat{R})$ has n nodes and m edges. Each concept or relation x_i in G_q has a corresponding set of candidate ontology terms Y_i. Our hypothesis space H is the Cartesian product over the sets Y_1, ..., Y_{m+n}.

$$H = Y_1 \times ... \times Y_{m+n} = \{(y_1, ..., y_{m+n}) : y_i \in Y_i\}$$

Each hypothesis $h \in H$ also describes a function $h(x)$ that maps x_i to y_i for $i \in \{1, ..., m+n\}$.

Let us define a fitness function $\varphi(h, G)$ that returns the fitness score of a hypothesis h on a query graph or subgraph G. We seek the hypothesis $h^* \in H$ that maximizes the fitness on the query graph G_q, which is computed as the summation of the fitness on each relation $\hat{r}_i = (\hat{e}_i, \hat{p}_i, \hat{e}'_i) \in \hat{R}$, i from 1 to m. More specifically,

$$h^* = \underset{h \in H}{argmax}\, \varphi(h, G_q) \tag{5.1}$$

$$\doteq \underset{h \in H}{argmax} \sum_{i=1}^{m} \varphi(h, \hat{r}_i) \tag{5.2}$$

Formula 5.2 achieves *joint disambiguation* because the joint concepts of different relations should be mapped to the same ontology class.

Let us denote $\hat{e}_i.c$ as $\hat{c}_i$ and $\hat{e}'_i.c$ as $\hat{c}'_i$. We make $\varphi(h, \hat{r}_i)$ be composed of three terms T_1 and T_2 and T_3 in Equation 5.3.

$$\begin{aligned} T_1 &= \mathrm{PMI}(\overrightarrow{h(\hat{c}_i)},\ \overrightarrow{h(\hat{c}'_i)}) \cdot \mathrm{sim_c}(\hat{c}_i,\ h(\hat{c}_i)) \cdot \mathrm{sim_c}(\hat{c}'_i,\ h(\hat{c}'_i)) \\ T_2 &= \mathrm{PMI}(\overrightarrow{h(\hat{c}_i)},\ h(\hat{p}_i)) \cdot \mathrm{sim_c}(\hat{c}_i,\ h(\hat{c}_i)) \cdot \mathrm{sim_r}(\hat{p}_i,\ h(\hat{p}_i)) \\ T_3 &= \mathrm{PMI}(\overrightarrow{h(\hat{c}'_i)},\ h(\hat{p}_i)) \cdot \mathrm{sim_c}(\hat{c}'_i,\ h(\hat{c}'_i)) \cdot \mathrm{sim_r}(\hat{p}_i,\ h(\hat{p}_i)) \end{aligned} \tag{5.3}$$

We use directed classes $\overrightarrow{h(\hat{c}_i)}$ and $\overrightarrow{h(\hat{c}'_i)}$ because we need consider direction in computing associations. Note that $\mathrm{sim_c}$ represents *concept similarity* and $\mathrm{sim_r}$ represents *relation similarity*. Each T_i is the product of the PMI association between two ontology terms in the hypothesis h, which are corresponding to two of the three elements $\hat{c}_i$, $\hat{p}_i$ and $\hat{c}'_i$ in the relation $\hat{r}_i$, and the semantic similarities between the two ontology terms and their corresponding user terms. Recall that $\mathrm{PMI}(\vec{c_x}, \vec{c_y})$ can measure association between two classes that are not directly connected.

Finally, the formula of $\varphi(h, \hat{r}_i)$ is shown in Equation 5.4. Since $\overrightarrow{h(\hat{c}_i)}$ and $\overrightarrow{h(\hat{c}'_i)}$ have two different directions each, there are totally four possible combinations of them. For every combination, we calculate $2T_1 + \max(T_2 + T_3,\, 0)$. Then, we take the maximum value of the results and use it as the output of $\varphi(h, \hat{r}_i)$. The max operator on $T_2 + T_3$ and

0 is used to impose a lower bound on $T_2 + T_3$ because $T_2 + T_3$ can easily go negative and even $-\infty$ when $\hat{r}_i$ needs to be mapped to an indirect relation. The lower bound of zero makes $\varphi(h, \hat{r}_i)$ be solely determined by $2T_1$ under such circumstances. We use a weight of two for T_1 for balance's purpose because there are two terms regarding to the connections between class and property. Moreover, the higher weight for T_1 than for T_2 and T_3 helps in the situations where the correct mapping of $\hat{p}_i$ is not in its candidate list. The higher weight gives us a better chance to map the concepts to the corresponding classes through T_1 when T_2 and T_3 fail.

(5.4)
$$\varphi(h, \hat{r}_i) = \max_{\substack{\overrightarrow{h(\hat{c}_i)} \in \{\leftarrow h(\hat{c}_i),\, \rightarrow h(\hat{c}_i)\} \\ \overrightarrow{h(\hat{c}'_i)} \in \{\leftarrow h(\hat{c}'_i),\, \rightarrow h(\hat{c}'_i)\}}} 2T_1 + \max(T_2 + T_3,\, 0)$$

We use the example in Figure 5.3 to illustrate the concept mapping algorithm described above. The SFQ in question has two relations. We refer to the relation $\frac{?}{Actress} \overset{born\,in}{\longrightarrow} \frac{?}{Place}$ as $\hat{r}_1$ and $\frac{?}{Actress} \overset{worked\,with}{\longrightarrow} \frac{Woody\,Allen}{Director}$ as $\hat{r}_2$. The hypothesis h^* that maximizes $\varphi(h, G_q)$ consists of all the highlight terms, namely (*Place*, *birthPlace*, *Actor*, ∅, *D̃irector*), which contains the correct concept mapping (*Place*, *Actor*, *D̃irector*). $\varphi(h^*, \hat{r}_1)$ has a value of 14.9, which is the sum of 7.4 obtained from $2T_1$, 2.9 from T_2 and 4.6 from T_3. On the other hand, $\varphi(h^*, \hat{r}_2)$ has a value of 7.92, which is contributed solely by $2T_1$. No property in the candidate list of "worked with" can make $T_2 + T_3$ positive when $h(\hat{c}_i)$ is *Actor* and $h(\hat{c}'_i)$ is *D̃irector*. Although h^* maximize $\varphi(h, G_q)$, it does not maximize $\varphi(h, \hat{r}_2)$. The hypothesis (*, *, *Film*, *notableWork*, *D̃irector*), referred to as h', achieves a higher $\varphi(h, \hat{r}_2)$ score, 8.3, which is the sum of 2.6 gained from $2T_1$, 1.2 from T_2 and 4.5 from T_3. However, h' has a low score on $\varphi(h, \hat{r}_1)$ because *Film* does not associate well with the candidates of "born in" and "Place", making h' inferior to h^*.

In the above example the correct concept mapping simply consists of all the top-one concept candidates. However, in general, the candidate with the highest semantic similarity score may not be the correct one. For example, consider the SFQ $\frac{?}{Soccer\,Club} \overset{has}{\longrightarrow} \frac{?}{Director}$. This time, *D̃irector* is no longer the correct mapping for "Director" because *D̃irector* is used for films but not soccer clubs in DBpedia. The most appropriate mapping in this context should be *SoccerManager* or *M̃anager*.

The algorithm of finding h^* can be easily extend to finding the top-k hypotheses. In the process of searching for h^*, we store the maximum value of $\varphi(h, G_q)$ we have found in a single variable and every time we compare the newly computed value of $\varphi(h, G_q)$ with that variable. When finding the top-k hypotheses, we use a priority queue with a fixed size of k, instead of a single variable, to store the top-k largest $\varphi(h, G_q)$ values. Every time, we compare the head of the queue, the smallest of the k values, to the newly computed value of $\varphi(h, G_q)$. If the head is smaller than the new value, we remove the head and insert the new value to the priority queue. The worst running time for completing a single operation increases from $O(1)$ to $O(\log k)$.

Time Complexity. Let k_1 and k_2 be the length of the candidate lists of SFQ concepts and relations, respectively. Let k_3 be the number of the top hypotheses we need find. Suppose the query graph $G_q = (\hat{E}, \hat{R})$ has n nodes and m edges. The complexity of a straightforward optimization algorithm is $k_1^n \cdot k_2^m \left[O(m) + O(\log k_3)\right]$ because (1) the hypothesis space H has a size of $k_1^n \cdot k_2^m$; (2) computing $\varphi(h, G_q)$ requires computing $\varphi(h, \hat{r}_i)$ on m relations; (3) comparing $\varphi(h, G_q)$ with and modify the priority queue of top k_3 hypotheses requires $O(\log k_3)$. We can significantly reduce this complexity by exploiting locality. The optimal mapping choice of a property can be determined locally when the two classes it links are fixed. So, we only iterate on all k_1^n combinations of classes. Moreover, we can iterate in a way such that the next combination differs from current combination only on one class with others remaining unchanged. This means we need only re-compute the links involving the changed class. The average number of links in which a class participates is $\frac{2m}{n}$. On the other hand, finding the property that maximizes the fitness of a link requires going through all k_2 choices in the candidate list, resulting in $O(k_2)$ running time. Put them together, the total computational complexity is reduced to $k_1^n \left[O(\frac{2m}{n}k_2) + O(\log k_3)\right]$.

Although the running time is still exponential in the number of concepts in G_q, it is not a serious issue in practical applications for three reasons. First, we expect that short queries with a small number of entities will dominate. Second, since we can do a much better job in measuring *concept similarity* than *relation similarity*, a small k_1 can be used for producing candidates of concepts and a relatively large k_2 for relations. Third, we can achieve further improvement by decomposing the graph into subgraphs and/or exploiting parallel computing.

We refer to the set of top-k hypotheses as H^*. The output of the concept mapping phase is the projection of H^* on the classes. The fitness scores of the hypotheses are discarded and will not be carried to the next phase.

5.3 Phase 2: Relation Mapping

5.3.1 Objective Function and Maximization

We first define the mapping space $\mathbb{S}$, in which we will search for the top-10 σs of the SFQ $G_q = (\hat{E}, \hat{R})$ that maximize the objective function $\phi(\sigma, G_q)$. Let $\{c_x \overset{*}{\rightsquigarrow} c_y\}$ represent the set of path starting with c_x and ending with c_y. We have

$$\mathbb{S} = \{\sigma \in \mathbb{M} \mid \exists h \in H^* \ (\forall \hat{e} \in \hat{E} \ \sigma(\hat{e}.c) = h(\hat{e}.c)) \wedge \\ (\forall (\hat{e}, \hat{p}, \hat{e}') \in \hat{R} \ \exists \mathcal{P} \in \{h(\hat{e}.c) \overset{*}{\rightsquigarrow} h(\hat{e}'.c)\} \ \mathcal{P}.l \leqslant L \wedge \sigma((\hat{e}, \hat{p}, \hat{e}')) = \mathcal{P})\}$$

where $\mathbb{M}$ represents the entire mapping space of G_q and L denotes the path length threshold.

The fitness score of σ on the query graph G_q is computed as the geometric mean of the fitness scores of σ on each relation $\hat{r}_i \in \hat{R}$. More specifically,

$$\phi(\sigma, G_q) = \sqrt[m]{\prod_{i=1}^{m} \phi(\sigma, \hat{r}_i)} \tag{5.5}$$

Analogous to the fitness function $\varphi(h, G_q)$ in the concept mapping phase, the σ that maximizes the $\phi(\sigma, G_q)$ does not necessarily maximize every $\phi(\sigma, \hat{r}_i)$.

Our searching problem for top-10 interpretations can be reduced to the problem of seeking the mapping $\sigma^* \in \mathbb{S}$ that maximizes $\phi(\sigma, G_q)$. More specifically,

$$\sigma^* = \underset{\sigma \in \mathbb{S}}{argmax}\ \phi(\sigma, G_q) \tag{5.6}$$

$$= \underset{\sigma \in \mathbb{S}}{argmax} \left(\prod_{i=1}^{m} \phi(\sigma, \hat{r}_i) \right)^{1/m} \tag{5.7}$$

Next, we start to describe the approach to compute $\phi(\sigma, \hat{r}_i)$.

5.3.2 Computing Fitness of Mapping a Relation to a Path

We combine two features to calculate $\phi(\sigma, \hat{r})$. One feature is the joint lexical semantic similarity between the SFQ relation $\hat{r}$ and the path to which it maps, $\sigma(\hat{r})$, which we also denotes as $\mathcal{P}$. The other feature is the schema path frequency of $\mathcal{P}$, $f(\mathcal{P})$. More specifically,

$$\phi(\sigma, \hat{r}) = Sim_{\bowtie}(\hat{r}, \mathcal{P}) \cdot (log(f(\mathcal{P})) + \beta) \tag{5.8}$$

where $Sim_{\bowtie}(\hat{r}, \mathcal{P})$ measures the joint lexical semantic similarity between $\hat{r}$ and $\mathcal{P}$; the term $log(f(\mathcal{P}))$ measures how much sense the path makes by itself; and β is a parameter to weight the relative importance between the two. The bigger β is, the less important role $f(\mathcal{P})$ plays.

To compute $Sim_{\bowtie}(\hat{r}, \mathcal{P})$, we need align terms appearing in $\hat{r} = (\hat{c}, \hat{p}, \hat{c'})$ and $\mathcal{P} = (\mathbf{C}, \mathbf{P}, \mathbf{D}, l)$. The two concepts $\hat{c}$ and $\hat{c'}$ in $\hat{r}$ are already aligned, as defined by σ, to the two ending classes of $\mathcal{P}$, c_0 and c_l, respectively. The only terms that we need to align are the predicate $\hat{p}$ in $\hat{r}$ and the properties $p_0, p_1, p_2, \cdots, p_{l-1}$ in $\mathcal{P}$.

We do not need to align $c_1, c_2, \cdots, c_{l-1}$ in $\mathcal{P}$ because we exclude them from the computation of $Sim_{\bowtie}(\hat{r}, \mathcal{P})$. There are several reasons why we ignore the intermediate classes. First, the semantics in $c_1, c_2, \cdots, c_{l-1}$ is often largely overlapped with that in $p_0, p_1, p_2, \cdots, p_{l-1}$. Second, because we apply product to join the similarity scores of the alignments, the less terms we include in the product, the less likely false negative errors can occur[3]. Third, the predicate $\hat{p}$, the only unaligned term in $\hat{r}$, is more likely to be semantically similar to the properties than the intermediate classes in $\mathcal{P}$. Fourth, some class terms do not have much specific semantics (e.g. *Thing*).

However, there is an exception in which we actually use some of the intermediate classes in computing $Sim_{\bowtie}(\hat{r}, \mathcal{P})$. This is when some of the properties $p_0, p_1, p_2, \cdots, p_{l-1}$ in $\mathcal{P}$ are *default relations* (e.g. *has*). In this case, we replace the "meaningless" properties with the (object) classes that the properties point to.

Since we ignore the intermediate classes in computing $Sim_{\bowtie}(\hat{r}, \mathcal{P})$, the schema paths that vary only on the intermediate classes have the same joint lexical semantic similarity with $\hat{r}$. However, their fitness scores with $\hat{r}$ are still different due to their varied schema path frequencies. In other words, the paths that have higher probability to be observed in

[3] A big deficiency of our lexical semantic similarity models is that sometimes they fail to find enough similarity between semantically similar terms, which we refer to as false negative errors

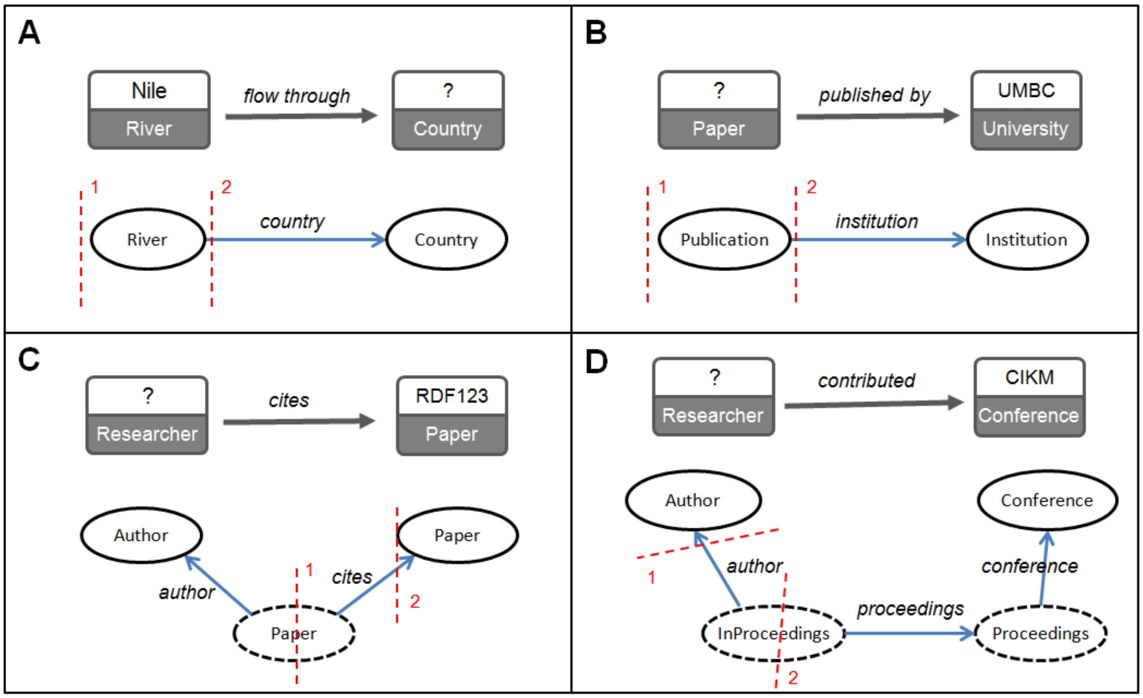


FIG. 5.4. Examples of heterogeneous alignments

the schema network will be ranked higher.

Due to the *semantic stretch* phenomenon we mentioned in Chapter 3, we cannot simply align the predicate $\hat{p}$ to the properties $p_0, p_1, p_2, \cdots, p_{l-1}$. In other words, the alignments are not homogeneous. Consider examples in Figure 5.4. In example A and B, neither is "flow through" semantically similar to *country*, nor is "published by" similar to *institution*. According to their meanings, the predicate "flow through" should be aligned to the class *River*, and the predicate "published by" to the class *Publication*.

By way of analogy, we might think semantics as something that can be stretched or is able to spread. The same (or close) semantics can stay within a single term or span over multiple continuous terms. For an instance, in example B the meaning of the class *Publication* in the schema path $\mathcal{P}$ spans over two terms, "Paper" and "published by", in the relation $\hat{r}$. For another instance, in example D the meaning of "Conference" in $\hat{r}$ stretches over three terms in $\mathcal{P}$, including *proceedings*, *conference* and *Conference*[4].

No matter how internal semantics stretches, if the path $\mathcal{P}$ has the same semantics as the relation $\hat{r}$, we can still cut the path into three parts that are corresponding to subject, predicate and object of the relation $\hat{r}$. The two red lines in each example in Figure 5.4 illustrate how the cutting should be done. The area to the left of red line 1 is called *subject region*, the area between red line 1 and 2 called *predicate region* and the area to the right of red line 2 called *object region*. The subject or object region sometimes can be empty, as shown in example A and B. This occurs when one of the ending classes of $\mathcal{P}$ is included in the predicate region. Although we could interpret it as the subject or object region being collapsed into the predicate region, this cutting behavior is exactly what we expect.

Each cutting defines a function. We refer to both the cutting and the function as ω. Let X represent the set $\{c_0, p_0, p_1, p_2, \cdot\cdot\cdot, p_{l-1}, c_l\}$ (i.e. all the terms in the path $\mathcal{P}$ except the intermediate classes). Let Y represent the set $\{\hat{c}, \hat{p}, \hat{c}'\}$ (i.e. all the schema terms in the relation $\hat{r}$). The cutting function ω maps from X to Y. More specifically,

$$\omega : X \rightarrow Y \tag{5.9}$$

where terms in the subject, predicate and object region are mapped to $\hat{c}$, $\hat{p}$ and $\hat{c}'$, respectively.

Based on the function ω, we can create alignments for the properties $p_0, p_1, p_2, \cdot\cdot\cdot, p_{l-1}$ and the predicate $\hat{p}$. Each $p_i \in \mathbf{P}$ is simply paired to $\omega(p_i)$. Let Z denote the set $\{\omega(p_i) \mid p_i \in \mathbf{P}\}$. If $\hat{p} \in Z$, then $\hat{p}$ has already been paired by one or more p_i; otherwise, we need pair $\hat{p}$ to $\omega^{-1}(\hat{p})$[5], which is either c_0 or c_l.

[4] Remind that we ignore intermediate class terms in the path $\mathcal{P}$

[5] Although ω is generally not invertible, it is invertible at this particular point.

The joint lexical semantic similarity with respect to the predicate $\hat{p}$ and the properties $p_0, p_1, p_2, \cdots, p_{l-1}$, which we refer to as $Sim_{\bowtie}(\hat{p}, \mathbf{P})$, can be computed using Equation 5.10.

(5.10)

$$Sim_{\bowtie}(\hat{p}, \mathbf{P}) = \begin{cases} \prod_{i=0}^{l-1} \mathrm{sim_r}(\omega(p_i), p_i) & \text{if } \hat{p} \in \{\omega(p_i) \mid p_i \in \mathbf{P}\} \\ (\prod_{i=0}^{l-1} \mathrm{sim_r}(\omega(p_i), p_i)) \cdot \mathrm{sim_r}(\hat{p}, \omega^{-1}(\hat{p})) & \text{if } \hat{p} \notin \{\omega(p_i) \mid p_i \in \mathbf{P}\} \end{cases}$$

The calculation of $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ is simply the product of similarity of the pairs in the *minimum pair set* that covers $\hat{p}$ and every $p_i \in \mathbf{P}$. The reason we use product rather than summation to integrate similarity scores of the pairs is that even if only one term in either the relation $\hat{r}$ or the path $\mathcal{P}$ cannot be well aligned we should not think $\hat{r}$ and $\mathcal{P}$ are semantically similar since one term can change the meaning drastically.

Now, we start to describe how to find the cutting, namely, the function ω. We seek the cutting function ω^* in the cutting space Ω that maximizes $Sim_{\bowtie}(\hat{p}, \mathbf{P})$. More specifically,

(5.11)

$$\omega^* = \underset{\omega \in \Omega}{argmax}\ Sim_{\bowtie}(\hat{p}, \mathbf{P})$$

The sequence $\{c_0, p_0, p_1, p_2, \cdot\cdot\cdot, p_{l-1}, c_l\}$ has $l+2$ elements, which leads to $l+3$ positions to cut. There are totally C^2_{l+3} different ways of selecting two cuts. Therefore, the cutting space Ω has a size of C^2_{l+3}. Going through every $\omega \in \Omega$ and compute $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ is computationally expensive, which requires a running time of $O(l^3)$. To address this problem, we develop a greedy algorithm, *SmartCutter*, to perform the cutting task that runs in $O(l)$. Figure 5.5 presents the pseudo code of *SmartCutter*. *SmartCutter* can handle all the four cases in Figure 5.4.

Algorithm SmartCutter

Input: subject $\hat{c}$, predicate $\hat{p}$, object $\hat{c}'$, the relation similarity model $\mathrm{sim_r}$, and the array t that contains the sequence $\langle c_0, p_0, p_1, p_2, \cdots, p_{l-1}, c_l \rangle$.

Output: u, an index of t that points to the rightmost term in the subject region, and v, an index of t that points to the leftmost term in the object region.

```
$best \leftarrow 0$
$bestsim \leftarrow 0$
**for** $i \leftarrow 0$ to $l + 1$ **do**
    **if** $\mathrm{sim_r}(\hat{p}, t[i]) > bestsim$ **then**
        $best \leftarrow i$
        $bestsim \leftarrow \mathrm{sim_r}(\hat{p}, t[i])$
$u \leftarrow best - 1$
$v \leftarrow best + 1$
**while** $u > 0$ **do**
    **if** $\mathrm{sim_r}(\hat{p}, t[u]) \geq \mathrm{sim_r}(\hat{c}, t[u])$ **then**
        $u \leftarrow u - 1$
    **else if** $u = l$ and $\mathrm{sim_r}(\hat{p}, t[u]) \geq \mathrm{sim_r}(\hat{c}, t[u]) \cdot bestsim$ **then**
        $u \leftarrow u - 1$
    **else**
        break
**while** $v < l + 1$ **do**
    **if** $\mathrm{sim_r}(\hat{p}, t[v]) \geq \mathrm{sim_r}(\hat{c}', t[v])$ **then**
        $v \leftarrow v + 1$
    **else if** $v = 1$ and $\mathrm{sim_r}(\hat{p}, t[v]) \geq \mathrm{sim_r}(\hat{c}', t[v]) \cdot bestsim$ **then**
        $v \leftarrow v + 1$
    **else**
        break
**return** $u, v$
```

FIG. 5.5. Predicate-driven greedy cutting algorithm

The general idea of *SmartCutter* is as follows. We first find the element in the sequence $\{c_0, p_0, p_1, p_2, \cdots, p_{l-1}, c_l\}$ that has the largest semantic similarity with the predicate $\hat{p}$. We call it as the best term. We assume that the best term is inside the predicate region and we initiate the predicate region to only include the best term. Next, we stretch the predicate region to the left of the best term until we meet an element u that is more similar to the subject $\hat{c}$ than to the predicate $\hat{p}$. Similarly, we stretch the predicate region to the right of the best term until we meet an element v that is more similar to the object $\hat{c}'$. Finally, the two elements u and v give the borders of the three regions.

Lines $1-6$ shows the code used to find the best term. Lines $9-15$ and $16-22$ show how we find u and v, respectively. The line $12-13$ and $19-20$ are used to deal with a special case that the best term is one of two ending classes, either c_0 or c_l. Proceeding next stretch step or not will determine whether $\hat{p} \in \{\omega(p_i) \mid p_i \in \mathbf{P}\}$ and which of the two cases to use for computing $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ in Equation 5.10.

Property 5.3.1. *$SmartCutter$ makes each stretch step iff $\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) \geqslant 0$.*

Proof. Without loss of generality, we prove the property in the process of stretching to the left of the best term. The proof is divided into Case 1, 2 and 3. Case 2 is further divided into 2a and 2b. Let the cutting function before one stretch step be ω' and after the stretch step be ω''. ω' and ω'' produce the same value on every $p_i \in \mathbf{P}$ except on p_k where $k = u - 1$.

Case 1. *the best term is not c_0 or c_l.*

$SmartCutter$ makes the stretch step if and only if $\mathrm{sim_r}(\hat{p}, p_k) \geq \mathrm{sim_r}(\hat{c}, p_k)$ because only lines $10-13$ control whether taking the stretch step towards the left and line 13 is unreachable in Case 1.

We have $\hat{p} \in \{\omega'(p_i) \mid p_i \in \mathbf{P}\}$ and $\hat{p} \in \{\omega''(p_i) \mid p_i \in \mathbf{P}\}$ since the best term belongs to $\mathbf{P}$ and maps to $\hat{p}$. Consequently, we have $Sim_{\bowtie}(\hat{p}, \mathbf{P}) = \prod_{i=0}^{l-1} \mathrm{sim_r}(\omega(p_i), p_i)$ for both $\omega = \omega'$ and $\omega = \omega''$.

(5.12)

$$\begin{aligned}\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) &= \prod_{i=0}^{l-1} \mathrm{sim_r}(\omega''(p_i), p_i) - \prod_{i=0}^{l-1} \mathrm{sim_r}(\omega'(p_i), p_i)\\ &= \prod_{i=0}^{k-1} \mathrm{sim_r}(\omega'(p_i), p_i) \cdot \prod_{i=k+1}^{l-1} \mathrm{sim_r}(\omega'(p_i), p_i) \cdot (\mathrm{sim_r}(\omega''(p_k), p_k) - \mathrm{sim_r}(\omega'(p_k), p_k))\\ &= \prod_{i=0}^{k-1} \mathrm{sim_r}(\omega'(p_i), p_i) \cdot \prod_{i=k+1}^{l-1} \mathrm{sim_r}(\omega'(p_i), p_i) \cdot (\mathrm{sim_r}(\hat{p}, p_k) - \mathrm{sim_r}(\hat{c}, p_k))\end{aligned}$$

From Equation 5.12, it follows that $\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) \geqslant 0 \iff \mathrm{sim_r}(\hat{p}, p_k) \geq \mathrm{sim_r}(\hat{c}, p_k)$. Therefore, we prove the property for Case 1.

Case 2. *the best term is c_l.*

Case 2a. *when $u = l$ in the while loop from line 9 to line 15.*

By $k = u - 1$, we know $k = l - 1$. $SmartCutter$ makes the stretch step if $\mathrm{sim_r}(\hat{p}, p_{l-1}) \geq \mathrm{sim_r}(\hat{c}, p_{l-1}) \cdot \mathrm{sim_r}(\hat{p}, c_l)$ due to line $12 - 13$. On the other hand, $SmartCutter$ makes the stretch step only if the condition in line 10 or 12 is satisfied. Either line 10 or 12 is satisfied, we all have $\mathrm{sim_r}(\hat{p}, p_{l-1}) \geq \mathrm{sim_r}(\hat{c}, p_{l-1}) \cdot \mathrm{sim_r}(\hat{p}, c_l)$. Therefore, it follows that $SmartCutter$ makes the stretch step only if $\mathrm{sim_r}(\hat{p}, p_{l-1}) \geq \mathrm{sim_r}(\hat{c}, p_{l-1}) \cdot \mathrm{sim_r}(\hat{p}, c_l)$.

We have $\hat{p} \notin \{\omega'(p_i) \mid p_i \in \mathbf{P}\}$ and $\hat{p} \in \{\omega''(p_i) \mid p_i \in \mathbf{P}\}$. Consequently, we have $Sim_{\bowtie}(\hat{p}, \mathbf{P}) = (\prod_{i=0}^{l-1} \mathrm{sim_r}(\omega(p_i), p_i)) \cdot \mathrm{sim_r}(\hat{p}, \omega^{-1}(\hat{p}))$ when $\omega = \omega'$ and $Sim_{\bowtie}(\hat{p}, \mathbf{P}) = \prod_{i=0}^{l-1} \mathrm{sim_r}(\omega(p_i), p_i)$ when $\omega = \omega''$.

(5.13)

$$\begin{aligned}
\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) &= \prod_{i=0}^{l-1} \mathrm{sim_r}(\omega''(p_i), p_i) - (\prod_{i=0}^{l-1} \mathrm{sim_r}(\omega'(p_i), p_i)) \cdot \mathrm{sim_r}(\hat{p}, \omega'^{-1}(\hat{p})) \\
&= \prod_{i=0}^{l-2} \mathrm{sim_r}(\omega'(p_i), p_i) \cdot (\mathrm{sim_r}(\omega''(p_{l-1}), p_{l-1}) - \mathrm{sim_r}(\omega'(p_{l-1}), p_{l-1}) \cdot \mathrm{sim_r}(\hat{p}, c_l)) \\
&= \prod_{i=0}^{l-2} \mathrm{sim_r}(\omega'(p_i), p_i) \cdot (\mathrm{sim_r}(\hat{p}, p_{l-1}) - \mathrm{sim_r}(\hat{c}, p_{l-1}) \cdot \mathrm{sim_r}(\hat{p}, c_l))
\end{aligned}$$

From Equation 5.13, it follows that $\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) \geqslant 0 \iff \mathrm{sim_r}(\hat{p}, p_{l-1}) \geq \mathrm{sim_r}(\hat{c}, p_{l-1}) \cdot \mathrm{sim_r}(\hat{p}, c_l)$. Therefore, we prove the property for Case 2a.

Case 2b. *when $u \leq l-1$ in the while loop from line 9 to line 15.*

This can be proved in the same way as for Case 1. except that we have $\hat{p} \in \{\omega'(p_i) \mid p_i \in \mathbf{P}\}$ and $\hat{p} \in \{\omega''(p_i) \mid p_i \in \mathbf{P}\}$ because both ω' and ω'' map p_{l-1} to $\hat{p}$.

Case 3. *the best term is c_0.*

There is no need to stretch to the left for this case.

□

Property 5.3.2. *$SmartCutter$ stops stretching towards either direction iff $\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) < 0$.*

Proof. The property can be deduced directly from Property 5.3.1. □

The reason why $SmartCutter$ stops when $\Delta Sim_{\bowtie}(\hat{p}, \mathbf{P}) < 0$ is that we already start to lose $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ score at the current step and it is more likely to lose than gain in the future steps since the cutting line will be put even closer to c_0 or c_l and farther from the best term.

We observed that $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ tends to be biased towards small l, that is, short paths. To counteract it, we raise $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ to the power of $\frac{1}{1+\alpha(l-1)}$. More specifically,

(5.14) $$Sim'_{\bowtie}(\hat{p}, \mathbf{P}) = (Sim_{\bowtie}(\hat{p}, \mathbf{P}))^{\frac{1}{1+\alpha(l-1)}}$$

where α is a parameter having a value in the range $[0..1]$. When α is 0, $Sim'_{\bowtie}(\hat{p}, \mathbf{P})$ is the same as $Sim_{\bowtie}(\hat{p}, \mathbf{P})$. When α is 1, $Sim'_{\bowtie}(\hat{p}, \mathbf{P})$ becomes the l-th root of $Sim_{\bowtie}(\hat{p}, \mathbf{P})$, which is approximately the geometric mean of the lexical similarity scores included in the product computation of $Sim_{\bowtie}(\hat{p}, \mathbf{P})$.

However, the method we use to counteract the bias brings a new problem of its own. As we know, $Sim_{\bowtie}(\hat{p}, \mathbf{P})$ is the product of similarity scores in three regions, subject, predicate and object. The paths that have very high similarity scores in the subject and object regions but low similarity scores in the predicate region can still enjoy a fairly high $Sim'_{\bowtie}(\hat{p}, \mathbf{P})$ score due to the power of $\frac{1}{1+\alpha(l-1)}$. This can cause a problem because the predicate $\hat{p}$ may not be well aligned by any term in the path $\mathcal{P}$. To address this problem, we impose $Sim'_{\bowtie}(\hat{p}, \mathbf{P})$ a upper limit, which is the maximum similarity score obtained in the predicate region. More specifically,

(5.15) $$Sim''_{\bowtie}(\hat{p}, \mathbf{P}) = \min\left(\underset{\{p_i \mid \omega(p_i)=\hat{p}\}}{max} sim(\hat{p}, p_i),\ Sim'_{\bowtie}(\hat{p}, \mathbf{P})\right)$$

Finally, the formula that we use to compute the overall joint lexical semantic similarity between $\hat{r}$ and $\mathcal{P}$ is shown in Equation 5.16.

(5.16) $$Sim_{\bowtie}(\hat{r}, \mathcal{P}) = \mathrm{sim_c}(\hat{c}, c_0) \cdot \mathrm{sim_c}(\hat{c}', c_l) \cdot (Sim''_{\bowtie}(\hat{p}, \mathbf{P}))$$

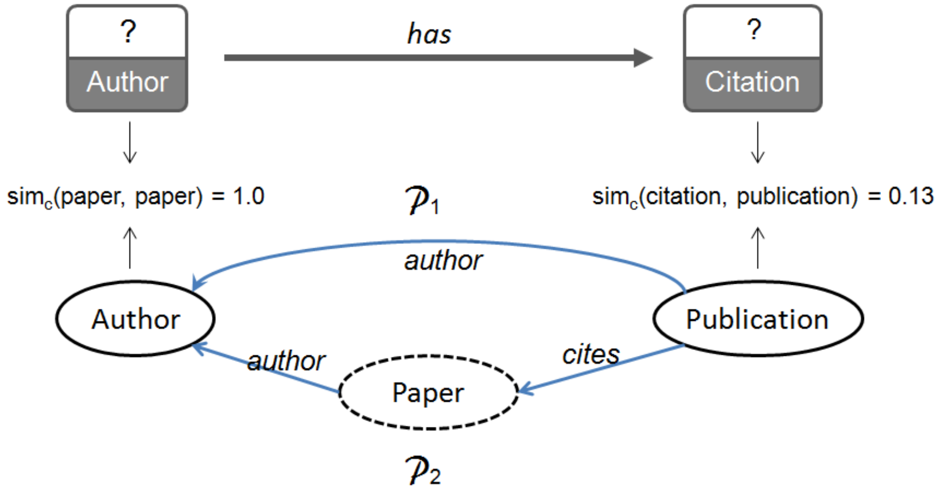


FIG. 5.6. Dealing with default relations in computing joint lexical semantic similarity

The formula is simply the product of three similarity terms. The first two similarity terms result from pairing the subject concept $\hat{c}$ to the starting class c_0 of the path $\mathcal{P}$ and pair the object concept $\hat{c}'$ to the ending class c_l. The third similarity term is $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$, resulting from pairing the predicate $\hat{p}$ to the terms in $\mathcal{P}$ and paring every property in $\mathbf{P}$ to $\hat{c}$, $\hat{p}$ or $\hat{c}'$.

5.3.3 Dealing with Default Relations

Computing $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ requires the predicate $\hat{p}$ to be meaningful since the cutting algorithm is predicate-driven. When it comes to default relations, we can replace the empty predicate $\hat{p}$ first with the subject concept $\hat{c}$ and next with the object concept $\hat{c}'$, compute $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ for both cases and finally combine the two values into one.

However, we cannot apply $\max$ operator or use simple average to combine them. Consider the example in Figure 5.6. If we substitute “Author” for the empty predicate $\hat{p}$, the path $\mathcal{P}_1$ will be preferable to $\mathcal{P}_2$, according to their $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ scores. If we substitute “Citation” for $\hat{p}$, $\mathcal{P}_2$ will be preferable to $\mathcal{P}_1$. It is the low semantic similarity between the object concept “Citation” and the class *Publication*, 0.13, that causes the incorrect ranking when we substitute “Author” for the predicate. The low semantic similarity implies that significant portion of semantics in “Citation” is not matched by *Publication*. In this case, if we use “Author” as the predicate, that portion of unmatched semantics may never be able to be matched, as illustrated by $\mathcal{P}_1$. Therefore, “Citation” should be given preference over “Author” in replacing the empty predicate.

Let the concept having smaller semantic similarity with its corresponding class be $\hat{c}_s$ and the one with larger similarity be $\hat{c}_b$. Let $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$ and $Sim''_{\bowtie}(\hat{c}_b, \mathbf{P})$ represent $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ after substituting $\hat{c}_s$ and $\hat{c}_b$ for $\hat{p}$, respectively. The joint lexical semantic similarity with respect to the empty predicate $\hat{p}$ in a default relation and the properties $\mathbf{P}$ in a path $\mathcal{P}$ is shown in Equation 5.17.

(5.17)

$$Sim''_{\bowtie}(\hat{p}, \mathbf{P}) = \begin{cases} Sim''_{\bowtie}(\hat{c}_s, \mathbf{P}) & \text{if } Sim''_{\bowtie}(\hat{c}_s, \mathbf{P}) \geqslant Sim''_{\bowtie}(\hat{c}_b, \mathbf{P}) \\ Sim''_{\bowtie}(\hat{c}_s, \mathbf{P}) + (Sim''_{\bowtie}(\hat{c}_b, \mathbf{P}) - Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})) \cdot (\dfrac{\mathrm{sim_r}(\hat{c}_s, \sigma(\hat{c}_s))}{\mathrm{sim_r}(\hat{c}_b, \sigma(\hat{c}_b))})^{\theta} & \text{otherwise} \end{cases}$$

where θ is a parameter in the range $[0, \infty)$. When θ is 0, $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ becomes the larger one between $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$ and $Sim''_{\bowtie}(\hat{c}_b, \mathbf{P})$. When θ is a large number, $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ is close to $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$. The ratio $\frac{\mathrm{sim_r}(\hat{c}_s,\sigma(\hat{c}_s))}{\mathrm{sim_r}(\hat{c}_b,\sigma(\hat{c}_b))}$ denotes the ratio of the smaller concept similarity to the bigger concept similarity. The lower the ratio is, the closer $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ is to $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$, or in other words, the more preference is given to $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$. If the ratio is 1.0, $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ again becomes the larger one between $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$ and $Sim''_{\bowtie}(\hat{c}_b, \mathbf{P})$.

We use the example in Figure 5.6 to illustrate the computation of $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ where we set $\theta = 1.0$. According to the two concept similarity scores, $\hat{c}_s$ and $\hat{c}_b$ represent “Citation” and “Author”, respectively, and the ratio $\frac{\mathrm{sim_r}(\hat{c}_s,\sigma(\hat{c}_s))}{\mathrm{sim_r}(\hat{c}_b,\sigma(\hat{c}_b))}$ have a value 0.13. For the path $\mathcal{P}_1$, $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$ and $Sim''_{\bowtie}(\hat{c}_b, \mathbf{P})$ produce 0.22 and 1.0, respectively. Therefore, $Sim''_{\bowtie}(\hat{p}, \mathbf{P}) = 0.22 + (1.0 - 0.22) * 0.13 = 0.32$, which is much closer to 0.22 than to

1.0. For the path $\mathcal{P}_2$, $Sim''_{\bowtie}(\hat{c}_s, \mathbf{P})$ and $Sim''_{\bowtie}(\hat{c}_b, \mathbf{P})$ produce 0.818 and 0.852, respectively. Hence, $Sim''_{\bowtie}(\hat{p}, \mathbf{P}) = 0.818 + (0.852 - 0.818) * 0.13 = 0.822$. Thus, $\mathcal{P}_2$ is more preferable than $\mathcal{P}_1$ by their $Sim''_{\bowtie}(\hat{p}, \mathbf{P})$ scores.

Chapter 6

FORMAL QUERY GENERATING AND ENTITY MATCHING

The result of a mapping σ is a graph, which we refer to as G_σ. We can literally translate G_σ into a SPARQL graph query. Figure 6.2 shows the SPARQL query produced from the mapping in Figure 6.1. We create variables for the classes in G_σ that are corresponding to SFQ nodes with a "?" mark, such as "?x" for *Actor*, and blank nodes for the classes that are corresponding to SFQ nodes with a "*" mark and all the intermediate classes, such as "_:b01" for *Film*. The variables and blank nodes are typed with the classes, such as "?x a dbo:Actor", and are inter-linked by the edges in G_σ, such as "_:b01 dbo:starring ?x".

However, creating SPARQL nodes for named entities can be complicated. Although names are the identifiers of entities, they can have certain variations. Typically, a knowledge base maintains an entity-name index that maps different names of an entity to the database id or URI of the entity. When it comes to RDF knowledge base, we can build this index on *rdfs:label* properties. However, it is often the case that this index is incomplete (i.e. not containing all the name variations). Therefore, we need an approach to match the entity name. Figure 6.2 shows a simple but effective approach that applies full-text or keyword search to match entity names. The *bif:contains* property is a Virtuoso [29] built-in text search function which finds literals containing specified text. Alternatively, we can

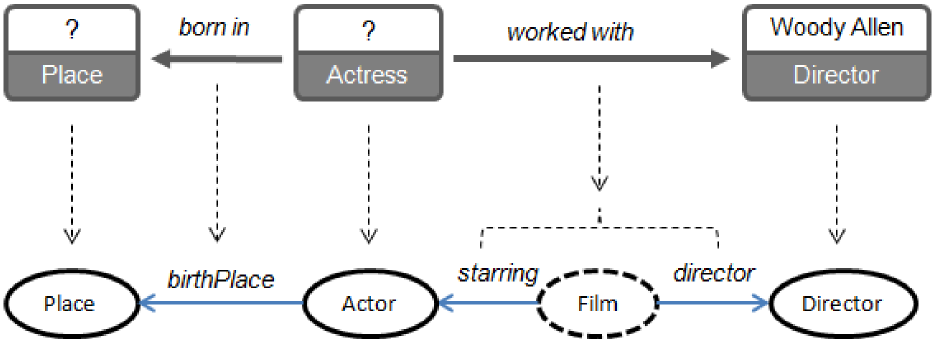


FIG. 6.1. A SFQ mapping example

use regex match to achieve the same end, which is supported by SPARQL 1.1. Figure 6.3 illustrates an example that how we match named entities using a regex filter. However, the implementation of full-text search, *bif:contains*, is much faster than that of regular expression matching in Virtuoso.

In Section 9.4.9, we will describe a pragmatic approach for named entity matching that is slightly more complicated. It first produces SPARQL queries using exact matches on the entity names. If the SPARQL queries return empty results, we then resort to keyword matching. Sophisticated approaches for named entity matching can involve developing algorithms for fast approximate string matching and retrieval, but they are out of scope of this thesis.

In the last chapter, we show how to disambiguate concepts and relations using novel algorithms but we have not done anything to disambiguate named entities. Actually, the name-matched entities can be disambiguated by the constraints in the SPARQL query. For example, since the entity "?0" with the label "Woody Allen" is the "dbo:director" of some film "_:b01", the other possible matches, like books, can be excluded by the query. Therefore, we do not need to develop a separate module to deal with named entity disambiguation since it comes free when the query is executed.

```
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?x, ?y WHERE {
  _:b01 a dbo:Film .
  ?x a dbo:Actor .
  ?y a dbo:Place .
  ?0 rdfs:label ?label0 .
  ?label0 bif:contains ’"Woody Allen"’ .
  ?x dbo:birthPlace ?y .
  _:b01 dbo:starring ?x .
  _:b01 dbo:director ?0 .
}
```

FIG. 6.2. This SPARQL query was automatically generated from the SFQ in Figure 6.1, “Which Actresses worked with Woody Allen and where were they born?”.

```
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?x ?y WHERE {
 _:b01 a dbo:Film .
 ?x a dbo:Actor .
 ?y a dbo:Place .
 ?0 rdfs:label ?label0 .
   FILTER regex(?label0, "woody allen", "i")
 ?x dbo:birthPlace ?y .
 _:b01 dbo:starring ?x .
 _:b01 dbo:director ?0 .
}
```

FIG. 6.3. An alternative SPARQL query that uses regex function to match a named entity.

Chapter 7

IMPROVING PMI IN MEASURING ASSOCIATION

Although Pointwise Mutual Information (PMI) has been mainly applied to statistical NLP, it is a general measure of statistical association. However, PMI has a well-known problem that it is biased towards low-frequency terms. In this chapter, we develop an approach to offset the bias in the domain of measuring word similarity and present a new metric PMI_{max}. The reason PMI can be applied to measure word similarity is that semantically similar words tend to occur near each other, which is different from the distributional hypothesis [42].

In Section 7.6, we simplify PMI_{max} and make it applicable to other domains. In particular, we show how to come up with a metric that can be used to measure statistical association between schema terms on the schema network, which is a totally different domain from measuring word similarity.

Readers shall not mistake this chapter for the place of implementation of our semantic similarity models, which is actually in the next chapter. Although PMI_{max} can work as a good semantic similarity metric, especially for measuring *relation similarity*, it has two limitations that prevent us from using it immediately in our schema-free querying system. First, PMI_{max} as well as PMI need a very large corpus to be effective. In the experiments conducted in this chapter we use a two-billion words corpus, but we still have to limit word frequency to be above 700 to make PMI measures statistically reliable. Second, similarity computation in our query interpretation algorithms requires similarity scores falling in the range $[0, 1]$ but PMI measures do not hold this property.

7.1 Introduction

Word similarity is a measure of how semantically similar a pair of words is, with synonyms having the highest value. It is widely used for applications in natural language processing (NLP), information retrieval, and artificial intelligence, including tasks like word sense disambiguation [86], malapropism detection [11], paraphrase recognition [71], image and document retrieval [105] and predicting hyperlink-following behavior [54]. There are two prevailing approaches to computing word similarity, based on either using of a thesaurus (e.g., WordNet [72]) or statistics from a large corpus. There are also hybrid approaches [75] combining the two methods. Many well-known word similarity measures have been based on WordNet [85, 64, 49] and most of semantic applications [86, 105, 11] rely on these taxonomy-based measures.

Organizing all words in a well-defined taxonomy and linking them together with different relations is a labor-intensive task that requires significant maintenance as new words and word senses are formed. Furthermore, existing WordNet-based similarity measures typically depend heavily on “IS-A” information, which is available for nouns but incomplete for verbs and completely lacking for adjectives and adverbs. Consequently, these metrics perform poorly (with accuracy no more than 25%) [48] in answering TOEFL synonym questions [55] where the goal is selecting which of four candidate choices is most like a synonym to a given word. In contrast, some corpus-based approaches achieve much higher accuracy on the task (above 80%) [96, 12].

We expect statistical word similarity to continue to play an important role in semantic acquisition from text [64] in the future. A common immediate application is automatic thesaurus generation, in which various statistical word similarity measures [46, 36, 62, 22, 112] have been proposed. These are based on the distributional hypothesis [42], which states that words occurring in the same contexts tend to have similar meanings. Thus, the meaning of a word can be represented by a context vector of accompanying words and their co-occurrences counts, modulated perhaps by weighting functions [22], measured either in document context, text window context, or grammatical dependency context. The context vectors can be further transformed to a space of reduced dimension by applying singular value decomposition (SVD), yielding the familiar latent semantic analysis (LSA) technique [55]. The similarity of two words is then computed as the similarity of their context vectors, and the most common metric is the cosine of the angle between the two vectors. We will refer this kind of word similarity as distributional similarity, following convention in the research community [77, 108, 11].

PMI has emerged as a popular statistical word similarity measure that is not based on the distributional hypothesis. Calculating PMI only requires simple statistics about two words: their marginal frequencies and their co-occurrence frequency in a corpus. In the ten years after PMI was introduced to statistical NLP by Church and Hanks [17] it was mainly used for measuring word association [68] and was not thought of as a word similarity measure. Along with other statistical association measures such as the t-test, χ^2-test, and likelihood ratio, PMI was commonly used for finding collocations [68]. PMI was also a popular weighting function used in computing distributional similarity measures [46, 62, 108]. Using PMI as a word similarity measure began with the work of Turney [103], who developed a technique he called PMI-IR that used page counts from a Web search engine to approximate frequency counts in computing PMI values for word pairs. This produced remarkably good performance in answering TOEFL synonym questions – an accuracy of 74% which outperformed LSA [55], and was the best result at that time.

Turney's result was surprising because finding synonyms was a typical task for distributional similarity measures, and PMI, a word association measure, performed even better. Terra and Clarke [96] redid the TOEFL experiments for a set of the most well-known word association measures including PMI, χ^2-test, likelihood ratio, and average mutual information. The experiments were based on a very large Web corpus, and document and text window contexts of various sizes were investigated. They found PMI performed the best overall, and obtained an accuracy of 81.25% with a window size of 16 to 32 words.

PMI-IR has subsequently been used as a word similarity measure in other applications with good results. Mihalcea [71] used PMI-IR, LSA, and six WordNet based similarity measures as a sub-module in computing text similarity and applying it to paraphrase recognition and found that PMI-IR slightly outperformed the others. In a task of predicting user click behavior, predicting the HTML hyperlinks that a user is most likely to select given an information goal, Kaur [54] also showed that PMI-IR performs better than LSA and six WordNet based measures.

Why PMI is effective as a word similarity measure is still not clear. Many researchers use Turney's good results of PMI-IR as empirical credence [96, 45, 71, 54, 50]. To explain the success of PMI, some propose the proximity hypothesis [45, 96] noting that similar words tend to occur near each other, which is quite different from the distributional hypothesis which assumes that similar words tend to occur in similar contexts. However, to our knowledge no further explanations have been provided in the literature.

In this chapter, we offer an intuitive explanation for why PMI can be used as a word similarity measure and illustrate behavioral differences between first-order PMI similarity and second-order distributional similarity. We also provide new experiments and examples, allowing more insight into PMI similarity. Our main contribution is introducing a novel metric, PMI_{max}, that enhances PMI to take into account the fact that words have multiple senses. The new metric is derived from the assumption that more frequent content words have more senses. We show that PMI_{max} significantly improves the performance of PMI in the application of automatic thesaurus generation and outperforms PMI on benchmark datasets including human similarity rating datasets and TOEFL synonym questions. PMI_{max} also has the advantage of not requiring expensive resources, such as sense-annotated corpora.

The remainder of the chapter proceeds as follows. In Section 2 we discuss PMI similarity and define the PMI_{max} metric. In Section 3 we use experiments in automatic thesaurus generation to determine the coefficients of PMI_{max}, evaluate its performance, and examine its assumptions. Additional evaluation using benchmark datasets is presented in Section 4. Section 5 discusses potential applications of PMI_{max} and our future work. We are particularly interested in exploiting behavioral differences between PMI similarity and distributional similarity in the application of semantic acquisition from text. Finally, we conclude the chapter in Section 6.

7.2 Approach

We start this section by discussing why PMI can serve as a semantic similarity measure. Then, we point out a problem introduced by PMI's assumption that words only possess a single sense, and we propose a novel PMI metric to consider polysemy of words.

7.2.1 PMI as a Semantic Similarity Measure

Intuitively, the semantic similarity between two concepts[1] can be defined as how much commonality they share. Since there are different ways to define commonality, semantic similarity tends to be a fuzzy concept. Is *soldier* more similar to *astronomer* or to *gun*? If commonality is defined as purely involving *IS-A* relations in a taxonomy such as WordNet,

[1] A concept refers to a particular sense of a word and we use an italic word to signify it in this section.

then *soldier* would be more similar to *astronomer* because both are types of people. But if we base commonality on aptness to a domain, then *soldier* would be more similar to *gun*. People naturally do both types of reasoning, and to evaluate computational semantic similarity measures, the standard practice is to rely on subjective human judgments.

Many researchers think that semantic similarity ought to be based only on *IS-A* relations and that it represents a special case of semantic relatedness which also includes antonymy, meronymy and other associations [11]. However, in the literature semantic similarity also refers to the notion of belonging to the same semantic domain or topic, and is interchangeable with semantic relatedness or semantic distance [68]. In this paper, we take the more relaxed view and use two indicators to assess the goodness of a statistical word similarity measure: (i) its ability to find synonyms, and (ii) how well it agrees with human similarity ratings.

We use Nida's example noted by Lin [62] to help describe why PMI can be a semantic similarity measure:

> A bottle of tezgüino is on the table.
>
> Everyone likes tezgüino.
>
> Tezgüino makes you drunk.
>
> We make tezgüino out of corn.

By observing the contexts in which the concept *tezgüino* is used, we can infer that *tezgüino* is a kind of alcoholic beverage made from corn, exemplifying the idea that a concept's meaning can be characterized by its contexts. By saying a concept has a context, we mean the concept is likely to appear in the context. For example, *tezgüino* is likely to appear in the context of "drunk", so *tezgüino* has the context "drunk". We may then define concept similarity as how much their contexts overlap, as illustrated in Figure 7.1. The larger the overlap becomes, the more similar the two concepts are, and vice versa.

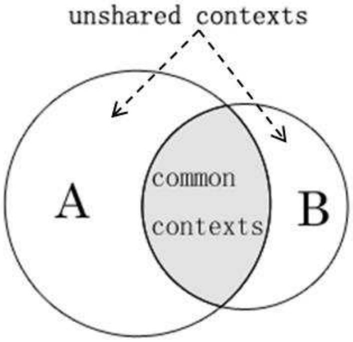


FIG. 7.1. Common contexts between concepts A and B.

Two concepts are more likely to co-occur in a common, shared context and less likely in an unshared one. In a shared context, both have an increased probability of appearing but in an unshared one, as in Figure 7.1, one is more likely but the other not. Generally, for two concepts with fixed sizes[2], the larger their context overlap is, the more co-occurrences result. In turn, the number of co-occurrences can be used to indicate the amount of common contexts between two concepts with fixed sizes.

The number of co-occurrences also depends on the sizes of the two concepts. Therefore, we need a normalized measure of co-occurrences to represent their similarity. PMI fits this role well. Equation 7.1 shows how to compute PMI for concepts in a sense-annotated text corpus, where f_{c_1} and f_{c_2} are the individual frequencies (counts) of the two concepts c_1 and c_2 in the corpus, and $f_d(c_1, c_2)$ is the co-occurrence frequency of c_1 and c_2 measured by the context window of d words and N is the total number of words in the corpus. In this chapter, $\log$ always stands for natural logarithm.

$$\mathrm{PMI}(c_1, c_2) \approx \log(\frac{f_d(c_1, c_2) \cdot N}{f_{c_1} \cdot f_{c_2}}) \tag{7.1}$$

Traditionally, PMI is explained as the logarithmic ratio of the actual joint probability of two

[2] The size of a concept refers to the frequency count of the concept in a corpus.

events to the expected joint probability if the two events were independent [17]. Here, we interpret it from a slightly different perspective and this interpretation is used in deriving our novel PMI metric in the next section. The term $f_{c_1} \cdot f_{c_2}$ can be interpreted as the number of all co-occurrence possibilities or combinations between c_1 and c_2[3]. The term $f_d(c_1, c_2)$ gives the number of co-occurrences actually fulfilled. Thus, the ratio $\frac{f_d(c_1,c_2)}{f_{c_1} \cdot f_{c_2}}$ measures the extent to which two concepts tend to co-occur. By analogy to the correlation in statistics which measures the degree that two random variables tend to co-increase/decrease, PMI measures the likelihood that two concepts tend to co-occur versus occurring alone. In this sense, we say that PMI computes the correlation between the concepts c_1 and c_2. We also refer to the semantic similarity that PMI represents as correlation similarity.

Because a concept has the largest context overlap with itself, a concept has the largest chance to co-occur with itself. In other words, a concept has the strongest correlation with itself (auto-correlation). Auto-correlation is closely related to word burstiness, a phenomenon that words tend to appear in bursts. If the “one sense per discourse” hypothesis [34] is applied, word burstiness would be essentially the same as concept burstiness. Thus word burstiness is a reflection of the auto-correlation of concepts.

It is interesting that synonym correlation derives from the auto-correlation of concepts. If identical concepts happen within a distance as short as a few sentences, writers often prefer using synonyms to avoid excessive lexical repetition. The probability of substituting synonyms depends on the nature of the concept as well as the writer’s literary style. In our observations, synonym correlation often has a value very close to auto-correlation for verb, adjective and adverb concepts, but a value a little bit lower for nominal concepts. One reason may be the ability to use a pronoun as an alternative to a synonym in English.

[3] The requirement for multiplication rather than addition can be easily understood by an example: if one individual frequency increased twofold, all co-occurrence possibilities would be doubled rather than increased by the individual frequency

Although verbs, adjectives and adverbs also have pronominal forms, they are less powerful than pronouns and used less frequently.

PMI similarity is related to distributional similarity in that both are context-based similarity measures. However, they use contexts differently which results in different behaviors. First, the PMI similarity between two concepts is determined by how much their contexts overlap, while distributional similarity depends on the extent that two concepts have similar context distributions. For example, *garage* has a very high PMI similarity with *car* because *garage* rarely occurs in contexts which are not subsumed by the contexts of *car*. However, distributional similarity typically does not consider *garage* to be very similar to *car* because the context distributions of *garage* and *car* vary considerably. While one might expect all words related by a PART-OF relation to have high PMI similarity, this is not the case. *Table* is not PMI-similar to *leg*, because there are many other contexts related to *leg* but not *table*, and vice versa.

Second, for two given concepts, distributional similarity obtains collective contextual information for each concept and computes how similar their context vectors are. For distributional similarity it does not require the concepts to co-occur in the same contexts to be similar. In contrast, PMI similarity emphasizes the propensity for two concepts to co-occur in the exactly same contexts. For example, the distributional similar concepts for *car* may include not only *automobile*, *truck*, and *train*, but also *boat*, *ship*, *carriage* and *chariot*. This shows the ability of distributional similarity to find “indirect” similarity. However, a problem with distributional similarity is that it cannot distinguish them and separate them into three categories: “land vehicle”, “water vehicle”, and “archaic vehicle”. On the other hand, the PMI-similar concepts for *car* do not contain *boat* or *carriage* because they do not co-occur with *car* in the same contexts frequently enough.

These differences suggest that PMI similarity should be a valuable complement to distributional similarity. Although the idea that PMI can complement distributional similarity is not new, current techniques [104, 50] have used them as separate features in statistical models and do not exploit how that they differ. In Section 7.5 we will show through examples that we can support interesting applications by exploiting their behavioral differences.

7.2.2 Augmenting PMI to Account for Polysemy

Since it is very expensive to produce a large sense-tagged corpus, statistical semantic similarity measures are often computed based on words rather than word senses [11]. However, when PMI is applied to measure correlation between words[4], it has a problem because it assumes that words only have a single sense. Consider “make” and “earn” as an example. “Make” has many senses, only one of which is synonymous with “earn”, and so it is inappropriate to divide by the whole frequency of “make” in computing the PMI correlation similarity between “make” and “earn”, since only a fraction of “make” occurrences have the same meaning of “earn”.

PMI has a well-known problem that it tends to over-emphasize the association of low frequency words [78, 107, 110]. We conjecture that the fact that more frequent content words tend to have more senses, as shown in Figure 7.4, is an important cause for PMI’s frequency bias. More frequent words are disadvantaged in producing high PMI value because they tend to have more unrelated or less related senses and thereby bear an extra burden by including the frequencies of these senses in their marginal counts. We should distinguish this cause from the one described by Dunning [28] that the normality assumption breaks down on rare words, which can be ruled out by using a minimum frequency threshold.

Although it can be difficult to determine which sense of a word is being used, this does not prevent us from making a more accurate assumption than the “single sense” assumption.

[4] The same formula as in Equation 7.1 is used, except that senses are replaced by words

We will demonstrate a significant improvement over traditional PMI by merely assuming that more frequent content words have a greater number of senses.

We start by modeling the number of the senses of an open-class word (i.e., noun, verb, adjective, or adverb) as a power function of the log frequency of the word with a horizontal translation q. More specifically,

$$y_w = a(\log(f_w) + q)^p \tag{7.2}$$

where f_w and y_w are the frequency (count) of the word w and its number of senses respectively; a, p and q are three coefficients needed to resolve. The displacement q is necessary because f_w is not a normalized measure of the word w and varies on the size of the selected corpus. We require $(\log(f_w)+q > 0)$ to only take the strict monotone increasing part of the power function. The power function assumption is based on observing the graphs in Figure 7.4, which show the dependence between a word's log frequency in a large corpus (e.g., two billion words) and its number of senses obtained from WordNet. This relationship, though simple, is better modeled using a power function rather than a linear one.

We next estimate a word pair's PMI value between their closest senses using two assumptions. Given a word pair, it is hard to know the proportions at which the closest senses are engaged in their own words. Since it can be either a major or minor sense, we simply assume the average proportion $\frac{1}{y_w}$. Consequently, the frequency of a word used as the sense most correlated with a sense in the other word is estimated as $\frac{f_w}{y_w}$.

The co-occurrence frequency between a word pair w_1 and w_2, represented by $f_d(w_1, w_2)$, is also larger than the co-occurrence frequency between the two particular senses of w_1 and w_2. To estimate the co-occurrence frequency between the two senses we assume that w_1 and w_2 have strongest correlation only on the two particular senses and otherwise normal correlation. Normal correlation is the expected correlation between common English words and we denote it using PMI value k[5]. Therefore, the co-occurrence frequency between the two particular senses of w_1 and w_2 can be estimated by subtracting the co-occurrence frequency contributed by other combinations of senses, denoted by x, from the total co-occurrence frequency between the two words. More specifically,

$$f_d(w_1, w_2) - x$$

where x is computed using the definition of PMI by solving Equation 7.3.

$$\log\Big(\frac{x \cdot N}{f_{w_1} \cdot f_{w_2} - \frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}}}\Big) = k \tag{7.3}$$

Equation 7.3 amounts to asking the question – with a correlation degree of k, how many co-occurrences are expected among the remaining co-occurrence possibilities resulted by excluding the possibilities between the two senses of interest from the total possibilities.

Finally, the modified PMI, called PMI_{max}, between the two words w_1 and w_2 is given in Equation 7.4.

$$\begin{aligned}\mathrm{PMI}_{max}(w_1, w_2) &= \log\Big(\frac{(f_d(w_1, w_2) - x) \cdot N}{\frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}}}\Big) \\ &= \log\Bigg(\frac{\Big(f_d(w_1, w_2) - \frac{e^k}{N} \cdot (f_{w_1} \cdot f_{w_2} - \frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}})\Big) \cdot N}{\frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}}}\Bigg)\end{aligned} \tag{7.4}$$

PMI_{max} estimates the maximum correlation between two words, i.e., the correlation between their closest senses. In circumstances where we cannot know the particular senses used, it is reasonable to take the maximum similarity among all possible sense pairs as

[5] More explanations will be given in Section 7.3.4 after the k is empirically learned and exhibited in Table 7.5.

a measure of word similarity. For example, when sense information is unavailable, the shortest path assumption is often taken to compute word similarity in the WordNet-based measures. While the assumptions made in deriving the PMI_{max} may appear naive, we will demonstrate their effectiveness in later sections. More sophisticated models may lead to better results and we plan to explore this in future work.

7.3 Experiments

In this section, we determine values for the coefficients used in PMI_{max} by selecting ones that maximize performance in automatic thesaurus generation, where the core task is to find synonyms for a target word. Synonyms can be synonymous to the target word in their major or minor senses, and synonyms with more senses tend to be more difficult to find because they have relatively less semantic overlap with the target. Because more frequent words tend to have more senses, the synonyms under-weighted by PMI are often those with high frequency. We hope PMI_{max} can fix or alleviate this problem and find synonyms without a frequency bias.

Since word similarity is usually measured within the same part of speech (POS) (e.g., in [85, 64, 49]), we learn different coefficient sets for nouns, verbs, adjectives and adverbs. We further examine the learned relations between the frequency of a word with a particular POS and its number of senses within the POS (i.e., Equation 7.2) using knowledge from WordNet.

This section also serves as an evaluation of PMI_{max} on the task of automatic thesaurus generation. We start by describing our corpus, evaluation methodology, and the gold standard. Next we present and analyze the performance of basic PMI, PMI_{max}, and a state-of-the-art distributional similarity measure.

7.3.1 Corpus Selection

To learn coefficients, we prefer a balanced collection of carefully written and edited text. Since PMI is sensitive to noise and sparse data, a large corpus is required. The British National Corpus (BNC) is one of the most widely used corpora in text mining with more than 100 million words. Though large enough for distributional similarity based approaches, it is not of sufficient size for PMI.

Given these considerations, we selected the Project Gutenberg [43] English eBooks as our corpus. It comprises more than 27,000 free eBooks, many of which are well-known. A disadvantage of the collection is its age – most of the texts were written more than 80 years ago. Consequently, many new terms (e.g., “software”, “Linux”) are absent. We processed the texts to remove copyright statements and excluded books that are repetitions of similar entries (e.g., successive editions of the CIA World Factbook) or are themselves a thesaurus (e.g., Roget’s Thesaurus, 1911). We further removed unknown thesaurus-like data in the corpus with a classifier using a simple threshold based on the percent of punctuation characters. This simple approach is effective in identifying thesaurus-like data, which typically consists of sequences of single words separated by commas or semi-colons. Last, we performed POS tagging and lemmatization on the entire corpus using the Stanford POS tagger [98]. Our final corpus contains roughly two billion words, almost 20 times as large as the BNC corpus.

7.3.2 Evaluation Methodology

Various methods for evaluating automatically generated thesauri have been used in previous research. Some evaluate their results directly using subjective human judgments [46, 92] and others indirectly by measuring the impact on task performance [23].

Some direct and objective evaluation methodologies have also been proposed. The simplest is to directly compare the automatically and manually generated thesauri [36]. However, a problem arises in that current methods do not directly supply synonyms but rather lists of candidate words ranked by their similarity to the target. To enable automatic comparison, Lin [62] proposed first defining word similarity measures for hand crafted thesauri and then transforming them into the same format as the automatically generated thesaurus – a vector of words weighted by their similarities to the target. Finally cosine similarity is computed between the vectors from machine generated and hand crafted thesauri. However, it is unclear if such transformation is a good idea since it adds to the gold standard a large number of words that are not synonyms but related words, a deviation from the original goal of automatic thesaurus generation.

We chose a more intuitive and straightforward evaluation methodology. We use the recall levels in six different top n lists to show how high the synonyms of the head word in an entry in the "gold standard" occur in the automatically generated candidate list. The selected values for n are 10, 25, 50, 100, 200, and 500. A key application for automated thesaurus induction is to assist lexicographers in identifying synonyms, and we feel that a list of 500 candidates is the largest set that would be practical. We considered using Roget's Thesaurus or WordNet as the gold standard but elected not to use either. Roget's categories are organized by topic and include many related words that are not synonyms. WordNet, in contrast, has very fine-grained sense definitions for its synonyms, so relatively few synonyms can be harvested without exploiting hypernyms, hyponyms and co-hyponyms. Moreover, we wanted a gold standard thesaurus that is contemporaneous with our collection.

We chose Putnam's Word Book [32] as our gold standard. It contains more than 10,000 entries, each consisting of a head word, its part of speech, all its synonyms, and sometimes a few antonyms. The different senses (very coarse in general) of the head word are separated by semi-colons, and within each sense the synonyms are separated by commas, as in the following example.

> quicken, v. revive, resuscitate, animate; excite, stimulate, incite; accelerate, expedite, hasten, advance, facilitate, further

While the coverage of Putnam's Word Book synonyms is not complete, it is extensive, so that our recall metric should be close to the true recall. On the other hand, measuring true precision is difficult since the top n lists can contain proper synonyms which are not included by Putnam. The different top n lists will give us a rough idea about how "precision" varies with the recall. In addition, we supply another measure – the average rank over all the synonyms contained by a top n list. We give an example below to show how to compute the recall measures and average ranks. The top 50 candidates computed using basic PMI for the verb "quicken" are:

> exhilarate, invigorate, energize, regenerate, alert, pulse, slacken, ***accelerate***, deaden, begrudge, husk, recreate, cluck, constrict, ***stimulate***, intensify, career, stagnate, lag, throb, toughen, whinny, enliven, ***resuscitate***, retard, broaden, rejuvenate, rebuff, lather, sharpen, plummet, pulsate, nerve, dull, miscalculate, weld, sicken, infuse, shrill, blunt, heighten, distance, deepen, neigh, near, kindle, rouge, freshen, amplify, hearten

Only three synonyms for the verb "quicken" in the gold standard appear among the top 50 candidates: "accelerate", "stimulate", and "resuscitate" with ranks 8, 15, and 24. Thus, the recall values for top 10, top 25 and top 50 lists are 1/12, 3/12 and 3/12, respectively (12 is the total number of synonyms for the verb "quicken" in our gold standard). The average ranks for top 10, top 25 and top 50 lists are 8, 15.67 and 15.67, respectively.

We initially process Putnam's Word Book to filter out entries whose head words have a frequency less than $10,000$ in the Gutenberg corpus and further eliminate words with frequency less than 700 in the synonym lists of the remaining head words. In our experiment, we observed that many synonyms have PMI values slightly above 7.0 (see Table 7.10).

	All		Unique Sense		1st Synonym	
	Entries	Pairs	Entries	Pairs	Entries	Pairs
noun	2286	10994	1103	3762	886	886
verb	1187	6866	623	2515	559	559
adj.	1015	5944	579	2143	454	454
adv.	39	109	32	76	27	27

Table 7.1. The number of entries and synonym pairs for each POS category under our three scenarios.

Using the thresholds $10,000$ and 700 enables them to co-occur at least four or five times, which is typically required to ensure PMI, a statistical approach, works reasonably [17]. In addition, multi-words terms and antonyms were removed.

Words can have multiple senses and many synonyms are not completely synonymous but overlap in particular senses. We would like to evaluate PMI's performance in finding synonyms with different degrees of overlap. To enable this kind of evaluation, we tested using three scenarios: (i) all entries; (ii) entries with a unique sense; and (iii) entries with a unique sense and only using the first synonym[6]. Our rationale is that the single sense words should have a greater semantic overlap with their synonyms and moreover the largest semantic overlap with their first synonyms. Although we require the head words to have a unique sense, their synonyms may still be polysemous. Table 7.1 shows the number of entries and synonym pairs in three scenarios with different part of speech tags.

7.3.3 Performance of Basic PMI

In our basic PMI algorithm, word co-occurrences[7] are counted in a moving window of a fixed size that scans the entire corpus. To select the optimal window size for the basic PMI metric, we experimented with 14 sizes, starting at ± 5 and ending at ± 70 with a step

[6] We exclude the entries whose first synonyms have a frequency less than 700

[7] Our word co-occurrence matrix is based on a predefined vocabulary of more than 22,000 common English words and its final dimensions are 26,000 × 26,000 when words are POS tagged.

		T10	T25	T50	T100	T200	T500	Avg
all entries	noun	0.22	0.36	0.47	0.58	0.68	0.80	0.52
	verb	0.23	0.36	0.46	0.57	0.69	0.83	0.52
	adj.	0.26	0.40	0.52	0.63	0.74	0.86	0.57
	adv.	0.29	0.50	0.62	0.75	0.84	0.96	0.66
unique sense	noun	0.27	0.43	0.55	0.67	0.76	0.86	0.59
	verb	0.30	0.45	0.56	0.67	0.78	0.90	0.61
	adj.	0.32	0.46	0.59	0.70	0.80	0.91	0.63
	adv.	0.32	0.55	0.68	0.77	0.87	0.96	0.69
1st synonym	noun	0.32	0.50	0.63	0.75	0.83	0.91	0.66
	verb	0.38	0.55	0.66	0.76	0.85	0.93	0.69
	adj.	0.36	0.55	0.68	0.78	0.86	0.95	0.70
	adv.	0.44	0.59	0.78	0.85	0.93	0.96	0.76

Table 7.2. Recall for basic PMI using a ± 40 words window

of five words. Windows were not allowed to cross a paragraph boundary and we used a stop-word list consisting of only the three articles “a”, “an” and “the”.

We found that performance initially improves as the window size increases. However, the performance enters a plateau when the window size reaches ± 40, a window that corresponds to about four lines of text in a typically formatted book. We note that our optimal window size is slightly larger than the 16-32 size obtained by Terra and Clarke [96]. Data sparseness may account for the difference because Terra and Clarke used a much larger corpus (53 billion words) and therefore data sparseness was less severe in their case. The six recall levels of basic PMI and their average for the context window of ± 40 words are shown in Table 7.2. The average ranks are presented in Figure 7.2 for the four POS categories. Six markers on each line correspond to six top n lists. The x and y coordinates of a marker are the recall level and the average rank for the corresponding list. The average ranks for unique sense scenario are omitted due to space limitations. Note that in populating the ranked candidate list, we rule out the words whose frequency is under 700. This is consistent with the preprocessing we did for the gold standard.

		T10	T25	T50	T100	T200	T500	Avg
all entries	noun	0.29	0.44	0.56	0.66	0.75	0.85	0.59
	verb	0.33	0.47	0.58	0.68	0.77	0.88	0.62
	adj.	0.38	0.53	0.64	0.73	0.82	0.90	0.67
	adv.	0.49	0.67	0.77	0.84	0.89	0.98	0.77
unique sense	noun	0.36	0.53	0.64	0.74	0.82	0.90	0.67
	verb	0.43	0.57	0.69	0.77	0.85	0.93	0.71
	adj.	0.45	0.61	0.71	0.80	0.88	0.94	0.73
	adv.	0.53	0.73	0.83	0.87	0.92	0.98	0.81
1st synonym	noun	0.45	0.65	0.75	0.85	0.89	0.94	0.76
	verb	0.57	0.71	0.82	0.87	0.91	0.97	0.81
	adj.	0.58	0.74	0.84	0.90	0.96	0.98	0.83
	adv.	0.63	0.93	0.93	0.93	0.96	1.00	0.90

Table 7.3. Recall for PMI_{max} using ± 40 words window

The table and figures show that the “first synonym” category has better performance than the “unique sense” one, which is again better than the “all entries” category. We conclude that the synonyms with more semantic overlap have stronger correlation as measured by basic PMI. Among all POS tags, adverbs have the best performance, followed by adjectives and verbs, with nouns exhibiting the worst performance.

7.3.4 PMI_{max} Coefficient Tuning and Performance

A number of coefficients must be determined for PMI_{max}. We find their optimal values by maximizing a utility function based on the performance of automatic thesaurus generation. *The utility function is defined as the average of the recall levels in the six different top n lists*. The intuitive basis behind this is to improve recall in the six lists while giving emphasis to smaller lists since each list subsumes all its smaller-sized lists. Increasing recall in a fixed top n list typically results in the improvement of precision in the list. Therefore, this utility function measures precision as well. To see why, suppose our gold standard supplies the complete set of synonyms for a target word. Then the ratio

		T10	T25	T50	T100	T200	T500	Avg
all entries	noun	0.38	0.49	0.57	0.66	0.74	0.83	0.61
	verb	0.34	0.45	0.53	0.62	0.72	0.84	0.58
	adj.	0.37	0.48	0.58	0.66	0.75	0.86	0.62
	adv.	0.53	0.66	0.72	0.78	0.83	0.94	0.74
unique sense	noun	0.47	0.59	0.67	0.75	0.81	0.88	0.70
	verb	0.44	0.56	0.64	0.73	0.81	0.90	0.68
	adj.	0.45	0.57	0.66	0.74	0.82	0.91	0.69
	adv.	0.55	0.68	0.73	0.77	0.84	0.93	0.75
1st synonym	noun	0.62	0.74	0.80	0.84	0.88	0.91	0.80
	verb	0.58	0.70	0.77	0.83	0.90	0.95	0.79
	adj.	0.62	0.72	0.81	0.86	0.91	0.96	0.81
	adv.	0.67	0.78	0.81	0.89	0.93	0.96	0.84

Table 7.4. Recall for Distributional Similarity – PPMIC

between the recall and the precision in a fixed top n list is a constant. Though not complete, our gold standard can be thought as supplying a random sample of synonyms. Thus, the recall in a top n list can be used to estimate the true recall and thereby the constant ratio property should hold.

We use a constrained brute force search to maximize the utility function. The PMI_{max} coefficients (a, p, q and k) form a four dimensional search space that is computationally expensive to search. By averaging the number of senses extracted from WordNet over nouns, verbs, adjectives and adverbs with frequency around the minimum threshold (i.e., 700) respectively, we found that their mean values all fall into the range between one and two. So we make a simplifying assumption that words with frequency of 700 in our corpus have just one sense, reducing the search space to three dimensions. With this assumption, we can solve the coefficients a to be $\frac{1}{(\log(700)+q)^p}$. Then Equation 7.2 can be updated to

$$y_w = \frac{(\log(f_w) + q)^p}{(\log(700) + q)^p} \tag{7.5}$$

In exploring the three dimensional space, we let p be in the range [0..10] stepped by 0.5,

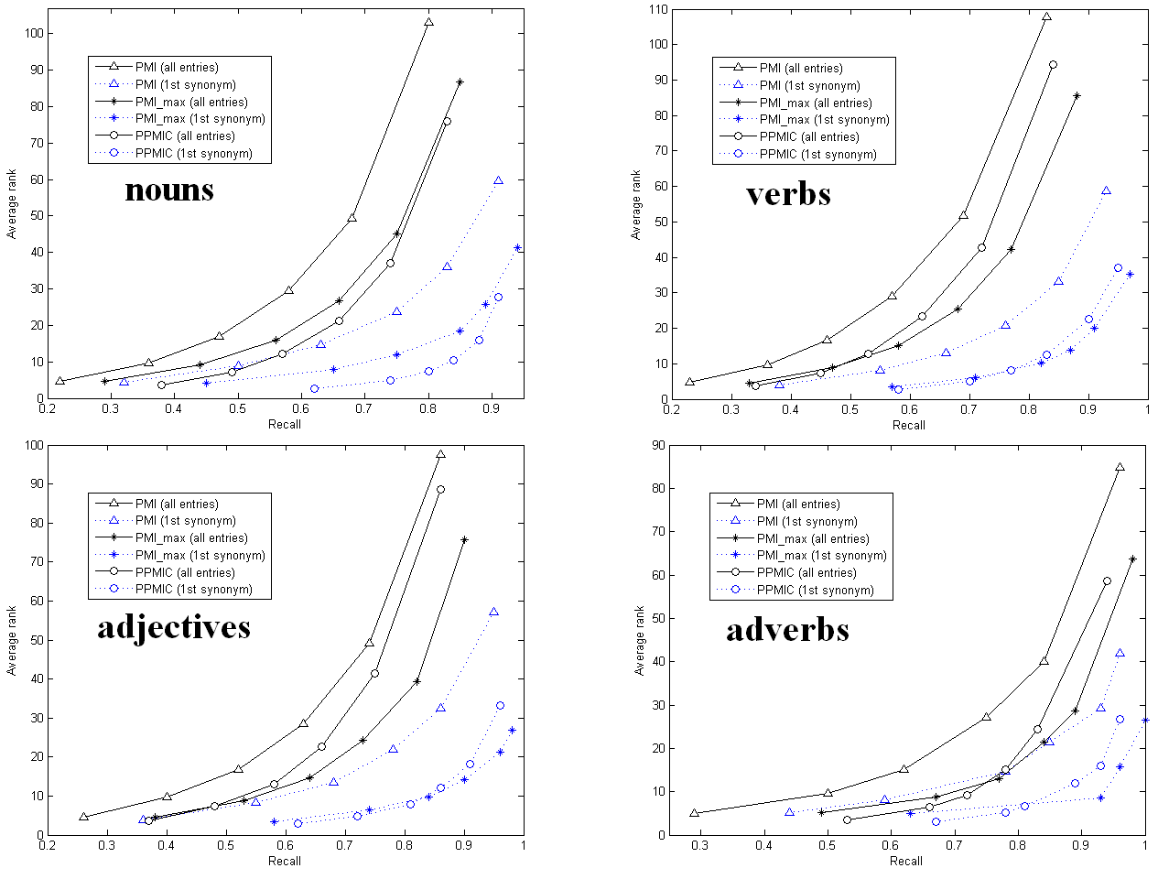


FIG. 7.2. Average ranks for nouns, verbs, adjectives and adverbs. The six marks on each line correspond to six top n lists and their x and y coordinates give the recall and the average rank.

which comprises 21 choices. To avoid searching in a continuous real range, we sample evenly spaced points. We choose the range [0..10] because we expect p to be positive and not a very high power. Similarly, we let q be in the range [-6..10] stepped by 1, yielding 17 choices. We set the left boundary as -6 because it is the smallest number that keeps $(\log(700) + q)$ positive and the right boundary to 10 because we don't expect the displacement to be large due to the large corpus that we use. We let e^k be in the range [0..100] stepped by 10, which has 11 values. This spans a range from a very weak correlation to fairly strong correlation under our setting of context window of ± 40 words. A total of 3927

	p	**q**	e^k	**k**
noun (1,2)	1.5	-4	30	3.4
verb (1,2)	1.5	-5	40	3.7
adj. (1)	2	-4	70	4.2
adj. (2)	5	4	70	4.2
adv.	3	0	40	3.7

Table 7.5. Optimal coefficients on three partitioned datasets and one intact dataset

candidate solutions are included in our search space.

In order to avoid overfitting, we randomly partitioned the Putnam entries under each POS, except adverbs[8], into two equal size datasets, resulting in three pairs of datasets for noun, verb and adjective and one single dataset for adverb. We then carried out 3927 experiments on each dataset separately, iterating and testing all the entries, and computing recall values for the six top n lists averaging over all entries, and finally calculating the utility function. All computations were based on the words co-occurrence matrix generated using the moving window of ± 40 words. The optimal coefficients are shown in Table 7.5. For nouns and verbs, the optimal coefficients on the two datasets are the same. In the case of adjectives, although the optimal coefficients are different, their curves generated by Equation 7.5 are close, which is illustrated in the adjective graph in Figure 7.4. The strong agreement on the pairs of datasets is not accidental. By sorting the 3927 solutions using their utility values, we find that the higher a solution appears, the closer its curve is to the optimal solution's curve. In other words, the curves tend to converge to the optimal one as their utilities increase. Note that two combinations of p and q, though may vary dramatically in individual p and q values, can yield close curves, as shown in the adjective graph in Figure 7.4.

With our experimental settings, PMI values as the Ks in Table 7.5 indicate a corre-

[8] We only had 39 adverbial entries and 109 synonym pairs in total, so we did not apply two-fold cross-validation to it.

lation which is close to "normal correlation" for their particular POS. If we compute PMI values for all possible word pairs (within the same POS) satisfying our filtering criteria and use them to draw a histogram, we observe a familiar bell distribution. "Normal correlation" is the center of the distribution. Note that according to the Equation 7.4, these Ks impose a lower bound on what PMI_{max} can compute. In many applications these lower bounds would not cause a problem because people are typically interested in correlations which are stronger than "normal correlation". For example, as illustrated in Table 7.10 for human similarity ratings, noun pairs holding PMI value around 3.4 would be judged as having no or very low similarity.

The performance of automatic thesaurus generation using two-fold cross-validation is shown in Table 7.3 and Figure 7.2. Although the coefficients are learned from the "all entries" scenario, the same coefficients are applied to generate the results for the "unique sense" and "first synonym" scenarios. As Table 7.3 shows, the recall values enjoy significant improvements over basic PMI for all of the scenarios, POS tags, and top n lists. Some recall values, for example, verbs in top 10 list as "first synonym", received a 50% increase. The improvements on all the recall values are statistically significant ($p < 0.001$ two-tailed paired t-test). Regarding average rank, the comparison should be based on the same recall level instead of the same top n list. This is because a list with larger recall may contain more synonyms with bigger ranks and therefore draw down the average rank. Figure 7.2 clearly shows that, at the same recall level, average ranks for PMI_{max} get across-the-board improvements upon basic PMI.

Compare PMI_{max}'s top 50 candidates for the verb "quicken" to those shown for PMI in Section 7.3.2.

> slacken, ***stimulate***, invigorate, regenerate, throb, ***accelerate***, intensify, exhilarate, kindle, deepen, deaden, sharpen, retard, enliven, near, awaken, ***revive***, thrill, heighten, overtake, pulse, broaden, stir, lag, sicken, infuse, expand, slow, brighten, dilate,

strengthen, dull, purify, refresh, ***hasten***, begrudge, spur, trot, career, nerve, freshen, hurry, blunt, sanctify, warm, cluck, inspire, lengthen, speed, impart

For this example, four synonyms for the word “quicken” appear in the top 50 candidate list and are marked as bold. Their ranks are 2, 6, 17 and 35, so the recall values for top 10, 25 and 50 lists are 2/12, 3/12 and 4/12 respectively. The average ranks for top 10, 25 and 50 lists are 4, 8.3 and 15. These numbers all improve upon the numbers using basic PMI. High frequency verbs like “spur”, “hurry”, and “speed”, which are not shown by basic PMI, also enter the ranking.

Figure 7.3 shows additional examples of synonym lists produced by PMI and PMI_{max}, displaying a word and its top 50 most similar words for each syntactic category. Words presented in both lists are marked as italic. The examples show that synonyms are generally ranked at the top places by both methods. However, antonyms can have rankings as high as synonyms because antonyms (e.g., “twist” and “straighten”) have many contexts in common and a single negation may reverse their meaning to one another. Among all POS examples, the noun “car” exhibits the worse performance. Nevertheless, synonyms and similar concepts for “car”, such as “automobile”, “limousine” and “truck” still show up in the top 50 lists of both PMI and PMI_{max}. The very top places of “chauffeur”, “garage” and “headlight” suggests that both measures highly rank a special kind of relation: nouns that are formed specifically for use in relation to the target noun. In the definitions of these nouns, the target noun is typically used. For example, “chauffeur” is defined in WordNet as “a man paid to drive a privately owned car”. The advantage of these nouns in computing PMI similarity comes from the fact that they seldom have their own contexts which are not subsumed by the contexts of the target noun. PMI similarity is context-based, which means that the similarity is not solely relied on “IS-A” relation but an overall effect of all kinds of relations embodied by the contexts in a corpus.

The examples show that PMI and PMI_{max} capture almost the same kind of semantic

(PMI) **twist_VB**: *contort*, *squirm*, *writhe*, *knot*, *twine*, *twirl*, *wriggle*, *warp*, card, *coil*, *wrench*, *loop*, *dislocate*, *braid*, interlock, untangle, snake, *distort*, dent, wiggle, *grimace*, unwind, mildew, slump, *stem*, flex, *splinter*, *crook*, thud, *stunt*, groove, *tangle*, *claw*, *mat*, hunch, lunge, *char*, *curl*, unhook, joint, clamp, blotch, crochet, constrict, rotate, sprain, lacquer, vein, *fork*, *protrude*

(PMI$_{max}$) **twist_VB**: *writhe*, *knot*, *squirm*, *contort*, *twine*, *wriggle*, *twirl*, *warp*, *coil*, *wrench*, *distort*, *curl*, *crook*, *stem*, round, *tangle*, *braid*, *grimace*, wind, spin, fasten, *loop*, jerk, *stunt*, *dislocate*, weave, curve, double, clutch, *protrude*, tie, screw, tear, *splinter*, bend, *mat*, hook, tug, crumple, *claw*, wrinkle, grip, roll, tighten, dangle, *char*, straighten, *fork*, pull, strangle

(PMI) **car_NN**: *garage*, *trolley*, *headlight*, *chauffeur*, *limousine*, *motorist*, *motor*, *siding*, *locomotive*, *caboose*, *subway*, *freight*, *automobile*, *conductor*, motorcycle, *driveway*, *axle*, *radiator*, *brake*, *throttle*, speeding, *uptown*, *curb*, *auto*, skid, *balloon*, *truck*, refrigerator, *driver*, downtown, parachute, *gasoline*, *steering*, spin, mileage, *passenger*, *racing*, *train*, *engine*, purr, suitcase, chute, tractor, *taxi*, *railroad*, traction, goggles, *elevator*, toot, standstill

(PMI$_{max}$) **car_NN**: *chauffeur*, *garage*, *motor*, *trolley*, *locomotive*, *conductor*, *automobile*, *limousine*, *freight*, *headlight*, *train*, *driver*, *brake*, *siding*, *passenger*, *engine*, *balloon*, *railroad*, *curb*, *axle*, wheel, *truck*, *motorist*, *auto*, *driveway*, *subway*, platform, track, *caboose*, speed, tire, compartment, *radiator*, baggage, depot, rail, *steering*, *elevator*, seat, shaft, station, *racing*, vehicle, *taxi*, *gasoline*, *throttle*, occupant, ambulance, *uptown*, road

(PMI) **ridiculous_JJ**: *nonsensical*, *laughable*, *puerile*, *absurd*, *preposterous*, *contemptible*, *sublime*, *ludicrous*, *grotesque*, *farcical*, *bizarre*, *melodramatic*, *insipid*, snobbish, *pedantic*, *comic*, *impertinent*, *incongruous*, *indecent*, *comical*, *odious*, panicky, *wasteful*, grandiose, *idiotic*, inane, *conceited*, *disgusting*, *satirical*, sobering, outlandish, narrow-minded, *despicable*, unreliable, stilted, messy, good-humored, *paltry*, irreverent, *extravagant*, regrettable, *degrading*, *humiliating*, humdrum, *pompous*, *frivolous*, exasperating, antiquated, *silly*, dowdy

(PMI$_{max}$) **ridiculous_JJ**: *absurd*, *sublime*, *contemptible*, *preposterous*, *grotesque*, *ludicrous*, *laughable*, *puerile*, *comic*, *nonsensical*, *odious*, *impertinent*, foolish, *silly*, *extravagant*, *comical*, *bizarre*, *incongruous*, childish, *insipid*, *disgusting*, *indecent*, *conceited*, vulgar, monstrous, fantastic, *idiotic*, *frivolous*, pathetic, *pedantic*, *satirical*, stupid, *paltry*, *melodramatic*, humorous, awkward, sentimental, *humiliating*, exaggerated, trivial, *farcical*, *pompous*, amused, *wasteful*, *despicable*, serious, *degrading*, senseless, funny, tragic

(PMI) **commonly_RB**: *supposedly*, *incorrectly*, *customarily*, *erroneously*, *technically*, *conventionally*, *ordinarily*, chemically, psychologically, traditionally, *infrequently*, credibly, sexually, predominantly, *popularly*, *improperly*, currently, *rarely*, *locally*, *seldom*, *usually*, mistakenly, *sparingly*, *generally*, legitimately, *frequently*, experimentally, *universally*, preferably, capriciously, *relatively*, hugely, rationally, *variously*, *exclusively*, vertically, negligently, *annually*, *habitually*, philosophically, fourthly, correspondingly, extensively, rigorously, *sometimes*, *necessarily*, *chiefly*, *invariably*, primarily, symmetrically

(PMI$_{max}$) **commonly_RB**: *generally*, *usually*, *frequently*, *seldom*, *rarely*, *sometimes*, *ordinarily*, often, *erroneously*, most, *technically*, *supposedly*, more, *universally*, *chiefly*, *incorrectly*, *exclusively*, especially, less, *necessarily*, *infrequently*, namely, *popularly*, *relatively*, *annually*, occasionally, hence, *invariably*, therefore, also, *customarily*, either, properly, likewise, *habitually*, widely, largely, formerly, very, *improperly*, comparatively, *variously*, *conventionally*, particularly, *sparingly*, however, *locally*, strictly, principally, equally

FIG. 7.3. Four pairs of examples, showing the top 50 most similar words by PMI and PMI_max

relations and many of the words in their top 50 lists are the same. Their key difference is how they rank low frequency and high frequency words. PMI is predisposed towards low frequency words and PMI_{max} alleviates this bias. For example, the topmost candidates for “twist”, “ridiculous” and “commonly” produced by PMI are “contort”, “nonsensical” and “supposedly” respectively, which are less common than the corresponding words generated by PMI_{max}, “writhe”, “absurd” and “generally”, although they have close meanings. The overall adjustment made by PMI_{max} is that low frequency words move down the list and high frequency words move up with the constraint that more similar words are still ranked higher. Low frequency words at the top of the PMI list are often good synonyms. Though moved down, they typically remain in the PMI_{max} list. Low frequency words which are more lowly ranked tend to be not very similar to the target; in the PMI_{max} list, they are replaced with higher frequency words.

In the example of “twist”, high frequency words (e.g., “round”, “wind”, “spin”, “fasten”, “jerk”, “weave”, “curve”, “double” and “bend”) move in the list to replace low frequency words including noisy words (e.g., “blotch” and “lacquer”), less similar words (e.g., “groove” and “lunge”), and less commonly used similar words (e.g., “flex” and “crochet”). In the example of “car”, two important similar words “vehicle” and “wheel” appear in the PMI_{max} list after the adjustment.

Because PMI_{max} estimates semantic similarity between the closest senses of two words, it has an advantage over PMI in discovering polysemous synonyms or similar words. As examples, “double” has a sense of *bend* and PMI_{max} find it similar to “twist”; “platform” can mean vehicle carrying weapons and PMI_{max} find it similar to “car”; “pathetic” has a sense of *inspiring scornful pity* and PMI_{max} find it similar to “ridiculous”.

The frequency counts of the verb “double”, noun “platform” and adjective “pathetic” in our Gutenberg corpus are 32385, 60114 and 28202 and their assumed senses (according to Equation 7.5 and Table 7.5) are 6.5, 4.5 and 6.0 respectively. They are ranked at

106th , 64th and 108th places in their PMI lists. They are able to move up in the PMI_{max} lists because they have more assumed senses than many words in the PMI lists. Here we zoom in on a concrete example that compares a moving-out word “blotch” to a moving-in word “double”. “Blotch” has a frequency count 1149 and co-occurs with “twist” 18 times, producing a PMI value 6.35. “Double” has a lower PMI value, 5.98, and co-occurs with “twist” 349 times. However, since “blotch” only has 1.5 assumed senses its PMI_{max} value is 8.71, which is smaller than the PMI_{max} value between “double” and “twist”, 9.75.

The PMI_{max} list for the adverb “commonly” has some words that are often seen in a stop words list, such as “more”, “less”, “hence”, “also”, “either”, “very” and “however”. These words have very high frequencies but few senses. PMI_{max} erroneously judges them similar to “commonly” because their high frequency mistakenly suggests many senses.

7.3.5 Comparison to Distributional Similarity

To demonstrate the efficacy of PMI_{max} in automatic thesaurus generation, we compare it with a state-of-the-art distributional similarity measure proposed by Bullinaria and Levy [12]. Their method achieved the best performance after a series of work on distributional similarity from their group [79, 58, 57]. The method is named Positive PMI components and Cosine distances (PPMIC) because it uses positive pointwise mutual information to weight the components in the context vectors and standard cosine to measure similarity between vectors. Bullinaria and Levy demonstrated that PPMIC was remarkably effective on a range of semantic and syntactic tasks, achieving, for example, an accuracy of 85% on TOEFL synonym test using the BNC corpus. Using PMI as the weighting function, and cosine as the similarity function is a popular choice for measuring distributional similarity [78]. What makes PPMIC different is a minimal context window size (± 1 word window) and the use of a high dimension context vector that does not remove of low frequency components. Bullinaria and Levy found that these uncommon settings are essential to make PPMIC work extremely well, though they are not generally good choices for other similarity measures.

Our PPMIC implementation differs from Bullinaria and Levy's in using lemmatized and POS-tagged words (noun, verb, adjectives and adverbs) rather than unprocessed words as vector dimension. This variation is simply due to convenience of reusing what we already have. We tested our implementation of PPMIC on TOEFL synonym test and obtained a score of 80%. The slightly lower performance may result from the variation we made or the use of the outdated Gutenberg corpus to answer questions about modern English. Nevertheless, 80% is a very good score on TOEFL synonym test. As an example, Bullinaria and Levy's previous best result, before inventing PPMIC, was 75% [57].

The performance of PPMIC in the automatic thesaurus generation is shown in Table 7.4 and Figure 7.2. As we did for PMI and PMI_{max}, we exclude words with frequency less than 700 or with different POS from the target word in the candidate list. Unlike PMI and PMI_{max}, PPMIC has very good performance on nouns, which is even slightly better than verbs and adjectives. Unsurprisingly, PPMIC outperforms PMI on almost all the recall values and average ranks. The improvements on the recall values are statistically significant ($p < 0.001$ two-tailed paired t-test). When comparing with PMI_{max}, PPMIC has obvious advantage on nouns but just competing performance on other POS categories. Although PPMIC leads PMI_{max} on recall values for the small top n lists, such as top-10, PMI_{max} can generally catch up quickly and outrun PPMIC for the subsequent longer lists. The same trend can also be observed on average ranks depicted in Figure 7.2. When considering all the recall values in Table 7.3 and Table7.4, PMI_{max} has a significantly better performance than PPMIC ($p < 0.01$ two-tailed paired t-test). Compared with PMI_{max}, PPMIC seems to be able to rank a portion of synonyms very highly but it fails to give high scores to the remaining synonyms.

	noun	verb	adj.	adv.
PMI	0.133	0.148	0.161	0.155
PMI_{max}	0.173	0.219	0.242	0.224
PPMIC	0.283	0.255	0.276	0.374

Table 7.6. MAP values of PMI, PMI_{max} and PPMIC

7.3.6 Mean Average Precision Evaluation

Mean Average Precision (MAP) is a common measure used to evaluate systems in information retrieval (IR) tasks. Automatic thesaurus generation can be evaluated as an IR task if we make an analogy between an IR query and the need to identify synonyms for a target word (in this analogy correct synonyms are the relevant documents). We compare PMI, PMI_{max} and PPMIC using MAP in Table 7.6. PMI_{max} is significantly better than PMI ($p < 0.01$ two-tailed paired t-test). PPMIC numerically outperforms PMI_{max} but the improvements are not statistically significant ($p > 0.05$ two-tailed paired t-test). A higher average precision does not necessarily entail a lower average rank. For example, suppose system A ranks three synonyms of a target word at 1st, 9th and 10th places and system B ranks them at 3rd, 5th and 6th places. A's average precision of 0.51 is higher than B's 0.41, but B's average rank is 4.67, which is smaller than A's, 6.67.

PPMIC's higher MAP value results from its much better performance in placing synonyms at the very top ranks. The top-1 precision of PMI, PMI_{max} and PPMIC are supplied in Table 7.7. PPMIC has excellent precision, considering our gold standard only provides a subset of synonyms. However, Tables 7.6 and 7.7 again imply that PPMIC's performance degrades faster than that of PMI_{max} in finding more synonyms. Typically lexicographers desire high recall, therefore, the higher MAP scores for PPMIC do not necessarily make it a more compelling approach for lexicography than PMI_{max}.

	noun	verb	adj.	adv.
PMI	0.120	0.160	0.163	0.103
PMI_{max}	0.168	0.256	0.261	0.179
PPMIC	0.433	0.442	0.436	0.487

Table 7.7. Top-1 Precision of PMI, PMI_{max} and PPMIC

7.3.7 Examining the Assumptions in PMI_{max}

The key assumption made for PMI_{max} is found in Equation 7.5. In Section 7.3.4 we determined coefficients by maximizing the performance of PMI_{max} for automatic thesaurus generation. Note that the function is learned using an automated statistical approach. We will examine this function by comparing it with knowledge from human judgments extracted from WordNet.

Since we learned a different combination of coefficients for each POS category, we examine them separately. For each POS-tagged word with frequency above 700 in our corpus, we get its number of senses within its POS from WordNet and group them into noun, verb, adjective, and adverb categories, resulting in 8467 nouns, 3705 verbs, 3763 adjectives and 1095 adverbs. We generated four scatter plots using the natural logarithm of the frequency of a POS-tagged word as x-axis and its number of senses as y-axis. Matlab's non-linear least squares problem solver (LSQCURVEFIT) was used to find coefficients p, q that best fit Equation 7.5 to the scatter plots, producing the results in Figure 7.4. For nouns, the automatically learned function is almost identical to the best-fit function and for verbs they are quite close. For adjectives and adverbs, both the learned functions have steeper slope than the corresponding best fit functions. The noun class probably enjoys the best match because the assumptions used in deriving PMI_{max} work best for nouns and worst for adjectives and adverbs.

The closeness between the learned functions and the best-fit functions suggests that fitting Equation 7.5 could be an alternative way to determine the coefficients of PMI_{max}.

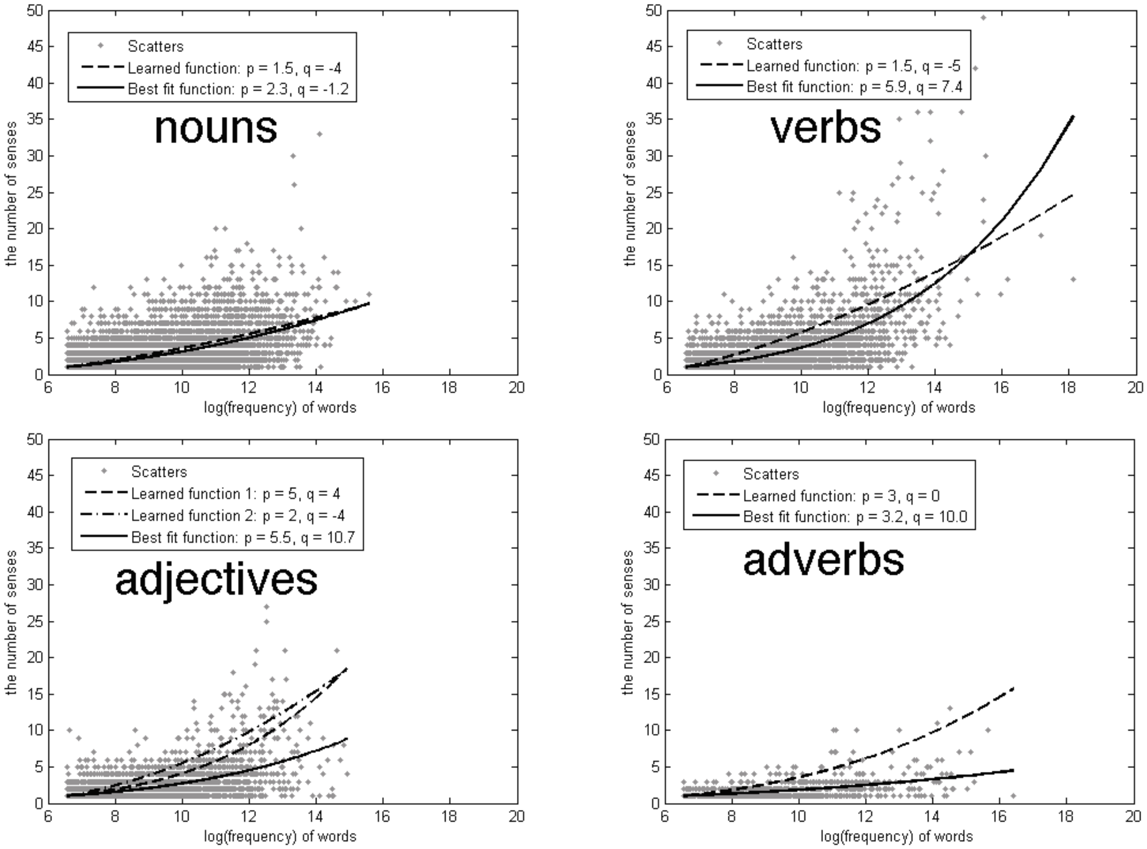


FIG. 7.4. The frequency vs the number of senses for nouns, verbs, adjectives and adverbs.

To see how this approach performs, we also give its recall values in the six top n lists for the "all entries" scenario in Table 7.8. Nouns and verbs have almost the same performance as in Table 7.3. Even adjectives and adverbs have close performance to their counterparts in Table 7.3. It shows that for adjectives and adverbs, most of the performance gain is achieved by changing the horizontal line $y = 1$ (i.e., the "single sense" assumption) to the places of the best-fit functions.

In deriving PMI_{max} in Section 7.2.2, we implicitly assumed that different senses of a word are not similar. However, it has been argued that WordNet's senses are too fine-grained [76], that is, different WordNet senses of the same word can be similar or highly

		T10	T25	T50	T100	T200	T500	Avg
all entries	noun	0.29	0.44	0.56	0.66	0.75	0.85	0.59
	verb	0.33	0.46	0.57	0.67	0.77	0.88	0.61
	adj.	0.37	0.52	0.63	0.73	0.81	0.89	0.66
	adv.	0.43	0.64	0.71	0.83	0.86	0.96	0.74

Table 7.8. Recall for PMI_{max} using coefficients in best-fit functions

correlated. According to this, the learned functions should have a lower slope than the best-fit functions, which is inconsistent with our results.

This is probably because our frequency-sense model is too simple and something is not very accurately modeled. Another possibility is that some other factors, which are irrelevant to senses, also contribute to the frequency bias of PMI. To counteract that part of bias, PMI_{max} could yield a steeper slope than what word polysemy requires. Other causes may possibly exist, such as different sets of words used in the automatic thesaurus generation experiment and in creating the plots, or missing senses in WordNet.

7.3.8 Low Frequency Words Evaluation

We learned coefficients for PMI_{max} using the occurrence frequency thresholds $10,000$ and 700 for target words and their synonyms. Now we would like to check if PMI_{max} learned in this way could produce consistently better results than PMI for low frequency target words. To this end, we extracted Putnam entries whose head words have a frequency between 700 and 2,000, obtaining 238, 103 and 155 entries and 598, 282 and 449 synonym pairs for nouns, verbs and adjectives, respectively. The recall values at six top n lists ("all entries" scenario) for PMI, PMI_{max} and PPMIC are shown in Table 7.9. Interestingly, PPMIC's performance for low frequency words is even better than its performance for high frequency words, as compared to Table 7.4. More data typically leads to more accurate results in statistical approaches. PPMIC's unusual result implies that its performance is

		T10	T25	T50	T100	T200	T500	Avg
PMI	noun	0.26	0.36	0.45	0.58	0.66	0.78	0.52
	verb	0.19	0.28	0.41	0.53	0.62	0.75	0.46
	adj.	0.23	0.35	0.42	0.57	0.74	0.87	0.53
PMI_{max}	noun	0.32	0.44	0.53	0.63	0.70	0.80	0.57
	verb	0.30	0.40	0.46	0.55	0.64	0.75	0.52
	adj.	0.31	0.42	0.54	0.65	0.74	0.87	0.59
PPMIC	noun	0.48	0.61	0.67	0.74	0.81	0.89	0.70
	verb	0.42	0.53	0.62	0.66	0.72	0.84	0.63
	adj.	0.43	0.56	0.66	0.74	0.81	0.88	0.68

Table 7.9. Recall for Low Frequency Words

also affected by the degree of word polysemy. The case of low frequency words is easier for PPMIC because they only have a few senses.

On the contrary, PMI and PMI_{max} have degraded performances compared to Tables 7.2 and 7.3 due to insufficient data for them to work reasonably. In our experiment, the expected co-occurrence between two strong synonyms (PMI value 8.0) with the frequencies 700 and 1,000 is only *one*. Thus, many low frequency synonyms lack even a single co-occurrence with their targets while many noisy, low frequency words are ranked at top places by chance. PMI_{max} produces consistently better results than PMI ($p < 0.001$ two-tailed paired t-test) but the improvement rate is generally lowered, especially for large top n lists. We see three reasons for this: some low frequency synonyms cannot be found regardless of the length of the ranked list because they have no co-occurrences with the target, PMI_{max} ranks noisy words downward but does not remove them in the relatively long lists, and the coefficients are not optimized for low frequency target words.

7.4 Benchmark Evaluation

In the preceding section, we demonstrated the effectiveness of PMI_{max} for automatic thesaurus generation. In this section, we will evaluate PMI_{max}, tuned for the automatic

Word Pair	Miller-Charles	Resnik [85]	J & C [49]	Lin [64]	Li et al [59]	CODC [16]	WebSVM [9]	PMI	PMI_{max}	PPMIC
car-automobile	3.92	8.0411	30	1	1	0.4229	0.98	7.570	10.498	0.392
gem-jewel	3.84	14.929	30	1	1	0.353	0.686	7.985	10.778	0.447
journey-voyage	3.84	6.7537	27.497	0.89	0.779	0.2666	0.996	5.336	8.567	0.477
boy-lad	3.76	8.424	25.839	0.85	0.778	0.2828	0.974	5.581	9.168	0.527
coast-shore	3.7	10.808	28.702	0.93	0.779	0.2923	0.945	6.606	10.039	0.511
asylum-madhouse	3.61	15.666	28.138	0.97	0.779	0.1845	0.773	8.016	9.614	0.102
magician-wizard	3.5	13.666	30	1	0.999	0.2076	1	8.008	10.242	0.295
midday-noon	3.42	12.393	30	1	1	0.2994	0.819	6.417	9.041	0.301
furnace-stove	3.11	1.7135	17.792	0.18	0.585	0.1982	0.889	6.935	9.585	0.310
food-fruit	3.08	5.0076	23.775	0.24	0.17	0.2355	0.998	5.970	9.388	0.337
bird-cock	3.05	9.3139	26.303	0.83	0.779	0.2295	0.593	6.567	9.607	0.287
bird-crane	2.97	9.3139	24.452	0.67	0.472	0	0.879	7.119	9.853	0.245
implement-tool	2.95	6.0787	29.311	0.8	0.778	0.2506	0.684	8.689	10.128	0.070
brother-monk	2.82	2.9683	19.969	0.16	0.779	0.1956	0.377	4.824	7.980	0.250
brother-lad	1.66	2.9355	20.326	0.2	0.355	0.1811	0.344	4.421	7.613	0.256
car-journey	1.16	0	17.649	0	0	0.2049	0.286	4.969	8.201	0.124
monk-oracle	1.1	2.9683	18.611	0.14	0.168	0	0.328	4.639	7.038	0.151
food-rooster	0.89	1.0105	17.657	0.04	0	0	0.06	4.721	7.019	0.064
coast-hill	0.87	6.2344	25.461	0.58	0.366	0	0.874	5.198	8.545	0.362
forest-graveyard	0.84	0	14.52	0	0.132	0	0.547	4.859	7.337	0.209
monk-slave	0.55	2.9683	20.887	0.18	0.35	0	0.375	3.798	6.026	0.238
coast-forest	0.42	0	15.538	0.16	0.17	0.1686	0.405	5.100	8.352	0.296
lad-wizard	0.42	2.9683	20.717	0.2	0.355	0	0.22	4.275	6.521	0.124
chord-smile	0.13	2.3544	17.535	0.2	0	0	0	4.436	7.012	0.178
glass-magician	0.11	1.0105	17.098	0.06	0	0	0.18	4.894	7.632	0.075
noon-string	0.08	0	12.987	0	0	0	0.018	3.757	5.742	0.081
rooster-voyage	0.08	0	12.506	0	0	0	0.017	3.679	4.919	0.044
Correlation Coef.	**1**	**0.791**	**0.836**	**0.834**	**0.883**	**0.837**	**0.847**	**0.796**	**0.856**	**0.654**

Table 7.10. Comparisons of PMI_{max} with PMI, PPMIC and previous measures on the MC dataset

thesaurus generation task, on common benchmark datasets including the Miller-Charles (MC) dataset [73] and the TOEFL synonym test [103], allowing us to compare our results with previous systems. In these datasets, there are words which occur less often in the Gutenberg corpus than our frequency cutoff of 700. For these words, we assume a single sense and apply Equation 7.4 to compute PMI_{max}.

7.4.1 Human Similarity Ratings

The widely used MC dataset [85, 49, 64, 59, 108, 16, 9] consists of 30 pairs of nouns rated by a group of 38 human subjects. Each subject rates “similarity of meaning” for each pair on a scale from 0 (no similarity) to 4 (perfect synonymy). The mean of the individual ratings for each pair is used as its semantic similarity. The MC dataset was taken from Rubenstein-Goodenough’s [88] original data for 65 word pairs by selecting ten pairs of high, intermediate, and low levels of similarity respectively. Although the MC experiment was performed 25 years after Rubenstein-Goodenough’s, the Pearson correlation coefficient between the ratings is 0.97. Four years later, the same experiment was replicated again by Resnik [85] with ten subjects. Their mean ratings also had a very high correlation coefficient of 0.96. Resnik also showed that the individual ratings in his experiment have an average correlation coefficient 0.8848 with the MC mean ratings.

Due to the absence of the word “woodland” in earlier versions of WordNet, only 28 pairs are actually adopted by most researchers. Our PMI implementation has a similar problem in that “implement” has very low frequency (117) in the Gutenberg corpus so that there are no co-occurrences for “implement” and “crane”. Since their PMI value is undefined, only 29 pairs can be used for our experiment. For these, the correlation coefficient between the MC and PMI ratings is 0.794, and for MC and PMI_{max} is 0.852. However, in order to compare with other people’s results, it is necessary to have an equal setting. Fortunately, the published papers of most previous researchers included lists of ratings for each pair, allowing the comparison of our work over 27 pairs as shown in Table 7.10.

The six previous measures in Table 7.10 give the best results that we can find in the literature on the MC dataset. Among them, Resnik [85], Jiang and Conrath [49], Lin [64], and Li et al [59] are WordNet-based approaches while CODC [16] and WebSVM [9] are Web-based, making use of page counts and snippets. PMI_{max} is the first corpus-based approach that enters the same performance level as other best approaches with a score 0.856, a 7.5% improvement over basic PMI. Although even basic PMI has a decent performance 0.796, PPMIC only obtains a correlation 0.654 which does not match its performance in automatic thesaurus generation or TOEFL synonym test. However this is not new. In an intensive study of distributional similarity, Weeds [108] applied ten different distributional measures, based on the BNC corpus, to the MC dataset with a top correlation coefficient of 0.62. The value of distributional similarity appears to vary significantly with the target words selected. Although it can generate a properly ranked candidate list, the absolute similarity value is not consistent across different target words selected.

We compare PMI_{max} with basic PMI on two other standard datasets: Rubenstein-Goodenough (RG) [88] and WordSim353 [31]. Among 65 word pairs in the RG dataset we removed five with undefined PMI and PMI_{max} values, and performed the experiment on the rest. WordSim353 contains 353 word pairs, each of which is scored by 13 to 16 human subjects on a scale from 0 to 10. We removed proper nouns, such as “FBI” and “Arafat”, because they are not supported by our current implementation of PMI and PMI_{max}. We also imposed a minimum frequency threshold of 200 to WordSim353 because it contains many modern words such as “seafood” and “Internet” which occur infrequently in our corpus. Choosing 200 as the threshold is a compromise between keeping most of the word pairs available and making PMI and PMI_{max} perform reliably. After all the preprocessing, 289 word pairs remains. The comparison results of PMI, PMI_{max} and PPMIC on the two datasets along with the MC dataset are shown in Table 7.11. Both the Pearson correlation and Spearman rank correlation coefficients are investigated.

While PMI_{max}’s Pearson correlation on the RG dataset is lower than its on the MC dataset, its Spearman rank correlation on the RG dataset is higher. WordSim353 is a more challenging dataset than MC and RG. Our results, though lower than those on MC and RG, are still very good [31]. The improvements of PMI_{max} over basic PMI using either Pearson

	MC(27)	RG(60)	WS353(289)
PMI (Pearson)	0.796	0.791	0.570
PMI_{max} (Pearson)	0.856	0.818	0.625
PPMIC (Pearson)	0.654	0.707	0.381
PMI (Spearman)	0.784	0.790	0.635
PMI_{max} (Spearman)	0.839	0.844	0.666
PPMIC (Spearman)	0.703	0.705	0.353

Table 7.11. Comparing PMI_{max} with PMI on three datasets

correlation or Spearman rank correlation are all statistically significant ($p < 0.05$ with two-tailed paired t-test). Both PMI and PMI_{max} have a consistently higher performance than PPMIC on the three datasets.

7.4.2 TOEFL Synonym Questions

Eighty synonym questions were taken from the Test of English as a Foreign Language (TOEFL). Each is a multiple choice synonym judgment, where the task is to select from four candidates the one having the closest meaning to the question word. Accuracy, which is the percentage of correctly answered questions, is used to evaluate performance. The average score of foreign students applying to US colleges from non-English speaking countries was 64.5% [55].

This TOEFL synonym dataset is widely used as a benchmark for comparing the performance of computational similarity measures [55, 103, 57, 96, 84, 77, 12]. Currently, the best results from corpus-based approaches are achieved by LSA [84], PPMIC [12] and PMI-IR [96], with scores of 92%, 85% and 81.25%, respectively.

Since the words used in our implementations of PMI, PMI_{max} and PPMIC are POS tagged, we first assign a POS tag to each TOEFL synonym question by choosing the common POS of the question and candidate words. The results on the TOEFL synonym test for PMI, PMI_{max}, and PPMIC are 72.5%, 76.25% and 80%, respectively. Although the

Gutenberg corpus has two billion words, it is still small compared with Web collections used by PMI-IR. Thus, unlike PMI-IR, data sparseness is still a problem limiting the performance of PMI_{max}. For example, there are three "no answer" questions[9] for which the question word has no co-occurrence with any of four candidate words. It is known that TOEFL synonym questions contain some infrequently used words [12]. In addition, some TOEFL words common in modern English, such as "highlight" and "outstanding", were uncommon at the time of Gutenberg corpus. 76.25% is an encouraging result because it demonstrates that PMI_{max} need not rely on search engines to be effective in a difficult semantic task like the TOEFL synonym questions.

7.5 Discussion and Application

PMI_{max} can be used in various applications that require word similarity measures. However, we are more interested in combining PMI_{max} with distributional similarity in the area of semantic acquisition from text because this direction is not yet explored. We start our discussion by looking at an example, the top 50 most similar words for the noun "car" generated by PPMIC.

> train, automobile, boat, wagon, carriage, engine, vehicle, motor, truck, coach, cab, wheel, ship, machine, cart, locomotive, chariot, canoe, vessel, craft, horse, bus, auto, driver, sleigh, gun, launch, taxi, buggy, barge, yacht, ambulance, passenger, freight, box, round, plane, trolley, station, team, street, track, window, rider, chair, mule, elevator, bicycle, door, shaft

The example shows that, as a state-of-the-art distributional similarity, PPMIC has an amazing ability to find the concepts that are functionality or utility similar to "car". These concepts, such as "train", "boat", "carriage", "vehicle", are typically neighboring concepts

[9] They are treated as incorrectly answered questions.

of “car” in a taxonomy structure such as WordNet. In contrast, PMI_{max}, as illustrated in the “car” example in Figure 7.3, can only find synonymous concepts and “siblings” concepts (e.g. “train” and “truck”) but miss the “cousin” concepts (e.g. “boat” and “carriage”). This is because PMI similarity measures the degree that two concepts tend to co-occur over the *exactly same* contexts. “boat” and “carriage” are missing because only a small proportion of contexts of “car” relates it to watercraft or archaic vehicles. Seemingly not as powerful as distributional similarity, PMI similarity provides a very useful complement to it.

There are many potential applications. Below we suggest two directions related to automatic thesaurus or ontology generation. First, we could devise an improved approach for high-precision automatic thesaurus generation by taking intersection of the candidate lists generated by PMI_{max} and a state-of-the-art distributional similarity, such as PPMIC. For example, the intersection of two top-50 candidate lists generated by PMI_{max} and PPMIC for “car” includes:

> train, automobile, engine, vehicle, motor, truck, wheel, locomotive, auto, driver, taxi, ambulance, passenger, freight, trolley, station, track, elevator, shaft

Many inappropriate distributional similar terms like “boat” and correlational similar terms like “chauffeur” are filtered out by the intersection. This makes synonyms, for example “auto”, have higher ranks in the resulting candidate list than either of the parent lists.

The second application is more interesting and challenging. We know that distributional similarity can find neighboring concepts of the target word in a taxonomy. A subsequent question on the course is how we could classify these concepts into different clusters corresponding to the “sibling”, “parent” and “cousin” sets in a taxonomy. This is an largely unsolved problem in ontology learning from text that the combination of PMI_{max} and distributional similarity can help address.

For example, of the 50 most distributional similar words of “car”, which are most likely to be classified together with “boat”? A simple approach is to take the intersection of two top-50 candidate lists generated by PPMIC and PMI_{max} for “car” and “boat” respectively[10]. The results for “boat” and several other examples obtained this way are shown in Figure 7.5. The words that can be classified with “boat” include its “sibling” concepts (conveyances on water) and “parent” concepts (vessel, craft) but no “cousin” concepts (conveyances on land). Similarly, the intersection list for “carriage” includes archaic vehicles and rejects modern ones. Although in the example of “chariot”, “car” appears in the list, it is due to a different sense of “car”[11]. This shows that polysemy of words can add even more complexity to this problem. Regarding to identifying the “parent” set, common words can be good cues. For example, “boat” and “ship” have the common word “vessel” whereas “carriage”, “chariot”, and “truck” share the words “vehicle” and “wheel”. This suggests that “boat” and “ship” may have parent “vessel” while “carriage”, “chariot”, and “truck” may have parent “vehicle” or “wheel”. This also implies that in the time of Gutenberg corpus, about eighty years ago, “vehicle” cannot be used to describe “boat” or “ship”. We confirm this hypothesis by checking our gold standard, Putnam’s Word Book, in which “vehicle” is shown as a synonym for “car”, “cart”, “wagon” and other conveyances on land but not for “boat” and “ship”, for which “vessel” is used instead. As another example, the word “horse” is only associated with “carriage” and “chariot”, which implies that they are historical vehicles powered by a “horse”.

[10] The target word is also included in its candidate list and the words in the intersection are ranked by their orders in the PPMIC list.

[11] The original meaning of car is similar to “chariot”, however this sense is missing in WordNet.

boat: ship, canoe, vessel, craft, launch, barge, passenger

ship: boat, vessel, passenger

carriage: train, vehicle, coach, cab, wheel, cart, horse, driver, passenger, station, street, window, door

chariot: car, vehicle, coach, wheel, horse, driver, team

truck: car, train, automobile, wagon, engine, vehicle, motor, wheel, cart, locomotive, auto, driver, ambulance, freight, trolley

FIG. 7.5. Examples for classifying distributional similar words for “car”

7.6 Simplification and Extension

As we discussed in Section 7.3.4, the parameter k in Table 7.5 imposes a lower bound on what PMI_{max} can compute. In other words, PMI_{max} cannot be used to measure association that is below normal association. The mathematical reason is that the term $f_d(w_1, w_2) - \frac{e^k}{N} \cdot (f_{w_1} \cdot f_{w_2} - \frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}})$ in Equation 7.4 must be a positive number since logarithm of a negative number is undefined. This causes a problem when we try to apply PMI_{max} to measure association between schema terms on the schema network.

To deal with this problem, we drop the term $\frac{e^k}{N} \cdot (f_{w_1} \cdot f_{w_2} - \frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}})$ from Equation 7.4, resulting in a simplified version of PMI_{max} as shown in Equation 7.6.

$$\text{(7.6)} \qquad \begin{aligned} \mathrm{PMI}^*(w_1, w_2) &= \log\Big(\frac{f_d(w_1, w_2) \cdot N}{\frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}}}\Big) \\ &= \log\Big(\frac{f_d(w_1, w_2) \cdot N}{f_{w_1} \cdot f_{w_2}}\Big) + \log(y_{w_1}) + \log(y_{w_2}) \\ &= \mathrm{PMI}(w_1, w_2) + \log(y_{w_1}) + \log(y_{w_2}) \end{aligned}$$

By applying Equation 7.5, it follows that

$$\text{PMI}^*(w_1, w_2) = \text{PMI}(w_1, w_2) + p\log\left(\log(f_{w_1}) + q\right) + p\log\left(\log(f_{w_2}) + q\right) - 2p\log\left(\log(700) + q\right) \tag{7.7}$$

We compare PMI^* with PMI_{max} on the Miller-Charles dataset. The parameters p and q in PMI^* are set with the same values as those used in PMI_{max}. The comparison result is given in Table 7.12. Interestingly, PMI^* achieves a very high correlation coefficient, 0.865, which even exceeds what PMI_{max} obtains, 0.856. By examining Table 7.12, we can find that the values produced by PMI_{max} and PMI^* are quite close, especially for those word pairs that are semantically similar. Their difference gradually increases as the word pair becomes less similar.

We also evaluated PMI^* on the TOEFL synonym test dataset. PMI^* achieves an accuracy of 76.25%, the same performance as PMI_{max} has. Again, PMI^* uses the same p and q parameters as those used by PMI_{max}.

The experiments shows that the drop of the term $\frac{e^k}{N} \cdot (f_{w_1} \cdot f_{w_2} - \frac{f_{w_1}}{y_{w_1}} \cdot \frac{f_{w_2}}{y_{w_2}})$ has little impact to the performance. More importantly, Equation 7.7 has a very simple interpretation, which allows it to move to other domains. The terms $p\log\left(\log(f_{w_1}) + q\right)$ and $p\log\left(\log(f_{w_2}) + q\right)$ simply compensate the bias that is caused by the term $\text{PMI}(w_1, w_2)$ since the larger f_{w_1} or f_{w_2} is, the bigger $p\log\left(\log(f_{w_1}) + q\right)$ or $p\log\left(\log(f_{w_2}) + q\right)$ will become.

Although it might work better if we relearn p and q when applying Equation 7.7 to the new domain, we chose to reuse p and q that had been learned from the automatic thesaurus generation. However, we need adjust the term occurrence and co-occurrence frequency counts in order to reuse p and q in a new domain that has a different universe size. Let the universe size of the Gutenberg Corpus be N and the universe size of the

Word Pair	human rate	PMI_{max}	PMI^*
car-automobile	3.92	10.498	10.513
gem-jewel	3.84	10.778	10.787
journey-voyage	3.84	8.567	8.717
boy-lad	3.76	9.168	9.285
coast-shore	3.7	10.039	10.079
asylum-madhouse	3.61	9.614	9.621
magician-wizard	3.5	10.242	10.251
midday-noon	3.42	9.041	9.087
furnace-stove	3.11	9.585	9.613
food-fruit	3.08	9.388	9.465
bird-cock	3.05	9.607	9.648
bird-crane	2.97	9.853	9.876
implement-tool	2.95	10.128	10.131
brother-monk	2.82	7.98	8.245
brother-lad	1.66	7.613	8.046
car-journey	1.16	8.201	8.426
monk-oracle	1.1	7.038	7.354
food-rooster	0.89	7.019	7.303
coast-hill	0.87	8.545	8.72
forest-graveyard	0.84	7.337	7.583
monk-slave	0.55	6.026	7.067
coast-forest	0.42	8.352	8.547
lad-wizard	0.42	6.521	7.015
chord-smile	0.13	7.012	7.423
glass-magician	0.11	7.632	7.872
noon-string	0.08	5.742	6.846
rooster-voyage	0.08	4.919	6.087
Correlation Coef.	**1**	**0.856**	**0.865**

Table 7.12. Comparisons of PMI^* with PMI_{max} on the MC dataset

schema network be N'. The term frequencies f_{t_1}, f_{t_2} and term co-occurrence frequency $f(t_1, t_2)$ in the schema network are corresponding to the word frequencies $\frac{N}{N'}f_{t_1}$, $\frac{N}{N'}f_{t_2}$ and word co-occurrence frequency $\frac{N}{N'}f(t_1, t_2)$ in the Gutenberg Corpus. After converting term occurrence and co-occurrence frequencies to word occurrence and co-occurrence frequencies, we can use Equation 7.7, which is learned on the Gutenberg Corpus, to approximately compute $\mathrm{PMI}^*(t_1, t_2)$ in the schema network. More specifically,

(7.8)

$$\begin{aligned}
\mathrm{PMI}^*(t_1, t_2) &= \log(\frac{\frac{N}{N'}f(t_1, t_2) \cdot N}{\frac{N}{N'}f_{t_1} \cdot \frac{N}{N'}f_{t_2}}) + p\log\left(\log(\frac{N}{N'}f_{t_1}) + q\right) + p\log\left(\log(\frac{N}{N'}f_{t_2}) + q\right) \\
&\quad - 2p\log\left(\log(700) + q\right) \\
&= \log(\frac{f(t_1, t_2) \cdot N'}{f_{t_1} \cdot f_{t_2}}) + p\log\left(\log(\frac{N}{N'}f_{t_1}) + q\right) + p\log\left(\log(\frac{N}{N'}f_{t_2}) + q\right) \\
&\quad - 2p\log\left(\log(700) + q\right)
\end{aligned}$$

As shown in Table 7.5, we learned different values of p and q for nouns and verbs. By comparing their performance on a small sample from the CAK of the DBLP+, we observed that p and q learned for verbs work better than those for nouns. Thus, we use $p = 1.5$ and $q = -5$ for the experiments in all other chapters in this thesis.

7.7 Conclusion

In this chapter, we described the characteristics of PMI as a measure of semantic similarity for words, and directly compared it with distributional similarity. We developed a new metric, PMI_{max}, by augmenting traditional PMI to take into account the number senses that a word has. We experimentally showed that PMI_{max} outperforms PMI in automatic thesaurus generation and on benchmark datasets for human similarity ratings and TOEFL synonym questions. PMI_{max} also achieves a correlation coefficient of 0.856 for the Miller-Charles dataset, which outperforms all previous corpus-based approaches.

In order to extend PMI_{max} to other domains, we created a simplified version, called PMI^*. We experimentally showed that PMI^* performs well and even outruns PMI_{max} on the Miller-Charles dataset with a correlation coefficient of 0.865.

Our experiments have demonstrated that PMI need not rely on Web search engine data or an information retrieval index to be effective in a range of semantic tasks. Compared with distributional similarity, PMI is a lightweight measure, though it requires a larger corpus to be effective. With the vast amount of data available today, data sparseness, becomes a much less severe issue than twenty years ago when Church and Hanks popularized the use of PMI in computational linguistics. We anticipate that PMI, PMI_{max} and PMI^* will play an important role in computing statistical association or semantic similarity in the future.

Chapter 8

SEMANTIC SIMILARITY MEASURES

We need to compute semantic similarity between concepts in the form of noun phrases (e.g., *City* and *Soccer Club*) and between relations in the form of short phrases (e.g., *crosses* and *birth date*). One way is distributional similarity [42], a statistical approach using a term's collective context information drawn from a large text corpus to represent the meaning of the term. Distributional similarity is usually applied to words but it can be generalized to phrases [65]. However, the large number of potential input phrases precludes precomputing and storing distributional similarity data and computing it dynamically as needed would take too long. Thus, we assume the semantics of a phrase is compositional on its component words and apply an algorithm to compute similarity between two short phrases using word similarity.

In this chapter, we first describe three word similarity measures that are used in this thesis. They include two semantic measures, a LSA model and a hybrid model combining LSA and WordNet, and a string similarity metric. Next, we describe the algorithm that makes use of the word similarity measures to compute short phrase similarity. Finally, as an evaluation of our text similarity measure, we describe our participation in *SEM 2013 STS (Semantic Textual Similarity) task [2]. Our system run, *ParingWords*, which was produced by an algorithm that extends the short phrase similarity metric, achieved the top place among 89 submitted runs from 35 teams.

8.1 Measuring Word Similarity

We expect two properties of any word similarity measure that can be used by our query system. First, it should produce similarity scores that fall in the range of $[0, 1]$. Second, it should have a good capability to differentiate semantically similar words from non-similar words. Ideally, it should give zero or near-zero scores to non-similar words and one or substantial portion of one scores (e.g. 0.5) to similar words. Accordingly, we define two types of errors that a word similarity measure can make. If the word similarity measure produces a low similarity score between two semantically similar words, we call it makes a *false negative* error. On the other hand, if the word similarity measure produces a high similarity score between two non-similar words, we call it makes a *false positive* error.

In this section, we first describe the LSA semantic similarity model, which is purely statistical. Next, we discuss the hybrid model that enhances the LSA similarity model with knowledge in WordNet. Finally, we present a string metric that measures how two words “look” like. This string similarity measure will be compared with the other two semantic measures in Chapter 9.

8.1.1 LSA Word Similarity

LSA Word Similarity relies on the distributional hypothesis that words occurring in the same contexts tend to have similar meanings [42]. LSA uses SVD transformation to reduce the dimension of the word co-occurrence matrix, which has been shown to be very effective to overcome the data sparseness problem of the word co-occurrence matrix [55].

Corpus Selection and Processing In order to produce a reliable word co-occurrence statistics, a very large and balanced text corpus is required. After experimenting with several corpus choices including Wikipedia, Project Gutenberg e-Books [43], ukWaC [7], Reuters News stories [87] and LDC gigawords, we selected the Web corpus from the Stanford WebBase project [94]. We used the February 2007 crawl, which is one of the largest collections and contains 100 million web pages from more than 50,000 websites. The WebBase project did an excellent job in extracting textual content from HTML tags but still has abundant text duplications, truncated text, non-English text and strange characters. We processed the collection to remove undesired sections and produce high quality English paragraphs. We detected paragraphs using heuristic rules and only retrained those whose length was at least two hundred characters. We eliminated non-English text by checking the first twenty words of a paragraph to see if they were valid English words. We used the percentage of punctuation characters in a paragraph as a simple check for typical text. We removed duplicated paragraphs using a hash table. Finally, we obtained a three billion words corpus of good quality English, which is available at [38].

Word Co-Occurrence Generation We performed POS tagging and lemmatization on the WebBase corpus using the Stanford POS tagger [98]. Word/term co-occurrences are counted in a moving window of a fixed size that scans the entire corpus[1]. We generated two co-occurrence models using window sizes ± 1 and ± 4 because we observed different natures of the models. ± 1 window produces a context similar to the dependency context used in [62]. It provides a more precise context but only works for comparing words within the same POS. In contrast, a context window of ± 4 words allows us to compute semantic similarity between words with different POS.

Our word co-occurrence models were based on a predefined vocabulary of more than 22,000 common English words and noun phrases. After words are POS tagged, the dimensions of our word co-occurrence matrices become 26,230 $\times$ 26,230. Our vocabulary

[1] We used a stop-word list consisting of only the three articles “a”, “an” and “the”.

Word Pair	±4 model	±1 model
1. doctor_NN, physician_NN	0.775	0.726
2. car_NN, vehicle_NN	0.748	0.802
3. person_NN, car_NN	0.038	0.024
4. car_NN, country_NN	0.000	0.016
5. person_NN, country_NN	0.031	0.069
6. child_NN, marry_VB	0.098	0.000
7. wife_NN, marry_VB	0.548	0.274
8. author_NN, write_VB	0.364	0.128
9. doctor_NN, hospital_NN	0.473	0.347
10. car_NN, driver_NN	0.497	0.281

Table 8.1. Ten examples from the LSA similarity model

includes only open-class words (i.e. nouns, verbs, adjectives and adverbs). There are no proper nouns in the vocabulary with the only exception of country names.

SVD Transformation Singular Value Decomposition (SVD) has been found to be effective in improving word similarity measures [55]. SVD is typically applied to a *word by document* matrix, yielding the familiar LSA technique. In our case we apply it to our *word by word* matrix. In literature, this variation of LSA is sometimes called HAL (Hyperspace Analog to Language) [14].

Before performing SVD, we transform the raw word co-occurrence count f_{ij} to its log frequency $log(f_{ij} + 1)$. We select the 300 largest singular values and reduce the $26,230$ word vectors to 300 dimensions. The LSA similarity between two words is defined as the cosine similarity of their corresponding word vectors after the SVD transformation. When the cosine of the angle between two word vectors is negative, we change it to zero so that all the similarity scores can fall in the range of $[0, 1]$.

LSA Word Similarity Examples Ten examples obtained using LSA similarity are given in Table 8.1. Examples 1 to 6 illustrate that the metric has a good property of differentiating similar words from non-similar words. Non-similar words can hardly obtain a score larger than 0.1 while highly similar words can often have a score above 0.5. Examples 7 and 8 show that the ± 4 model can detect semantically similar words even with different POS while the ± 1 model yields much worse performance. Example 9 and 10 show that words that are semantically related but not substitutable as concepts can have a high score by the ± 4 model but a significantly lowered score by the ± 1 model. We refer to the ± 1 model and the ± 4 model as our implementation of *concept similarity* and *relation similarity* respectively since the ± 1 model has a good performance on nouns or concepts and the ± 4 model is good at computing similarity between relations, such as “marry” and “wife”.

TOEFL Synonym Evaluation We evaluated the ± 1 and ± 4 models on the 80 TOEFL synonym test questions. The ± 1 and ± 4 models correctly answered 73 and 76 questions, respectively. One question is not answerable because the question word “half-heartedly” is not in our vocabulary. If we exclude this question, ± 1 achieves an accuracy of 92.4% and ± 4 achieves 96.2%, which outperforms all previous corpus-based approaches. It is worth mentioning that this outstanding performance is obtained even without tuning the models on the TOEFL synonym questions.

It is interesting that our models can perform so well since we do not introduce brand-new techniques. However, there are several places that can distinguish our approach from all others. First, we used a very large corpus with high quality. Second, we performed POS tagging and lemmatization on the corpus. Third, our vocabulary includes only open-class or content words. Which aspect contributes most to the boost of performance has not yet been investigated and we plan to do it in our future work.

8.1.2 Hybrid Word Similarity

Although our LSA word similarity model performs very well, it still has limitations. Its capability to differentiate similar words from related words is good but not perfect. Some related words can have similarity scores almost as high as what similar words get. Word similarity is typically low for synonyms having many word senses since information about different senses are mashed together [40]. The similarity between words with different POS can sometimes be lower than what is expected.

To lessen the above issues, we develop hybrid word similarity measures that use a three-step procedure to combine the LSA word similarity models and human-crafted knowledge in WordNet. Step 1 improves the capability of the models to differentiate similar words from non-similar words. Step 2 reduces false negative errors brought by words that have many senses. Step 3 reduces false negative errors occurring between words with different POS. We apply Step 1 and 2 to both concept and relation similarity models and Step 3 only to the relation similarity model.

Step 1. Boosting LSA similarity using WordNet We increase the similarity between two words if any relation in the following list holds.

- They are in the same WordNet synset.
- One word is the direct hypernym of the other.
- One word is the two-link indirect hypernym of the other.
- One adjective has a direct *similar to* relation with the other.
- One adjective has a two-link indirect *similar to* relation with the other.
- One word is a derivationally related form of the other.
- One word is the head of the gloss of the other or its direct hypernym or one of its direct hyponyms.

- One word appears frequently in the glosses of the other and its direct hypernym and its direct hyponyms.

We use the algorithm described in [21] to find a word gloss header. We require a minimum LSA similarity of 0.1 between the two words to filter out noisy data when extracting WordNet relations.

We define a word's "significant senses" to deal with the problem of WordNet trivial senses. The word "year", for example, has a sense "a body of students who graduate together" which makes it a synonym of the word "class". This causes problems because "year" and "class" are not similar, in general. A sense is significant, if any of the following conditions is met: (i) it is the first sense; (ii) its WordNet frequency count is not less than five; or (iii) its word form appears first in its synset's word form list and it has a WordNet sense number less than eight.

We assign path distance of zero to the category 1, path distance of one to the category 2, 4 and 6, and path distance of two to the other categories. The new similarity between word x and y by combining LSA similarity and WordNet relations is shown in the following equation

$$sim_{\oplus}(x,y) = sim_{LSA}(x,y) + 0.5e^{-\alpha D(x,y)} \tag{8.1}$$

where $D(x,y)$ is the minimal path distance between x and y. Using the $e^{-\alpha D(x,y)}$ to transform simple shortest path length has been demonstrated to be very effective according to [59]. The parameter α is set to be 0.25, following their experimental results. The new similarity $sim_{\oplus}(x,y)$ can be a value exceeding 1.0. For *concept similarity*, we divide $sim_{\oplus}(x,y)$ by 1.5 to normalize it. For *relation similarity*, we simply cut the excess over 1.0.

Step 2. Dealing with words of many senses For a word w with many WordNet senses (currently ten or more), we use its synonyms with fewer senses (at most one third of that of w) as its substitutions in computing similarity with another word. Let S_x and S_y be the sets of all such substitutions of the words x and y respectively. The new similarity is obtained using Equation 8.2.

$$\begin{aligned} sim(x,y) = max(& \max_{s_x \in S_x \cup \{x\}} sim_{\oplus}(s_x, y), \\ & \max_{s_y \in S_y \cup \{y\}} sim_{\oplus}(x, s_y)) \end{aligned} \tag{8.2}$$

Step 3. Dealing with words of different POS $sim(x, y)$ obtained in Step 2 can sometimes make false negative errors for words with different POS, especially when one of the words has many senses. For example, $sim(x, y)$ produces a similarity score 0.555 to the word pair "manager_NN" and "leader_NN" but only 0.003 to the word pair "manager_NN" and "lead_VB". The verb "lead" has many senses, which probably leads to the very low similarity with "manager_NN".

By using "leader", a derivative of "lead", as the bridge, we can actually find similarity between "manager_NN" and "lead_VB". A high similarity 0.608, produced by $sim(x, y)$, exists between "leader_NN" and "lead_VB". Because "manager_NN" is similar to "leader_NN" and "leader_NN" is similar to "lead_VB", we can deduce that "manager_NN" is similar to "lead_VB".

Let D_x and D_y be the sets of the derivatives of the words x and y respectively. Equation 8.3 shows how we find the indirect similarity between the words x and y.

$$sim'(x,y) = max(\max_{d_x \in D_x} (sim(x,d_x) \cdot sim(d_x,y) \cdot r(d_x))\,, \max_{d_y \in D_y} (sim(x,d_y) \cdot sim(d_y,y) \cdot r(d_y))) \tag{8.3}$$

where $r(z)$ is a ratio used to diminish false positive errors, whose value depends on the POS of the derivative z. if z is a verb, $r(z) = 1.0$; if z is a noun, $r(z) = 0.5$. The noun derivative has a lower ratio because similarity between nouns, in general, is significantly higher than similarity between verbs. If the word x (y) has more than ten senses, we allow its derivative d_x (d_y) can have a different stem from x (y). If $sim'(x,y)$ is larger than $sim(x,y)$, we replace $sim(x,y)$ with the value of $sim'(x,y)$.

8.1.3 String Word Similarity

String word similarity measures how two words "look" like, i.e. their surface likeness. There are many existing string similarity measures. We select a simple and common measure that is based on character bigrams and Dice coefficient. Its implementation is as follows. First, we computes character bigram sets for each of the two words without using padding characters. Next, Dice coefficient is applied to get the degree of overlap between the two sets. Let X and Y be the character bigram sets for the word x and y, respectively. Equation 8.4 gives our string similarity metric.

$$sim(x,y) = \frac{2\,|X \cap Y|}{|X| + |Y|} \tag{8.4}$$

where $|A|$ gives the cardinality of the set A. This string similarity metric has been used in the STS task to deal with words that are out of our vocabulary.

8.2 Measuring Lexical Similarity

We use a phrase similarity algorithm to compute similarity between concepts (classes) and relations (properties) in the form of noun phrases or verb phrases. For two given phrases P_1 and P_2, we pair the pos-tagged words in P_1 to the pos-tagged words in P_2 in a way that it maximizes the sum of word similarities of the resulting word-pairs. The maximized sum of word similarities is further normalized by the number of word-pairs. The same process is repeated for the other direction, i.e., from P_2 to P_1. The scores from both directions are then combined using average. The specific metric is shown in Equation 8.5.

$$\begin{aligned} sim(P_1, P_2) &= \frac{\sum_{w_1 \in \{P_1\}} \max\limits_{w_2 \in \{P_2\}} sim(w_1, w_2)}{2 \cdot |P_1|} \\ &+ \frac{\sum_{w_2 \in \{P_2\}} \max\limits_{w_1 \in \{P_1\}} sim(w_2, w_1)}{2 \cdot |P_2|} \end{aligned} \tag{8.5}$$

where $sim(x, y)$ is the word similarity measure of choice. Our metric follows the one proposed by Mihalcea et al. [71]. However, our metric differs from theirs in that (1) we rely only on word similarity and ignores other features like tf-idf and (2) we allow pairing words with different parts-of-speech.

Computing semantic similarity between concepts requires additional work. Before running algorithm on two noun phrases, we compute the semantic similarity of their head nouns. If it exceeds an experimentally determined threshold, 0.075, we apply the above metric but with their head nouns being prior-paired and if not, the phrases have similarity of zero. Thus we know that *dog house* is not similar to *house dog*.

Since concepts (classes) and relations (properties) are still lexical items, what Equation 8.5 computes is still lexical similarity. Therefore, the type of the word similarity model can also be used to refer to the resulting phrase similarity. For example, if LSA word similarity is used, the phrase similarity can be referred to as LSA lexical similarity. For another instance, if the concept similarity model is used, the phrase similarity can also be called concept similarity.

8.3 STS Evaluation

In this section, we describe a semantic text similarity system developed for the *SEM 2013 STS shared task [2]. The system is an extension of the phrase similarity in Section 8.2 and it achieved the top place among 89 submitted runs from 35 teams. The success of this system further verifies the excellent performance of our word similarity and phrase similarity measures.

8.3.1 Introduction

Measuring semantic text similarity has been a research subject in natural language processing, information retrieval and artificial intelligence for many years. Previous efforts have focused on comparing two long texts (e.g., for document classification) or a short text with a long text (e.g., Web search), but there are a growing number of tasks requiring computing the semantic similarity between two sentences or other short text sequences. They include paraphrase recognition [27], Twitter tweets search [93], image retrieval by captions [19], query reformulation [70], automatic machine translation evaluation [53] and schema matching [39].

There are three predominant approaches to computing short text similarity. The first uses information retrieval's vector space model [69] in which each text is modeled as a "bag of words" and represented using a vector. The similarity between two texts is then computed as the cosine similarity of the vectors. A variation on this approach leverages web search results (e.g., snippets) to provide context for the short texts and enrich their vectors using the words in the snippets [89]. The second approach is based on the assumption that if two sentences or other short text sequences are semantically equivalent, we should be able to align their words or expressions. The alignment quality can serve as a similarity measure. This technique typically pairs words from the two texts by maximizing the summation of the word similarity of the resulting pairs [71]. The third approach combines different measures and features using machine learning models. Lexical, semantic and syntactic features are computed for the texts using a variety of resources and supplied to a classifier, which then assigns weights to the features by fitting the model to training data [90].

For evaluating different approaches, the 2013 Semantic Textual Similarity (STS) task asked automatic systems to compute sentence similarity according to a scale definition ranging from 0 to 5, with 0 meaning unrelated and 5 semantically equivalent [3, 2]. The example sentence pair “The woman is playing the violin” and “The young lady enjoys listening to the guitar” is scored as only *1* and the pair “The bird is bathing in the sink” and “Birdie is washing itself in the water basin” is given a score of *5*.

The vector-space approach tends to be too shallow for the task, since solving it well requires discriminating word-level semantic differences and goes beyond simply comparing sentence topics or contexts. Our system uses an *align-and-penalize* algorithm, which extends the second approach by giving penalties to both the words that are poorly aligned and the alignments causing semantic or syntactic contradictions.

8.3.2 Align-and-Penalize Approach

First we hypothesize that STS similarity between two sentences can be computed using

$$STS = T - P' - P'' \tag{8.6}$$

where T is the term alignments score, P' is the penalty for bad term alignments and P'' is the penalty for syntactic contradictions led by the alignments. However P'' had not been fully implemented and was not used in our STS submissions. We show it here just for completeness.

Aligning terms in two sentences We start by applying the Stanford POS tagger to tag and lemmatize the input sentences. We use our predefined vocabulary, POS tagging data and simple regular expressions to recognize multi-word terms including noun and verb phrases, proper nouns, numbers and time. We ignore adverbs with frequency count larger than $500,000$ in our corpus and stop words with general meaning.

Equation 8.7 shows our aligning function g which finds the counterpart of term $t \in S$ in sentence S'.

$$g(t) = \underset{t' \in S'}{argmax}\; sim'(t, t') \tag{8.7}$$

$sim'(t, t')$ is a wrapper function over $sim(x, y)$ in Equation 8.2 that uses the *relation similarity* model[2]. It compares numerical and time terms by their values. If they are equal, 1 is returned; otherwise 0. $sim'(t, t')$ provides limited comparison over pronouns. It returns 1 between subject pronouns *I*, *we*, *they*, *he*, *she* and their corresponding object pronouns. $sim'(t, t')$ also outputs 1 if one term is the acronym of the other term, or if one term is the head of the other term, or if two consecutive terms in a sentence match a single term in the other sentence (e.g. “long term” and “long-term”). $sim'(t, t')$ further adds support for matching words[3] not presented in our vocabulary using the string similarity metric in Equation 8.4. If the string similarity is larger than two thirds, $sim'(t, t')$ returns a score of

[2] The model developed for the STS task is slightly different from the model described in Section 8.1.2. Please refer to [41] for details.

[3] We use the regular expression “[A-Za-z][A-Za-z]*” to identify them.

1; otherwise 0.

$g(t)$ is direction-dependent and does not achieve one-to-one mapping. This property is useful in measuring STS similarity because two sentences are often not exact paraphrase of one another. Moreover, it is often necessary to align multiple terms in one sentence to single term in the other sentence, such as when dealing with repetitions and anaphora or, e.g., mapping "people writing books" to "writers".

Let S_1 and S_2 be the sets of terms in two input sentences. We define term alignments score T as the following equation shows.

$$\frac{\sum_{t \in S_1} sim'(t, g(t))}{2 \cdot |S_1|} + \frac{\sum_{t \in S_2} sim'(t, g(t))}{2 \cdot |S_2|} \tag{8.8}$$

Penalizing bad term alignments We currently treat two kinds of alignments as "bad", as described in Equation 8.9. For the set B_i, we have an additional restriction that neither of the sentences has the form of a negation. In defining B_i, we used a collection of antonyms extracted from WordNet [74]. Antonym pairs are a special case of disjoint sets. The terms "piano" and "violin" are also disjoint but they are not antonyms. In order to broaden the set B_i we will need to develop a model that can determine when two terms belong to disjoint sets.

$$\begin{gathered} A_i = \{\langle t, g(t) \rangle \,|t \in S_i \wedge sim'(t, g(t)) < 0.05\} \\ B_i = \{\langle t, g(t) \rangle \,|t \in S_i \wedge t \text{ is an antonym of } g(t)\} \\ i \in \{1, 2\} \end{gathered} \tag{8.9}$$

We show how we compute P' in Equation 8.10.

$$
\begin{aligned}
P_i^A &= \frac{\sum_{\langle t,g(t)\rangle \in A_i} (sim'(t,g(t)) + w_f(t) \cdot w_p(t))}{2 \cdot |S_i|} \\
P_i^B &= \frac{\sum_{\langle t,g(t)\rangle \in B_i} (sim'(t,g(t)) + 0.5)}{2 \cdot |S_i|} \\
P' &= P_1^A + P_1^B + P_2^A + P_2^B \qquad (8.10)
\end{aligned}
$$

The $w_f(t)$ and $w_p(t)$ terms are two weighting functions on the term t. $w_f(t)$ inversely weights the log frequency of term t and $w_p(t)$ weights t by its part of speech tag, assigning 1.0 to verbs, nouns, pronouns and numbers, and 0.5 to terms with other POS tags.

8.3.3 Results and discussion

Table 8.2 presents the official results of the *ParingWords* run and the other two runs submitted by our team in the 2013 STS task. Each entry gives a run's Pearson correlation on a dataset as well as the rank of the run among all 86 runs submitted by the 35 teams. The last row shows the mean of the correlations and the overall ranks of our three runs.

Our other two runs use a support vector regression model to combine a large number of general and domain specific features, including the output of the *ParingWords* run. However, the machine learning models suffered the overfitting problem and did not perform as well as the simple approach used by the *ParingWords* run.

Dataset	PairingWords	Galactus	Saiyan
Headlines (750 pairs)	0.7642 (3)	0.7428 (7)	**0.7838 (1)**
OnWN (561 pairs)	0.7529 (5)	0.7053 (12)	0.5593 (36)
FNWN (189 pairs)	**0.5818 (1)**	0.5444 (3)	0.5815 (2)
SMT (750 pairs)	0.3804 (8)	0.3705 (11)	0.3563 (16)
weighted mean	**0.6181 (1)**	0.5927 (2)	0.5683 (4)

Table 8.2. Performance of our three runs on test sets.

Chapter 9

EVALUATION

In this chapter, we evaluate our approach on two datasets, DBLP+ and DBpedia. Each of them contains tens of millions of facts or triples. DBpedia has a broad and relatively shallow domain whereas DBLP+ has a narrow but deeper domain. Besides showing how well our schema-free querying approach run on the two datasets, we also present a variety of experiments that analyze performance and examine the hypotheses and assertions we made in the earlier chapters.

We evaluate and compare three different similarity metrics across most of the experiments. The first is the hybrid semantic similarity model combining LSA and WordNet. The second is the LSA semantic similarity, a pure statistical measure. The third one is string similarity. The comparison between them tells us how much better the system can be by using semantic measures. If the LSA semantic similarity is used, our system is then purely built on statistics learned from data in the knowledge base and a large text corpus. Such a system requires no human-crafted knowledge including data schema, pre-collected mappings and a thesaurus for synonym extension.

Our approach assumes that there is a big index mapping from all the names of an entity to the id of the entity. However, in reality this index is often incomplete. On the DBpedia dataset, we show a pragmatic approach to match entities when formulating formal queries.

In the followings, we will first describe our evaluation methodology and then present evaluation results on DBLP+ and DBpedia.

9.1 Evaluation Methodology

In this section, we will describe our performance metrics, how to tune the parameters and the methods we use to validate the evaluation.

9.1.1 Performance Metrics

Mean Reciprocal Rank (MRR) is the principal metric we use to evaluate our system. MRR is a popular measure for evaluating information retrieval systems that produce a list of possible responses to each query q_i in a given sample Q. Equation 9.1 shows the formula of MRR.

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{r_i} \tag{9.1}$$

where $|Q|$ denotes the cardinality of the set Q and r_i is the rank of the first correct response in the list for the query q_i.

In our context, a response means an interpretation or mapping of the query q_i. Our system only produce a list of top-10 interpretations. Hence, if the correct interpretation are not in the top 10, its reciprocal rank $\frac{1}{r_i}$ will simply have a value of zero. Essentially, MRR measures how high the first correct interpretation are in the output list. That is, how fast the user can reach a correct interpretation by looking through the list.

Alternatively, we can use precision and recall metrics to evaluate our system. In this case, we take further steps by generating formal queries (e.g. SPARQL) for the best interpretation of the query q_i and actually execute the formal queries to get answers. If the formal queries return no result we accept for the next best interpretation. This continues until we get a non-empty result. Then the answer set is compared with gold standard answers for computing precision and recall. In contrast to typical IR tasks that produce different precisions at different recall levels for a given query, our precision and recall are two single values. Equation 9.2 gives their definition for every query q_i.

$$\begin{aligned} Precision_i &= \frac{number\ of\ correct\ system\ answers}{number\ of\ system\ answers} \\ Recall_i &= \frac{number\ of\ correct\ system\ answers}{number\ of\ gold\ standard\ answers} \end{aligned} \tag{9.2}$$

The overall performance is measured by precision and recall averaged over the query sample Q, which is shown in Equation 9.3. Since the term "average precision" has a special meaning in IR research community, in order to avoid terminology confusion we use the terms *mean precision* and *mean recall*.

$$\begin{aligned} Mean\ Precision &= \frac{1}{|Q|}\sum_{i=1}^{|Q|} Precision_i \\ Mean\ Recall &= \frac{1}{|Q|}\sum_{i=1}^{|Q|} Recall_i \end{aligned} \tag{9.3}$$

However, precision and recall metrics are not our preference because computing them requires an additional step to run the query on a triplestore or RDBMS. As shown in next section, we use grid searching to find optimal parameters of our approach, which is very computationally expensive. Therefore, MRR has clear advantage over precision and recall metrics due to fast computation. In this chapter, precision and recall metrics are only employed when we compare the performance of our system with other systems on the DBpedia dataset.

9.1.2 Parameter Optimization

We need experimentally determine seven parameters for the approach. Three of them come from the first phase of the approach, including the size of the class candidate list k_1, the size of the property candidate list k_2 and the size of the top concept mapping hypothesis list k_3. Three are from the second phase, including α that counteracts the bias of the joint lexical semantic similarity in Equation 5.14, β that balances similarity and popularity in Equation 5.8, and θ that weights subject and object similarity for default relations in Equation 5.17. The last parameter is γ, which controls the transfer rate in Equation 4.4 for building the schema path model.

Since exhaustively searching into seven-dimensional space is very computationally expensive, we use a "step-by-step" method that divide the parameters and sets their values in order by exploiting their different characters. A simple way to select the values of k_1, k_2 and k_3 is to use sufficiently large numbers. Although using large k numbers demands time to carry out the experiments, it helps to find the optimal values for the other four parameters. Once α, β, θ and γ are resolved, we can in turn vary the values of k numbers to see their impact on the performance.

Grid searching in the parameter space composed by α, β, γ and θ is still too expensive in our scenario so we further divide them. Among the four parameters, α and γ are probably the most correlated because both of them adjust the relative weight of long schema paths against short schema paths. Moreover, both α and γ have finite ranges while β and θ have not. Thus, we first apply grid searching to determine the optimal values of α and γ while fixing β and θ with the prior values, 0 and 1, respectively. After the values of α and γ are resolved, we experiment with different values of β to find its optimal whereas θ remains as 1. When α, γ and β are all known we finally vary the value of θ to locate the one that yields the best performance. By doing these steps, we actually imply assumptions that 1) selecting optimal values of α and γ is independent of the values of β and θ and 2) selecting optimal value of β is independent of the value of θ. Obviously, the assumptions are not strictly true and we make them simply for making the problem tractable. Although the parameters solved in this way may not be the true global optimal, they often work as good approximation.

9.1.3 Validation

To demonstrate the generalizability of our approach, we carry out two-fold cross-validation and cross-domain validation. In two-fold cross-validation, we randomly partition the query sample Q into two equal half Q_1 and Q_2. Then, we train on Q_1 and test on Q_2 and vice versa. The validation result is the average performance of the two rounds. In cross-domain validation, we train the system using the query sample Q on the DBLP+ dataset and then test the system using the other query sample Q' on the DBpedia dataset. We further compare the system's performance with other existing systems which have already reported results on Q'. The cross-domain validation verifies if the methods and parameters in our approach are of general nature and can be applied to different domains.

9.2 Evaluation Environment

Unless specified otherwise, all the evaluations were done on a computer with 2.33GHz Intel Core2 CPU and 8GB memory. We call it the testing machine.

9.3 Evaluation on DBLP+

In this section, we discuss our evaluation results on the DBLP+ dataset, which we created by augmenting the DBLP dataset with data from CiteSeerX and ArnetMiner.

9.3.1 Data

DBLP is a dataset about computer science publications. It records more two million publications published in more than 1,000 journals and 5,000 conferences or workshops. The DBLP dataset has high data quality and it even does a good job in disambiguating author names and recognizing alias. However, it misses some important relations that computer science researchers will be interested in querying against. For example, DBLP has only a collection of 110,000 citations, almost all of which occur before early 90s. For another instance, DBLP provides no information about either the subjects of publications or author affiliations.

CiteSeerX is another well-known source in the computer science publication domain. It automatically crawled and indexed a large collection of computer science publications with a richer set of relations than what DBLP has. Its citation data is several orders of magnitude larger than DBLP's. It has much more information about authors such as the emails, homepages and affiliations. It also includes abstract as one of the metadata of papers. The problem with CiteSeerX is its data quality. For example, many paper titles in CiteSeerX contains some extra prefix or suffix characters. Unlike DBLP whose data is processed and compiled by humans, CiteSeerX is an automated system that depends on programs.

ArnetMiner is also a good work in developing academic search tools. One remarkable feature of ArnetMiner is its subject categories that are automatically mined from its collection of publications using topic models. Moreover, it provides a high quality dataset of about two million ACM citations.

The more relations are included in our dataset, the better our approach can be evaluated. Besides, we also intend to develop an online demo of our system as a way to advance research in this direction. It would be much more appealing to the community if the online

Statistical property	**Value**
Number of types	31
· classes	19
· attribute types	12
Number of properties	30
· object properties	17
· datatype properties	13
Number of relations (triples)	42,243,494
Number of type definitions	12,169,009
Number of instances	3,363,179

Table 9.1. DBLP+ dataset statistics

system can answer questions like “*Give me papers most cited by the papers published in VLDB conference 2012*” or “*Show me how the subjects of papers published by Microsoft changed over the years between 2000 to 2012*”. This kind of queries would be useful to discover hidden knowledge or new trends and no other systems have ever supported answering them. These are motivations by which we think it necessary to create a new dataset that combines the strength of DBLP, CiteSeerX and ArnetMiner.

Our DBLP+ dataset is built on top of DBLP and consists of all publications in DBLP. It does not include any other publications from CiteSeerX or ArnetMiner. What we add to DBLP is some new relations about the DBLP publications that are retrieved from other sources. To enable this, we need integrate CiteSeerX and ArnetMiner data with DBLP data. The key for the integration is linking identical publications. Most of publications have unique title, making the linking task relatively easier. However, we pre-process the titles of CiteSeerX publications to remove the extra prefix or suffix characters based on simple patterns. Besides title, we also use other metadata to facilitate the mapping, which includes authors, publication year and publication type (article, inProceedings, book and etc.). If two publications with the same title are found, we require their aforementioned metadata, if available, to be the same before we link them. If not all the metadata is available for the

class	number of instances
Article	888,583
Author	1,192,562
Book	9,597
Conference	5,523
Proceedings	19,302
Country	182
Editor	20,269
InBook	22,717
InProceedings	1,193,702
Institution	20,441
Journal	1,283
Paper	2,082,285
Person	1,192,562
Publication	2,141,550
Publisher	1,003
Series	682
Thesis	6,922
Thing	3,363,179
Venue	6,806

Table 9.2. DBLP+ classes with their number of instances

two publications, we employ a set of rules to decide whether we link them based on their existing metadata. Once the two publications are linked as identical, we can further link the publication authors in different sources and so on. The whole process is tedious and involves many details. It took us three months to complete the integration.

The statistical properties of the DBLP+ dataset are shown in Table 9.1[1]. There are 31 types in the DBLP+ ontology, including 19 classes and 12 attribute types. There are much more type definitions than instances because one instance can have multiple types. Table 9.2 shows how many instances the 19 classes are populated by[2]. Five of 12 attribute types are predefined data types, including *N̂umber*, *D̂ate*, *Ŷear*, *T̂ext* and *L̂iteral*, of which

[1] The count of type definitions does not include attribute types.

[2] The count of *Thing* does not include the instances of attribute types.

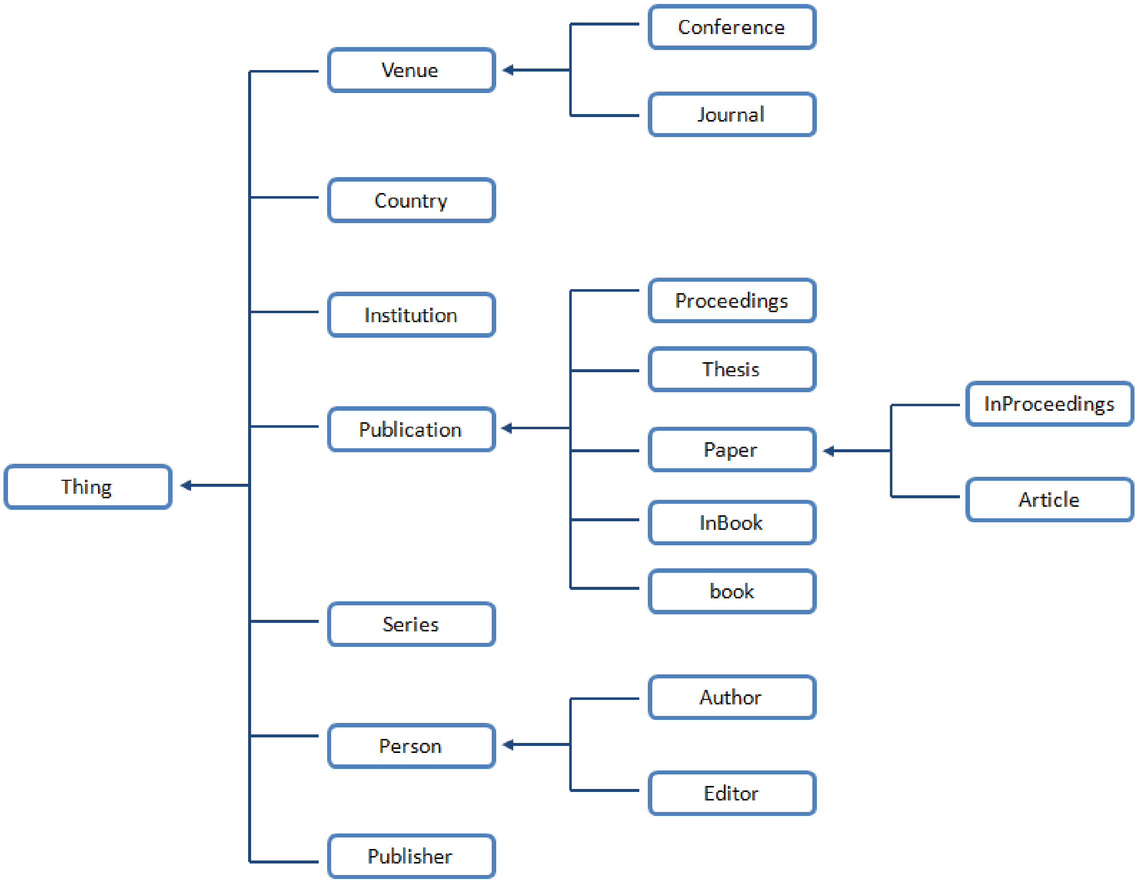


FIG. 9.1. DBLP+ class hierarchy

$\hat{Literal}$ is the super type of the other four. The remaining seven attribute types are automatically inferred from the labels of data type properties, including $\hat{Abstract}$, $\hat{DOI}$, $\hat{E}$-*mail*, $\hat{Homepage}$, $\hat{ISBN}$, $\hat{Subject}$ and $\hat{Name}$.

The 19 classes of DBLP+ are organized into a hierarchy tree as in Figure 9.1. A well-defined ontology helps improving performance of our system. For example, we introduce the *Paper* class, which is not in DBLP , and make it the super class of *InProceedings* and *Article*. If the *Paper* class did not exist in our ontology, our system would have difficulty in answering queries involving “paper” because mapping “paper” to either *InProceedings* or *Article* is not fully correct. The taxonomy structure can also be beneficial for a better recall.

Our semantic similarity measure is not perfect and sometimes our model may mistakenly produce a low similarity score between an input concept and its target class. However, it is likely that our model can find similarity between the input concept and the superclass of its target class. Although the resulting mapping is a “into” not “onto”, it still helps improving recall.

Table 9.3 shows the distribution of the relations over the 30 properties in DBLP+. Datatype properties are distinguished from object properties by an initial “@” character. Our DBLP+ dataset has about 4,400,000 citations, 40 times more than the citations in DBLP. It contains subjects for more than two million publications, authors and venues. It also offers relations about e-mail of author, institution of author, abstract of publication, institution of publication and country of institution, all of which are not in DBLP.

Unlike a typical ontology, there are no domain and range definitions for the DBLP+ properties. For natural language interface systems, the domain and range information has been mainly used for suggesting whether a class and a property is connectible. However, in many cases, domain and range definitions have difficulties in serving this purpose. For example, the property *institution* in DBLP+ is used to describe both *Publication* and *Author*. Therefore, there is no a single domain for *institution*. For another example, in DBLP+ only *Book* and *Proceedings* have the attribute *@ISBN*. If we define the domain of *@ISBN* as the union set of *Book* and *Proceedings*, a question arises that whether *@ISBN* is connectible to *Publication*. Only a small portion of the class *Publication* owns the property *@ISBN* but it still makes sense when people ask “give me ISBN of Lushan Han’s publications”.

Instead, we take a much more flexible and labor-saving approach – compute association degree between properties and classes directly from data as discussed in Section 4.4. Figure 9.2 gives three *directed* classes and their associated properties along with association

property	**number of relations (triples)**
@DOI	1,661,415
@ISBN	25,067
@abstract	739,209
@e-mail	130,283
@homepage	29,227
@issueNumber	794,558
@name	3,363,219
@numberOfCitations	3,360,814
@numberOfPublications	1,239,293
@pageNumbers	1,973,524
@publicationYear	2,140,823
@subject	2,104,388
@volumeNumber	899,169
author	5,772,593
book	21,556
cites	4,483,687
conference	1,212,985
country	14,410
editor	45,778
firstAuthor	2,109,302
firstEditor	16,219
institution	495,908
journal	888,583
primaryAuthor	3,796,811
proceedings	1,099,788
publisher	28,318
secondAuthor	1,687,509
secondEditor	13,567
series	13,206
venue	2,082,285
total	42,243,494

Table 9.3. DBLP+ properties with their number of relations

←**Book**: @ISBN 8.4, publisher 8.4, series 7.9, editor 6.0, firstEditor 5.9, secondEditor 5.8, @publicationYear 4.8, firstAuthor 4.7, @pageNumbers 4.6, primaryAuthor 4.5, @numberOfCitations 4.4, @name 4.4, author 4.3, secondAuthor 4.0, cites 3.9, @volumeNumber 3.7, institution 0.1

←**Proceedings**: editor 8.9, firstEditor 8.7, secondEditor 8.7, publisher 8.5, series 8.4, @ISBN 8.4, conference 5.4, @numberOfPublications 5.3, @volumeNumber 4.9, @publicationYear 4.9, @numberOfCitations 4.5, @name 4.5, cites 0.0

←**Publication**: author 5.2, cites 5.2, primaryAuthor 5.2, @publicationYear 5.1, firstAuthor 5.1, venue 5.1, @pageNumbers 5.1, secondAuthor 5.1, @DOI 5.1, conference 5.0, proceedings 5.0, @volumeNumber 5.0, journal 5.0, @issueNumber 5.0, @abstract 5.0, @subject 4.9, @numberOfCitations 4.7, @name 4.7, institution 4.6, editor 4.5, publisher 4.4, @ISBN 4.4, book 4.3, firstEditor 4.3, secondEditor 4.2, series 4.2, @numberOfPublications 0.9

FIG. 9.2. Three classes with their associated properties from the CAK of DBLP+

degree measured by PMI[3]. Since *Book*, *Proceedings* and *Publication* are similar concepts, their associated properties are largely overlapped. However, not all the properties of them are the same. For example, *Proceedings* but not *Book* has the *conference* property. Moreover, the same property can have different association degree with the three classes. For example, the property *@ISBN* has a high PMI 8.4 with both *Book* and *Proceedings* and a much lower one 4.4 with *Publication*. These information is used to disambiguate competitive concept candidates in our phase 1 algorithm.

If we only consider direct relations, the degree of connectivity between the 18 classes[4] is low. The histogram in Figure 9.3 shows the distribution of connectivity degree of 324 class pairs resulted from pairing every $\leftarrow C_i$ with every $\rightarrow C_j$ where i $\in$ 1..18 and j $\in$ 1..18. The degree of connectivity between two *directed classes* $\leftarrow C_1$ and $\rightarrow C_2$ is defined as the number of distinct schema paths that connect $\leftarrow C_1$ and $\rightarrow C_2$. When it comes to direct relations, it is equivalent to the number of distinct properties that can go from C_1 to C_2 but not including the other way around. Figure 9.3 indicates that the 324 *directed class* pairs are either not connected or very loosely connected.

However, the degree of connectivity increases drastically as we include indirect rela-

[3] The complete view of association knowledge between classes and properties is supplied in Appendix A.1

[4] These do not include the most general class *Thing*.

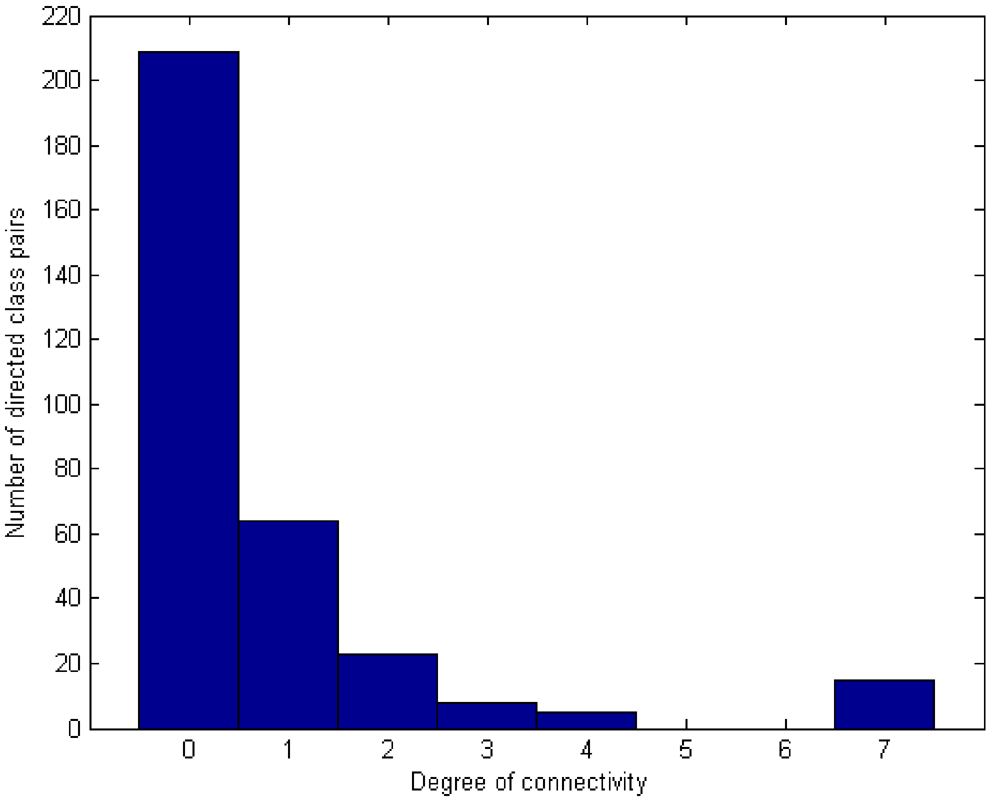


FIG. 9.3. distribution of connectivity degree among 18 classes (distance = 1)

tions even with short paths. Figure 9.4 shows distribution of connectivity degree among the same 324 *directed class* pairs with path length no larger than three. The transfer rate γ of the schema path model used to produce the histogram is 0.4. Two schema paths are the same only when all the following conditions are met: (1) the two paths have the same length (2) the class labels in the two paths exactly match (3) the property labels in the two paths exactly match (4) the directions of the properties in the two paths are the same. That is the reason why the degree of connectivity between some classes can be as large as thousands. The high degree of connectivity means a large number of mapping candidates, which is a challenge to both accuracy and efficiency of our approach. The schema path model with a

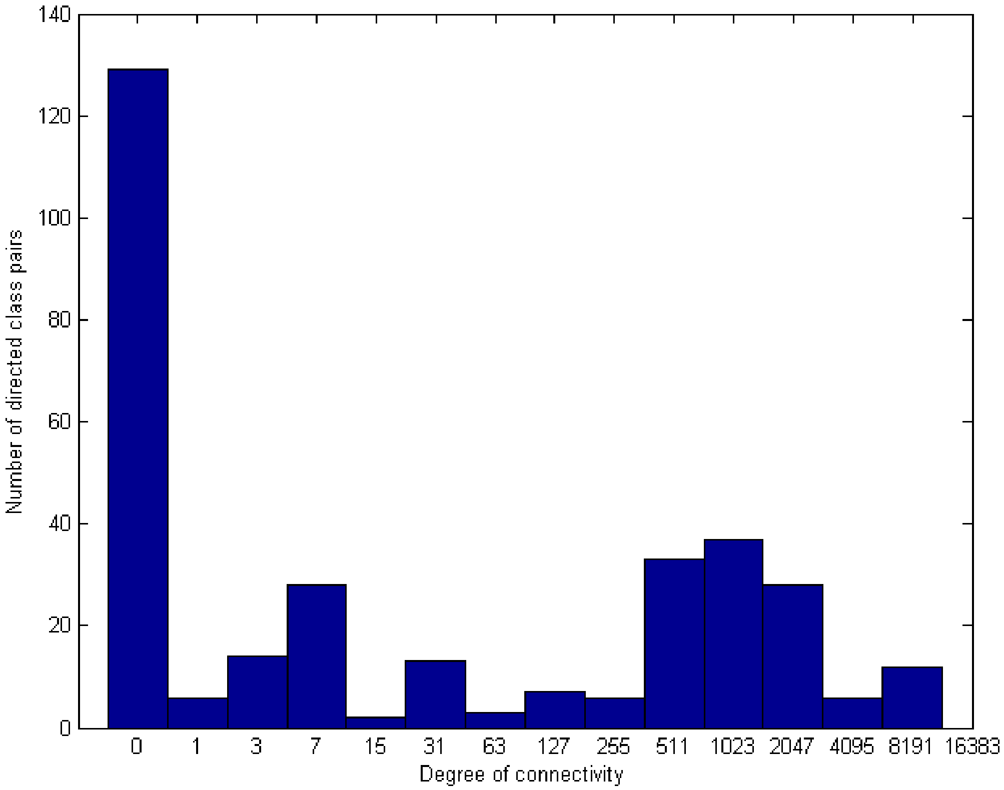


FIG. 9.4. distribution of connectivity degree among 18 classes (distance ≤ 3)

maximum path length of *three* is the one we used in the following experiments.

9.3.2 Query Set

We created 64 test questions on the DBLP+ dataset. In all these questions, we substitute variables for instance names. Here is one example: "*Give me author x of the paper y*". Since the mapping algorithms run at schema level and the instance names are not used, it is not necessary to include instance names for the purpose of resolving the parameters and evaluating the mapping algorithms.

The 64 test questions are shown in Table 9.4. The questions are classified into four categories: Direct Single (DS), Indirect Single (IS), Direct Multiple (DM) and Indirect Multiple (IM). **DS** means the question has a single relation which is a direct relation; **IS** means the question has a single relation which is an indirect relation; **DM** means the question contains multiple relations, all of which are direct relations; **IM** means the question contains multiple relations, at least one of which is an indirect relation. There are totally 31 DS, 15 IS, 8 DM, 10 IM questions in the collection. Of 25 IS and IM questions, four questions IS 5, IS 8, IS 9 and IS 14 have a relation with a path length of three. Of 18 DM and IM questions, five questions DM 8, IM 6, IM 7, IM 9 and IM 10 have three relations with four variables and all others have two relations with three variables.

We created 220 SFQs (schema-free structured queries) for the 64 test questions. We also call each SFQ a testcase. We get much more testcases than test questions because we tried many ways to rephrase the questions by substituting other terms. For example, seven SFQs are created for DS 1 "*give me author x of the paper y.*" as shown in the following list. The entire collection of 220 testcase is referenced in Appendix B. Most of the rephrasing is done for the DS and IS questions because many DM and IM questions consist of the relations of the DS and IS questions.

1. ?x/Author, has, ?y/Paper
2. ?x/Scholar, published ,?y/Paper
3. ?x/Person, author of, ?y/Paper
4. ?x/Person, wrote, ?y/Paper
5. ?x/Person, published, ?y/Paper
6. ?x/Person, has, ?y/Paper
7. ?x/Person, has, ?y/Publication

Our gold standard for the testcases are the correct interpretations of the SFQs. An interpretations is serialized into a string, following a specific syntax. For example, a cor-

Category & ID	Question
DS 1	give me author x of the paper y
DS 2	list paper x in the book y
DS 3	list paper y published in the conference x
DS 4	list paper y published in the journal x
DS 5	list InProceedings x in the proceedings y
DS 6	list subject y of paper x
DS 7	list subject y of conference x
DS 8	list subject y of venue x
DS 9	list subject y of journal x
DS 10	list subject y of author x
DS 11	list person x who published the book y
DS 12	give me publisher y of book x
DS 13	give me person x who is the second author of the article y
DS 14	give me person y who is the first editor of the proceedings x
DS 15	show me the email y of person x
DS 16	show me the homepage y of person x
DS 17	show me the abstract y of paper x
DS 18	show me in what series y the book x is
DS 19	list paper x that cites paper y
DS 20	show the number of citations y of author x
DS 21	show the number of publications y of author x
DS 22	list author x in the institution y
DS 23	list the proceeding y of the conference x
DS 24	give ISBN y of book x
DS 25	give volume number of a article
DS 26	give issue number of a article
DS 27	show page numbers y of a article x
DS 28	show institution y that published the paper x
DS 29	show the year y when a paper x is published
DS 30	show book x that is edited by person y
DS 31	give me publisher x of the proceedings y
IS 1	give conference x held in the year y
IS 2	show me in what series y the paper x is
IS 3	list person x who cites the paper y?
IS 4	list person x who is cited by the paper y?
IS 5	show person x who cites the person y
IS 6	give ISBN y of paper x
IS 7	list authors x and y who co-authored
IS 8	give country x that has paper in the journal y
IS 9	give country x that published paper in the conference y
IS 10	list author x from country y
IS 11	list the publisher y of the inProceedings x
IS 12	show me book x that is cited by person y
IS 13	show the author y of the conference proceedings x
IS 14	show person x who cites the proceedings y
IS 15	show author x who contributed to the conference y
DM 1	list the institution z of the author x who wrote the book y
DM 2	list paper x in the book y of ISBN z
DM 3	list person x who published the book y with ISBN z
DM 4	list paper x in the book y from publisher z
DM 5	list person x who published the book y of publisher z
DM 6	give me research area z of the authors x from the institution y
DM 7	list the number of citation z received by publication y in the conference x
DM 8	list paper x published in the issue z and volume u of journal y
IM 1	list the editor x of the conference y in the year z
IM 2	list all the papers x of the conference y in the year z
IM 3	give me papers x that are cited by venue y in the year z
IM 4	list the author x of the conference y in the year z
IM 5	Who are the authors z in the journal x with volume no. ?
IM 6	list paper z published by editor x of the conference y of the year u
IM 7	list the institutions u of the author y with whom the person x from the organization z has co-authored
IM 8	give me authors y and editors z of proceedings x
IM 9	give me research areas x of the papers y published by a company z in the year u
IM 10	give me the venues x of the papers y published by an organization z in the year u

Table 9.4. 64 test questions on DBLP+

rect interpretation of the query [*?x/Researcher, contributed to, ?y/Conference*] is shown as follows.

[Author $<$ InProceedings $>$ Conference]; [author, conference]

The single relation in the query is mapped to a schema path. Inside the first brackets are the classes on the schema path and the "$<$" or "$>$" between two classes shows the direction of the property connecting the classes. Inside the second brackets are the properties on the schema path, which are in order with "$<$" or "$>$" in the first brackets. If the SFQ contains multiple relations, the corresponding schema paths are separated by "$||$". The correct interpretation of the SFQ may not be unique. For example, the following three interpretations are also correct because they all lead to the same results.

1. [Person $<$ InProceedings $>$ Conference]; [author, conference]
2. [Author $<$ Paper $>$ Conference]; [author, conference]
3. [Author $<$ Publication $>$ Conference]; [author, conference]

9.3.3 Resolving parameters α and γ

As we discussed in Section 9.1.2, we can use sufficiently large numbers to set k_1, k_2 and k_3, which are the size of the class candidate list , the property candidate list and the concept mapping hypothesis list, respectively. For the DBLP+ dataset, we use 10 for k_1, 20 for k_2 and 40 for k_3. The DBLP+ dataset only have 31 classes and attribute types and 30 properties. On the other hand, our semantic similarity measures perform very well. Therefore, 10 and 20 are large enough numbers for k_1 and k_2. We make k_2 double size of k_1 because relation similarity is more difficult to measure than concept similarity. We experimentally found that our phase 1 algorithm is effective to rank correct concept mapping hypotheses to top places. Thus, 40 is quite a large number for k_3 on the DBLP+ dataset which only has dozens of classes and types.

We first apply grid searching to resolve the optimal values of α and γ while β is fixed with 0 and θ with 1[5]. Both α and γ have a range between 0.0 and 1.0 according to their definitions. We discretize the ranges of α and γ using steps of 0.05 and 0.10, yielding 21 and 11 choices respectively. Thus, totally 231 parameter candidates are included in our search space. The objective function we used is the MRR measure over the entire 220 test cases. Figure 9.5 shows the searching result using the hybrid semantic similarity model. The entire surface has high MRR scores falling between 0.879 and 0.945 except on the line where γ is 0, which produces a constant MRR score 0.706. If γ is 0, schema path probability will be 0.0 for all the paths having length larger than 1 and therefore all the testcases whose interpretation involves indirect relations will fail. The steep fall of MRR score at $\gamma = 0$ implies that significant portion of the 220 testcases involves indirect relations. The MRR function reaches its peak, 0.945, when α is 0.25 and γ is 0.4. The surface exhibits an overall tendency of going up towards this global maximum point.

To further analyze how α and γ affect performance individually, we show mean MRR along one dimension in Figure 9.8 and Figure 9.11. The height of a bar at a particular α or γ level represents the mean MRR averaged over all the candidates at that α or γ level. Figure 9.8 shows a smooth curve going up as α increases from 0 to 0.25 and go down thereafter. Figure 9.11 depicts a similar pattern where the highest mean MRR score is obtained at $\gamma = 0.4$ but the bars decline to a lesser extent after reaching the peak. The α and γ levels that maximize the performance of mean MRR in Figure 9.8 and Figure 9.11 are consistent with the optimized α and γ in Figure 9.5.

In order to compare our semantic similarity measures, we carry out the same grid searching experiment using the LSA semantic similarity model. Figure 9.6 presents the result of grid searching. Figure 9.9 and Figure 9.12 show the mean MRR along α and γ

[5] See Section 9.1.2 for the independence assumption we make to justify this optimization method

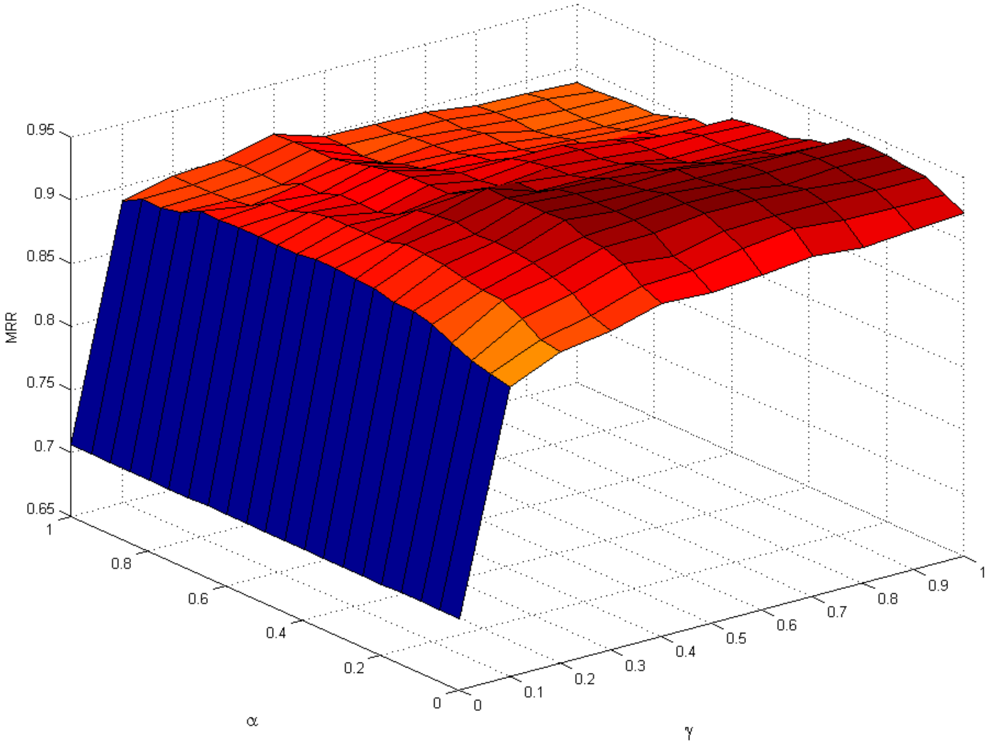


FIG. 9.5. Grid searching result using the hybrid semantic similarity model

dimensions respectively. In general, the LSA model performs slightly worse than the hybrid model. The surface in Figure 9.6 has MRR scores in the range $[0.839 \mathinner{..} 0.914]$ when $\gamma \neq 0$ and a constant MRR score 0.679 when $\gamma = 0$. The two highest MRR scores, both being 0.914, are reached at $(\alpha = 0.75,\ \gamma = 0.6)$ and at $(\alpha = 0.25,\ \gamma = 0.6)$. We can see that the surface at $(\alpha = 0.75,\ \gamma = 0.6)$ has a protrusion while the area around $(\alpha = 0.25,\ \gamma = 0.6)$ is flat. Thus, $(\alpha = 0.25,\ \gamma = 0.6)$ makes a better choice than $(\alpha = 0.75,\ \gamma = 0.6)$ because a protrusion could be caused by chance. This view is also supported by Figure 9.9 in which $\alpha = 0.25$ produces a better mean MRR score than $\alpha = 0.75$ does. The optimal α and γ levels for the mean MRR measure are 0.3 and 0.6, respectively, as shown in Figure 9.9 and Figure 9.12. Although these numbers are not the same as the ones produced by the

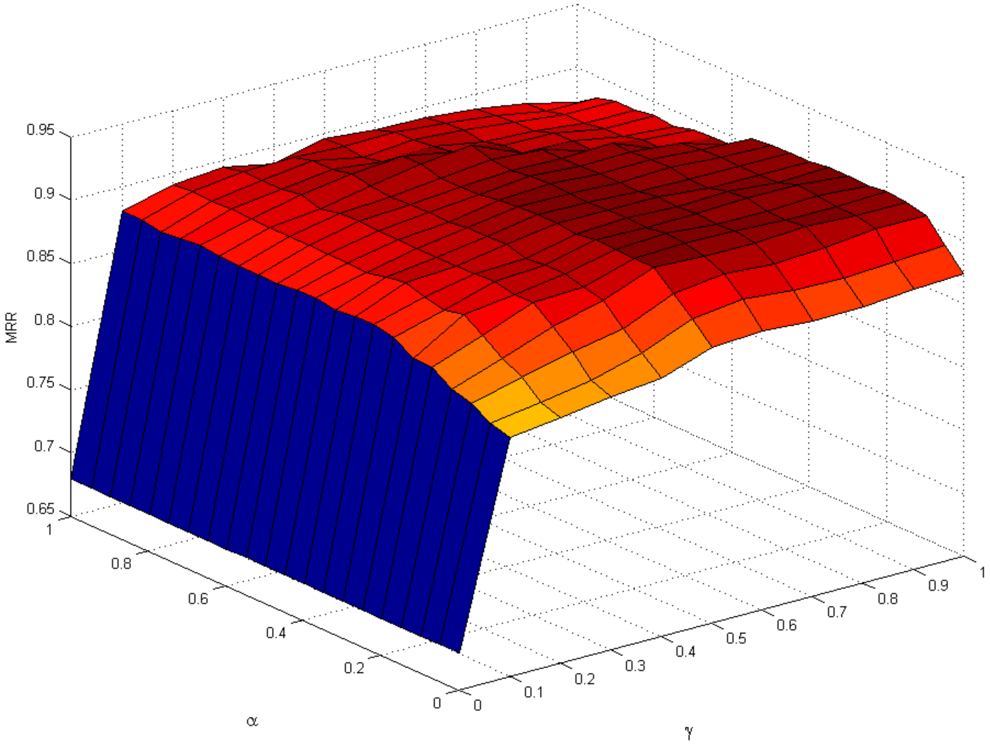


FIG. 9.6. Grid searching result using the LSA semantic similarity model

hybrid semantic similarity model, they are fairly close. Moreover, the patterns exhibited by the bars in Figure 9.9 and Figure 9.12 basically resembles the ones in Figure 9.8 and Figure 9.11.

We repeat the experiment again, replacing the semantic similarity model with the string similarity measure described in Section 8.1.3. This helps us to know how much we can gain by using semantic measures. The result of grid searching is shown in Figure 9.7. The MRR scores of the surface are in the range $[0.517 \mathinner{..} 0.533]$ when $\gamma \neq 0$ and are constantly 0.402 when $\gamma = 0$. The maximum MRR performance 0.533 is obtained at $(\alpha = 1,\ \gamma = 1)$. Figure 9.10 and Figure 9.13 show the mean MRR performance along single dimension α and γ, where the best mean MRR is achieved at $\alpha = 0.55$ and $\gamma = 1.0$,

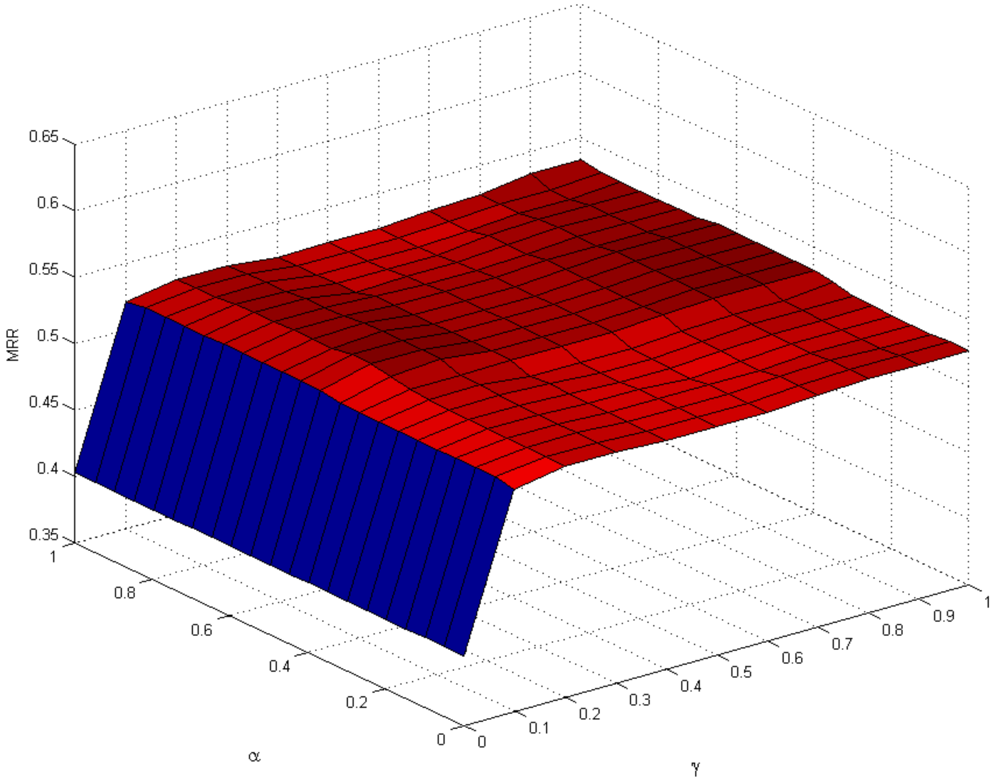


FIG. 9.7. Grid searching result using string similarity

respectively. The string similarity measure yields a much worse performance than the two semantic similarity measures. If we compare the best MRR performance of each measure, the hybrid and LSA semantic similarity measures improve upon the string similarity measure by 77% and 71%. It is worth mentioning that LSA, like string similarity, is a purely computational method that does not rely on any human-crafted knowledge.

The surface in Figure 9.7 is flat, which differs from the surfaces in Figure 9.5 and Figure 9.6. The bars in Figure 9.10 and Figure 9.13 also appear to be level with the exception at $\gamma = 0$. This indicates that the MRR performance tends to be independent of either α or γ if the string similarity is used. String similarity has roughly binary semantics, either

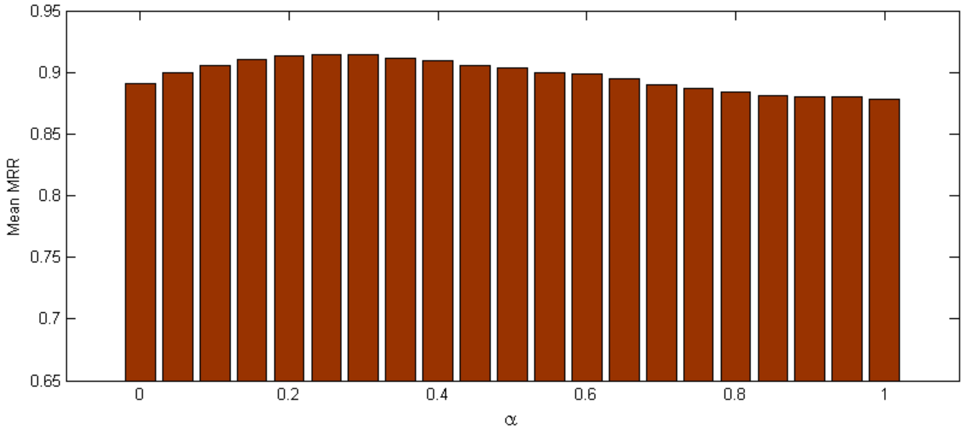


FIG. 9.8. Mean MRR along the α dimension using the hybrid semantic similarity model

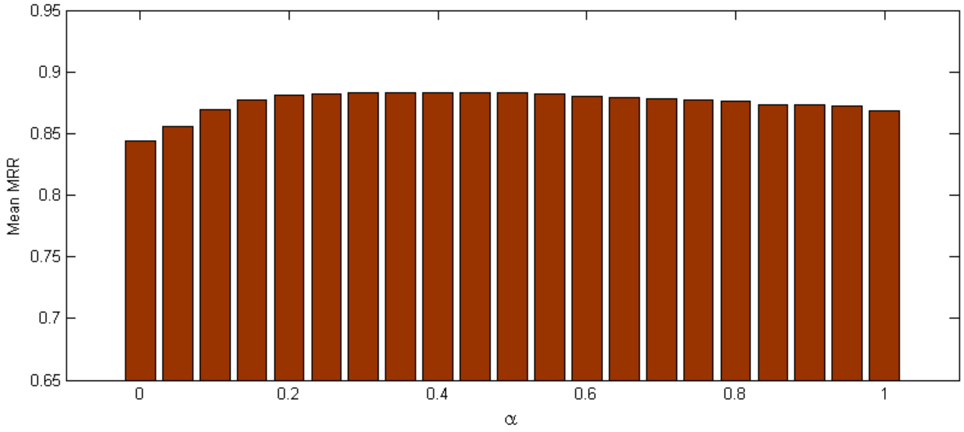


FIG. 9.9. Mean MRR along the α dimension using the LSA semantic similarity model

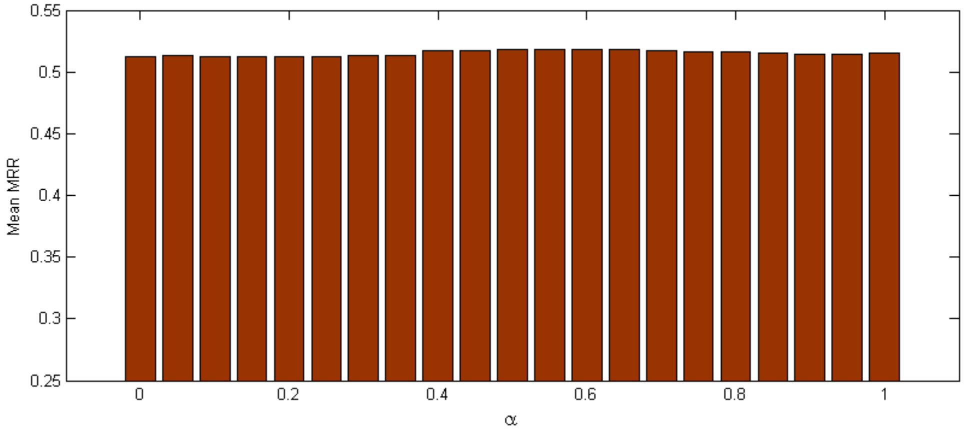


FIG. 9.10. Mean MRR along the α dimension using the string similarity measure

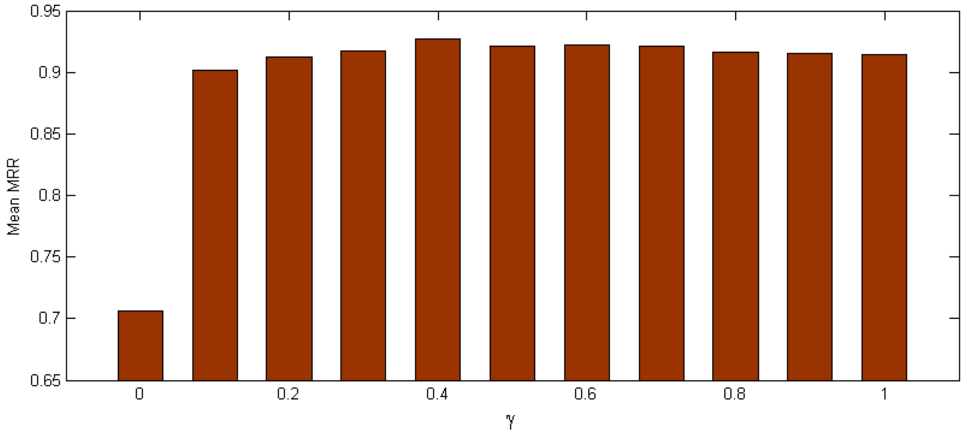


FIG. 9.11. Mean MRR along the γ dimension using the hybrid semantic similarity model

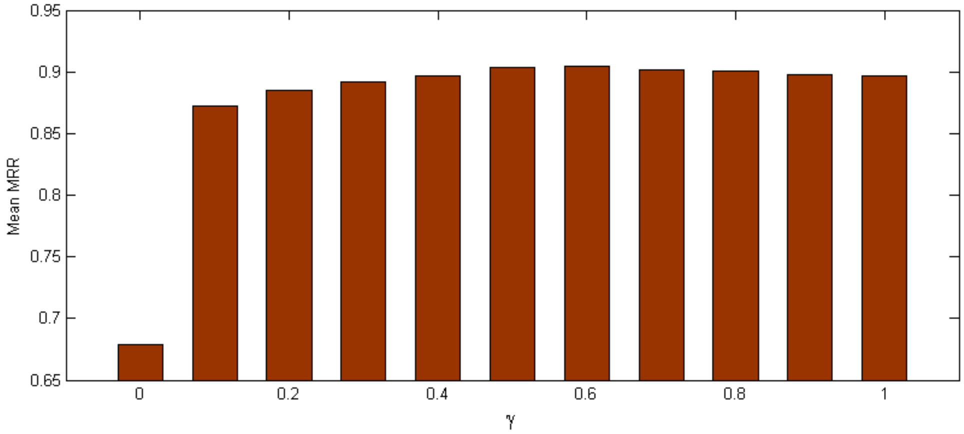


FIG. 9.12. Mean MRR along the γ dimension using the LSA semantic similarity model

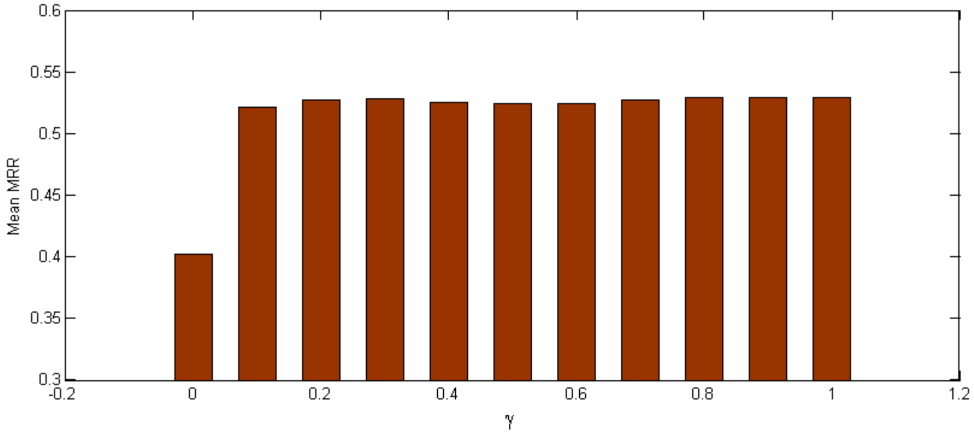


FIG. 9.13. Mean MRR along the γ dimension using the string similarity measure

true when two strings largely match or otherwise false. It lacks the ability to carry the degree of semantic similarity that α operates on. The binary nature of string similarity also makes similarity dominate over popularity in computing the overall ranking metric. On the other hand, γ only affects the computation of popularity so it has little effect on the overall performance.

9.3.4 Resolving parameter β

With the optimal values of parameters α and γ being resolved, we start tuning the parameter β. An assumption is made that selecting optimal value of β is independent of the value of θ, which allows us to still fix θ with 1.0. The parameter β is used to weigh *popularity* against *similarity* in the scoring metric in Equation 5.8. The larger β is, the less importance *popularity* plays. Figure 9.14, Figure 9.15 and Figure 9.16 give the plots showing how the MRR performance changes over different values of β, using hybrid, LSA and string similarity measures respectively. The range we select for β is $[-5 \mathinner{..} 8]$, with a step of 0.25. The left boundary is -5 because we want the term $log(Popularity) + \beta$ to be positive generally. The right boundary of 8 is large enough to show the tendency of the plots.

Figure 9.14, Figure 9.15 and Figure 9.16 see their highest MRR score 0.946, 0.914 and 0.539 at β level 2.25, 2.0 and 2.0, respectively. The plots in Figure 9.14 and Figure 9.15 share the MRR pattern of first climbing up and then tend to decline as β increases. The plot in Figure 9.16, however, does not show a tendency of decline. We also experimented with very large β levels. We found that MRR performance of the hybrid and LSA semantic similarity models drop to 0.831 and 0.866 respectively when β reaches $1,000$ and maintain these scores unchanged even as β go up to $1,000,000$. The string similarity also shows decline but very small, still having a MRR score 0.533 at $\beta = 1,000,000$.

When β becomes as large as $1000,000$, *semantics* dominates the relation ranking met-

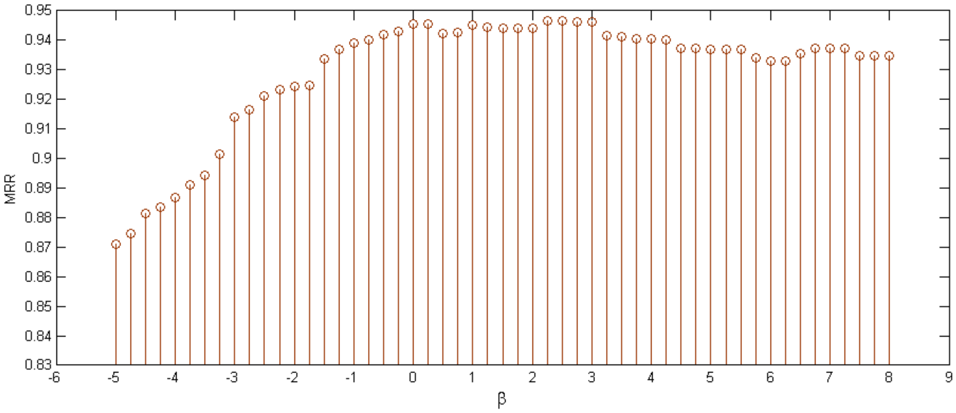


FIG. 9.14. MRR performance versus β, using the hybrid semantic similarity model

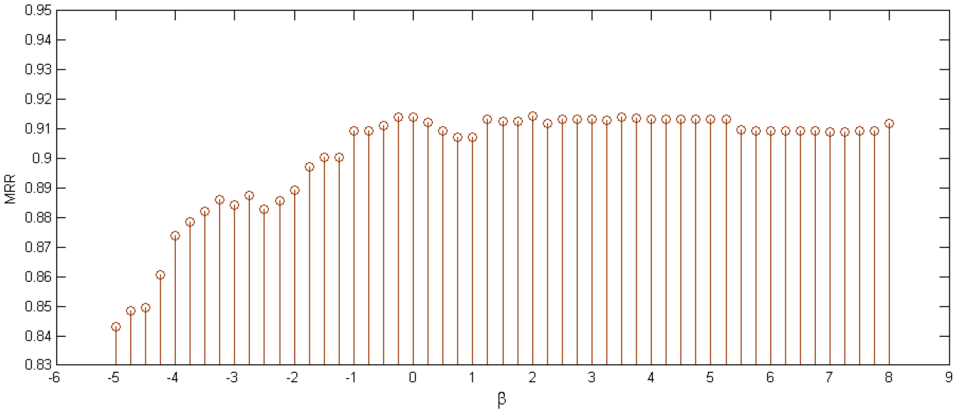


FIG. 9.15. MRR performance versus β, using the LSA semantic similarity model

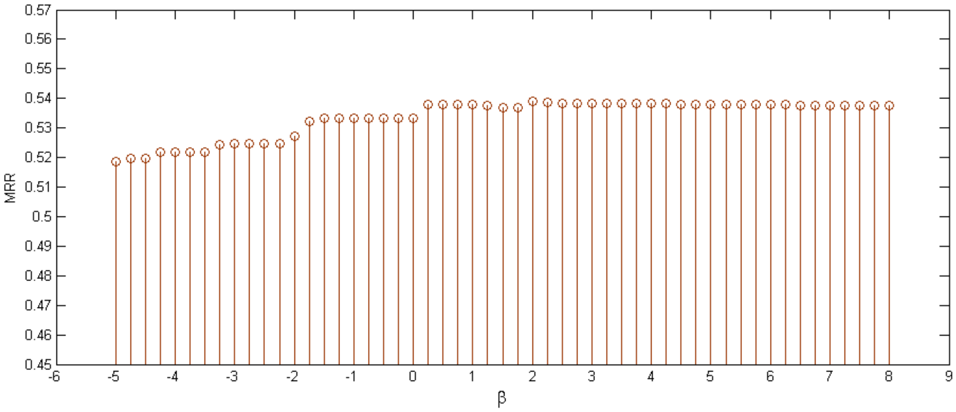


FIG. 9.16. MRR performance versus β, using the string similarity measure

ric in Equation 5.8 and *popularity* matters only if *semantics* scores from two candidates are tied. The fairly good performance of our semantic similarity models at very large β levels implies that *popularity* plays a less important role than *semantics* in Equation 5.8. Meanwhile, it also demonstrates that *popularity* is necessary for achieving better MRR performance.

9.3.5 Resolving parameter θ

The parameter θ addresses the problem of *default relations* that the mappings of the subject and object concept can be semantically uneven, resulting in unequal amount of information loss (See Section 5.3.3 for details). For a *default relation*, the empty predicate need to be filled in by either the subject or object concept. Due to the principle that the less information is lost the better the mapping is, the concept losing more information in the concept mapping is more preferred to substitute for the empty predicate. The two possible substitutions lead to two different similarity scores for a candidate path. The parameter θ controls how the two similarity scores are combined by considering different amount of information loss in mapping the subject and object concepts.

θ is the last parameter to tune because we believe it has the lest dependency to the other parameters. The range we set to tune θ is $[0 \mathinner{..} 4]$. When θ is 0, the combined similarity is whatever the larger one of the two similarity scores; when θ is positive infinite, the combined similarity is the one resulted by the substitution having more information loss in the concept mapping. Figure 9.17, Figure 9.18 and Figure 9.19 show that the hybrid semantic similarity, the LSA semantic similarity and the string similarity achieve their highest MRR scores 0.946, 0.925 and 0.539 at θ levels 1.0, 1.5 and 1.0, respectively. Our experiments with large θ levels (up to $10,000$) demonstrates that for all the three measures, the MRR performance keeps constant as θ goes beyond 4.0.

If the parameter θ were not introduced, a typical way to combine the two similarity

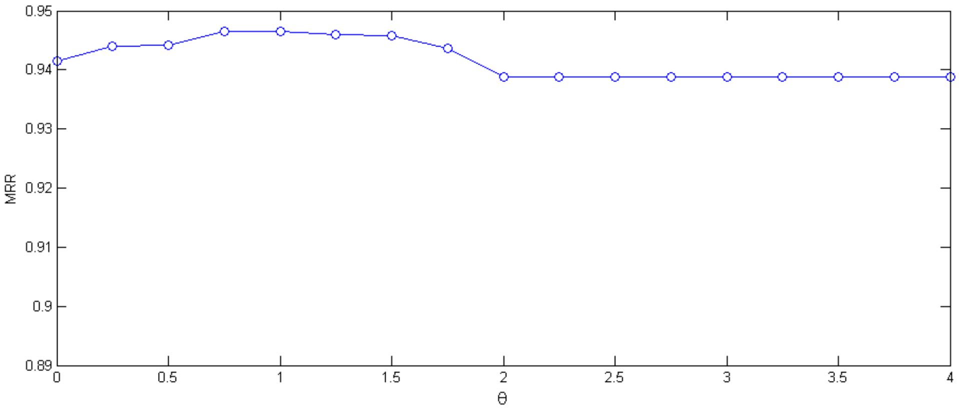


FIG. 9.17. MRR performance versus θ, using the hybrid semantic similarity model

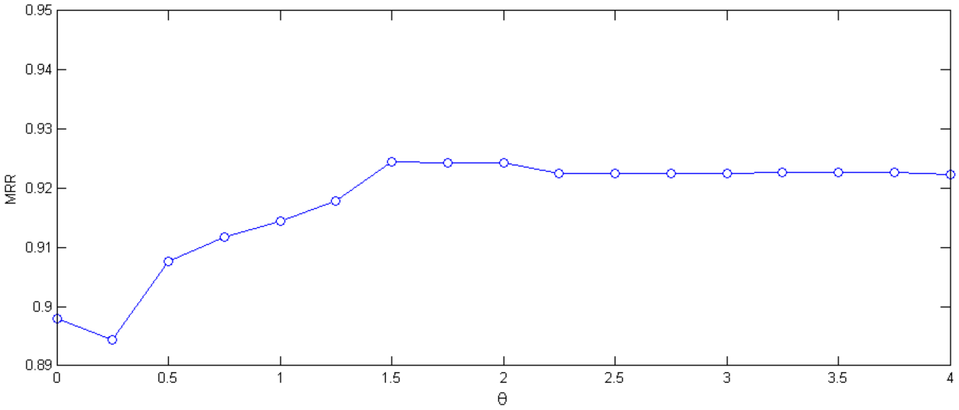


FIG. 9.18. MRR performance versus θ, using the LSA semantic similarity model

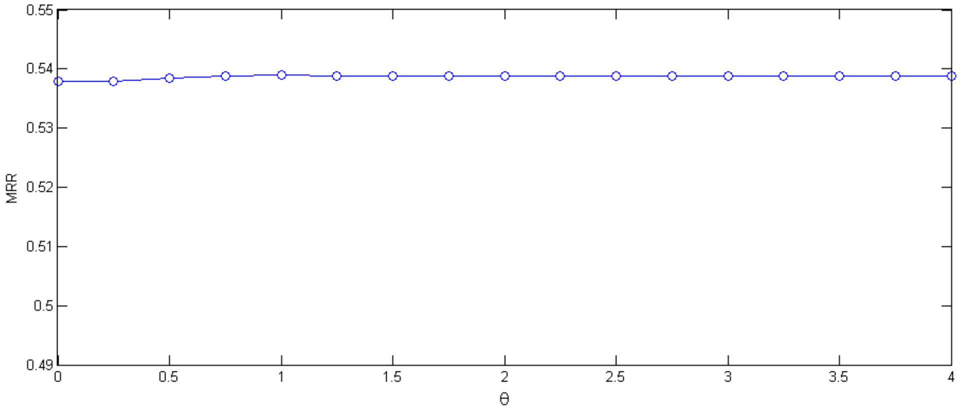


FIG. 9.19. MRR performance versus θ, using the string similarity measure

	α	γ	β	θ	MRR
hybrid semantic similarity	0.25	0.4	2.25	1.0	0.946
LSA semantic similarity	0.25	0.6	2.0	1.5	0.925
string similarity	1.0	1.0	2.0	1.0	0.539

Table 9.5. Optimal parameters and MRR performance for three similarity measures

scores from two substitutions would be to use whatever the larger one. It is exactly what our algorithm does when $\theta = 0$. Therefore, the plots also reveal how much performance we can improve by introducing θ. Among the three measures, the LSA semantic similarity has the largest improvement. The string similarity has almost no improvement, which could again be ascribed to its binary semantics.

9.3.6 Results using optimal parameters

The resolved optimal parameters as well as their corresponding MRR performance are summed up in Table 9.5. The parameters k_1, k_2 and k_3 are manually set to be 10, 20 and 40, respectively. The hybrid semantic similarity has the highest MRR performance 0.946, a 2.3% improvement over the LSA semantic similarity and a 75.5% improvement over the string similarity. The optimal MRR score of the LSA semantic similarity is 0.925, which improves upon that of the string similarity by 71.6%.

We can have more insight into the performance by examining the distribution of the ranks over the 220 testcases that the MRR score combines. The distribution is provided in Table 9.6 for each of the three semantic measures. For the hybrid semantic similarity, the correct interpretations of 202 testcases are ranked at 1st place and only one falls out of top 10, making a top-10 coverage of 99.5%. This means the user would always find answers to her questions if she could look through all the top 10 interpretations. Moreover, she can still have a 95.5% chance to find correct answers if she only check the top 2 interpretations. The LSA semantic similarity works slightly worse, with a 98.2% top-10 coverage and 93.6%

Rank	**hybrid**	**LSA**	**string**
1	202	196	115
2	8	10	5
3	3	3	2
4	1	1	1
5	4	5	0
6	1	0	1
7	0	1	0
8	0	0	0
9	0	0	0
10	0	0	0
> 10	1	4	96

Table 9.6. Distribution of ranks produced by the best runs of three similarity measures

top-2 coverage. The string similarity has the worst performance, with a 56.4% top-10 coverage and 54.5% top-2 coverage.

The individual reciprocal ranks of the entire 220 testcases are provided as bars in Figure 9.20, Figure 9.21 and Figure 9.22 for the hybrid, LSA and string similarity measures respectively. The bars of the testcases in four categories (DS, IS, DM and IM) are marked with four different colors. Since most of testcases for the hybrid and LSA semantic similarity measures have perfect reciprocal ranks, white slots in the figures indicates the testcases whose correct interpretations failed to be ranked at 1st place. Many of white slots in Figure 9.20 and Figure 9.21 are overlapped, a reflection of the fact that the hybrid similarity is built upon the LSA semantic similarity.

However, there are also some testcases for which the LSA similarity performs better than the hybrid similarity. This manifests that knowledge in WordNet can sometimes cause errors to a particular domain. For example, the verb “publish” has a LSA similarity score 0.32 with “editor” and a score 0.41 with “author”. However, because in WordNet the verb “publish” is a direct hypernym of the second sense of the verb “edit”, the hybrid similarity between “publish” and “editor” increases to 0.56, becoming larger than the one between "publish" and "author". Due to this change the hybrid similarity maps *published* to *editor*, instead of *author*, in its first interpretation and receives a lowered reciprocal rank on the testcase [*?x/Person, published, ?y/Book*].

The white slot occurring rates in DS, IS, DM and IM categories for the hybrid similarity are 0.07, 0.14, 0.1 and 0.0, respectively. As expected, testcases in DS performs better than those in IS because testcases for direct relations are easier to deal with than those for indirect relations. However, it may be counterintuitive that DM and IM perform so good compared with DS and IS categories since multiple-relation questions are more complicated than single-relation questions. There are two reasons accountable for this. First, we include many different ways to rephrase questions for DS and IS but much less for DM and IM. Second, multiple-relation questions benefit from joint disambiguation and therefore enjoy better performance in disambiguating each single relation. For example, a relation [*?x/Paper, in, ?y/Book*] by itself is ambiguous to machines and the correct interpretation is ranked at the 3rd place using the hybrid similarity as shown in the following list.

1. [Paper $>$ Journal]; [journal]
2. [Paper $>$ Journal $<$ Publication]; [journal, journal]
3. [Publication $>$ Book]; [book]

Paper in our ontology is defined as refereed papers, including journal articles and conference inProceedings but not book chapters. Thus, the concept "Paper" in the query should not be mapped to the *Paper* in the ontology. However, because "book" is semantically similar to "journal"[6] and $Paper \overset{journal}{\rightarrow} Journal$ makes a reasonable schema path, the query is mistakenly interpreted as "give paper x in journal y". This disambiguation error can be avoided when the relation appears with another relation in the multiple-relation testcase [*?x/Paper, in, ?y/Book; ?y/Book, has, ?z/ISBN*]. Since in our DBLP+ dataset *Book* has the

[6] The hybrid similarity model tells a similarity score 0.53 between book and journal

	average testcase runtime
hybrid semantic similarity	0.259 seconds
LSA semantic similarity	0.349 seconds
string similarity	0.203 seconds

Table 9.7. Average testcase runtime for three similarity measures

ISBN property and *Journal* has not, the concept “Book” is then correctly mapped to the class *Book*, thus making *Publication* a better choice to be mapped to than *Paper*.

Figure 9.23, Figure 9.24 and Figure 9.25 show runtime of each of the 220 testcases for the hybrid, LSA and string similarity. The runtime of the DS testcases keeps at the same level as the IS testcases because our system does not have prior knowledge about whether the relation in question is direct or indirect and it runs the exactly same algorithm to map the relation. In general, the testcases in DM and IM requires more execution time than those in DS and IS because there are more than one relation to deal with in each query in the DM and IM categories. However, there are many DS and IS testcases that took more time to run than many of DM and IM testcases. We can also observe high variance of the runtime within each category. These phenomena demonstrate another important factor influencing the runtime – the connectivity degree among the classes. As shown in Figure 9.4, the connectivity degree exhibits a high variance, varying from a few connectivities to thousands. If a query interpretation requires checking paths between two heavily connected classes, it will take much more time than those who do not.

The average time to run a testcase for three similarity measures is shown in Table 9.7. The string similarity runs faster than the other two because it produces significantly smaller class and property candidate lists. The reason why the hybrid similarity took less time than the LSA similarity is not very clear. It might have something to do with the normalization of the hybrid similarity which makes the hybrid similarity model produces generally lower similarity score than the LSA similarity model. It is also observed that in some testcases

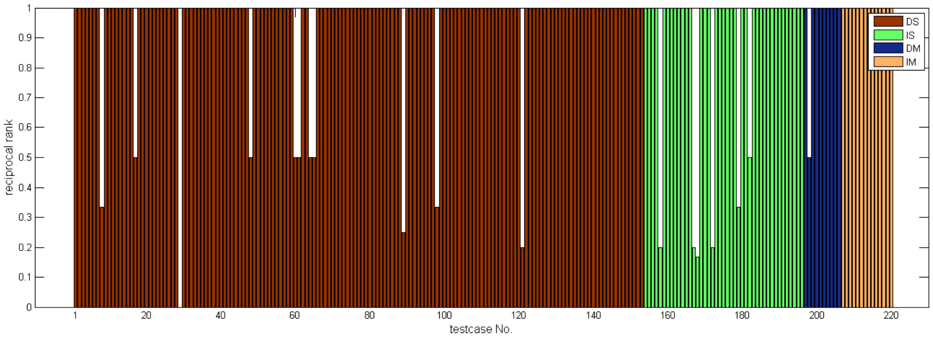


FIG. 9.20. Reciprocal ranks of 220 testcases in four categories using the hybrid similarity

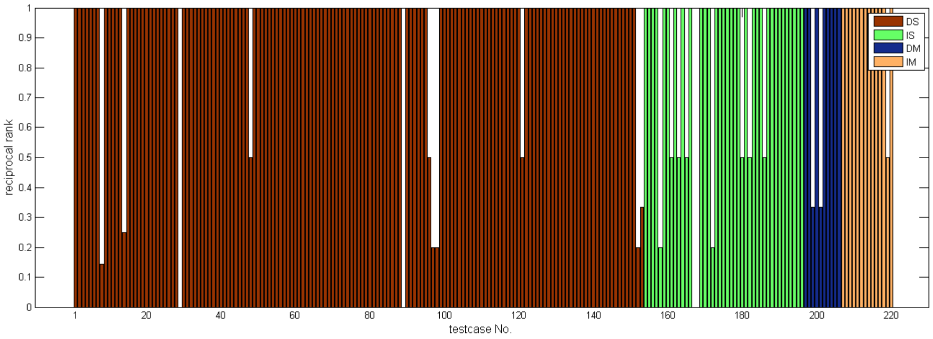


FIG. 9.21. Reciprocal ranks of 220 testcases in four categories using the LSA similarity

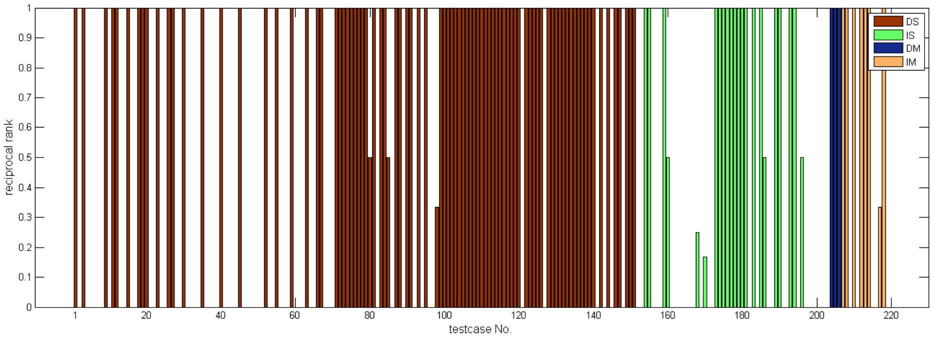


FIG. 9.22. Reciprocal ranks of 220 testcases in four categories using the string similarity

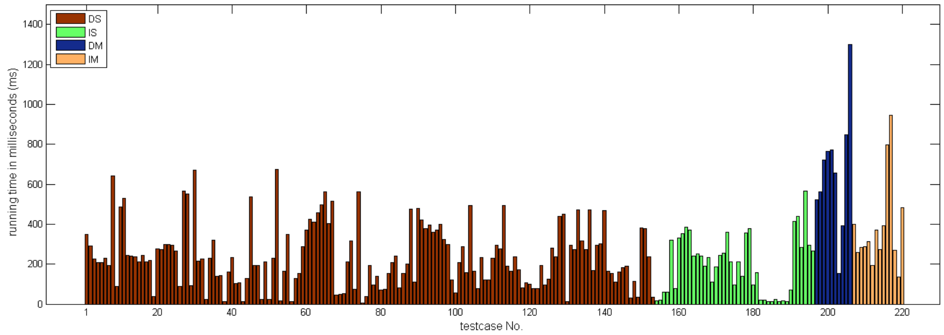


FIG. 9.23. Runtime of 220 testcases in four categories using the hybrid similarity

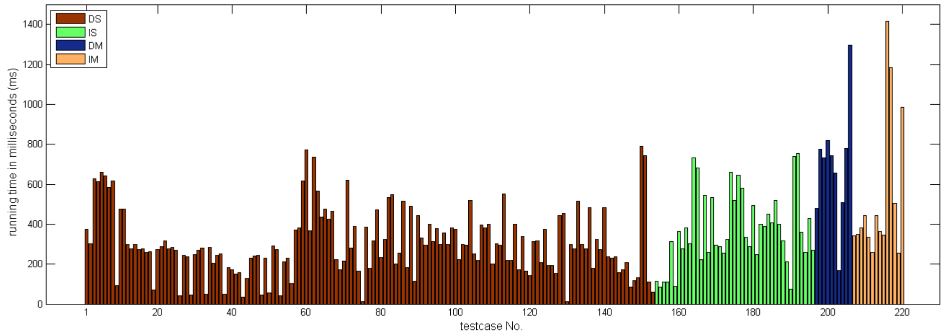


FIG. 9.24. Runtime of 220 testcases in four categories using the LSA similarity

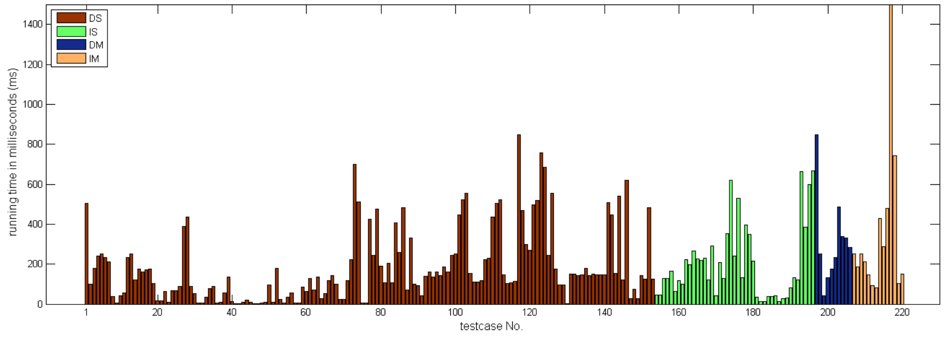


FIG. 9.25. Runtime of 220 testcases in four categories using the string similarity

the class *Thing* was included in the candidate list of the LSA similarity with a score just above threshold and did not appear in the corresponding list of the hybrid similarity.

For those who want to have more complete view on the performance, we provide the top 10 interpretations of all 200 testcases using the hybrid similarity in Appendix B. It is very informative to see what else interpretations are ranked at top places besides the correct one. It helps understand what ambiguity confuses our system and what problems our approach is challenged by.

9.3.7 Performance analysis on k_3

To make the experiments feasible, we manually chose the values for the parameters k_1, k_2 and k_3, which stand for the size of the class candidate list, the property candidate list and the concept mapping hypothesis list, respectively. Now the values of parameters α, γ, β, θ are resolved and it is time to see how performance is impacted by using different k values. However, since the DBLP+ dataset only has a few dozens of classes/types and properties, we skip the performance analysis on k_1 and k_2 because it would not produce an accurate picture if the lists are often not fully occupied.

Figure 9.26, Figure 9.27 and Figure 9.28 show how the MRR performance varies against k_3 using the hybrid, LSA and string similarity, respectively. The range of k_3 is from 1 to 40. All figures exhibit the same pattern that the MRR performance keep climbing rapidly at the very beginning, then slow down and finally maintain the top level as k_3 becomes increasingly large. The figures demonstrate that 40 is indeed a sufficiently large number for setting k_3 in the experiments of resolving α, γ, β, θ.

We can also see the efficacy of the phase 1 algorithm in the figures. The more effective the phase 1 algorithm is, the smaller k_3 is required to reach the full potential of MRR performance. When k_3 is 5, the MRR score of the hybrid, LSA and string similarity is 0.936, 0.911 and 0.534, achieving 98.9%, 98.5% and 99.1% of their full potential respectively.

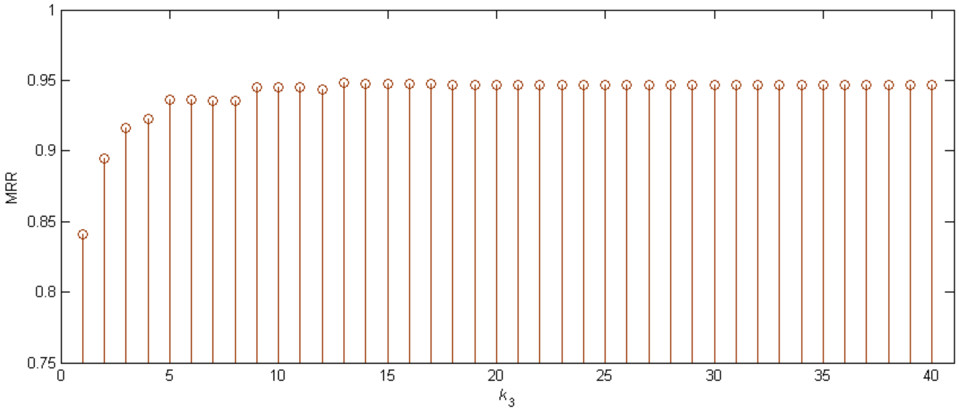


FIG. 9.26. MRR performance versus k_3, using the hybrid similarity

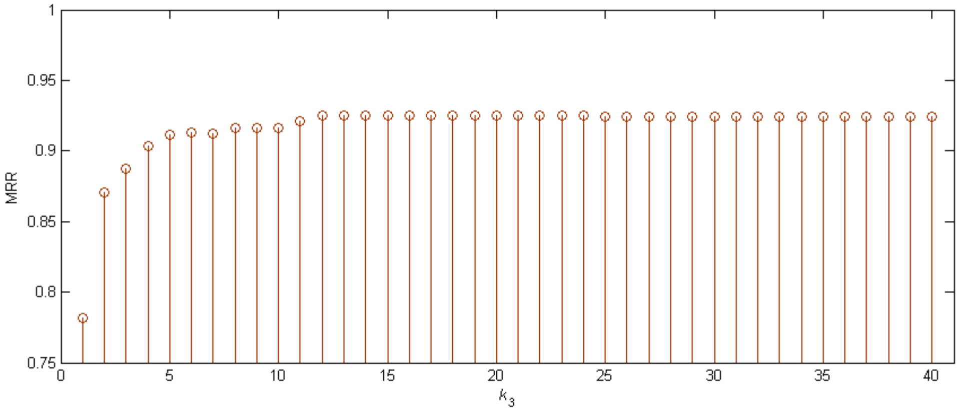


FIG. 9.27. MRR performance versus k_3, using the LSA similarity

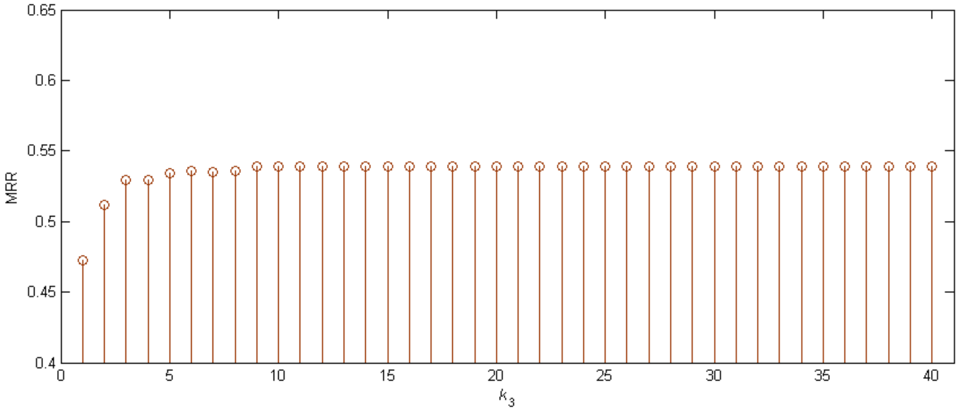


FIG. 9.28. MRR performance versus k_3, using the string similarity

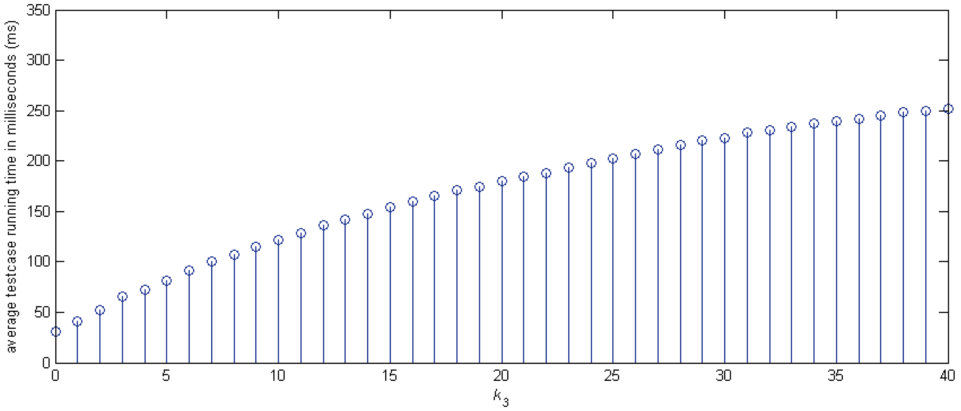


FIG. 9.29. Average testcase runtime versus k_3, using the hybrid similarity

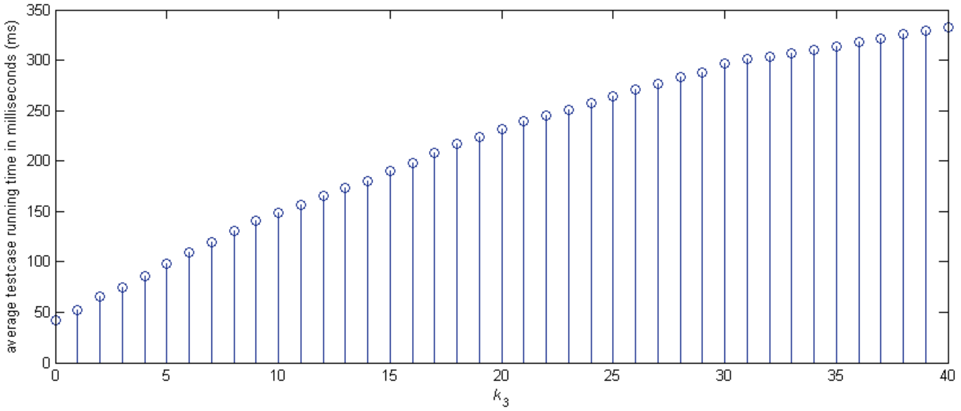


FIG. 9.30. Average testcase runtime versus k_3, using the LSA similarity

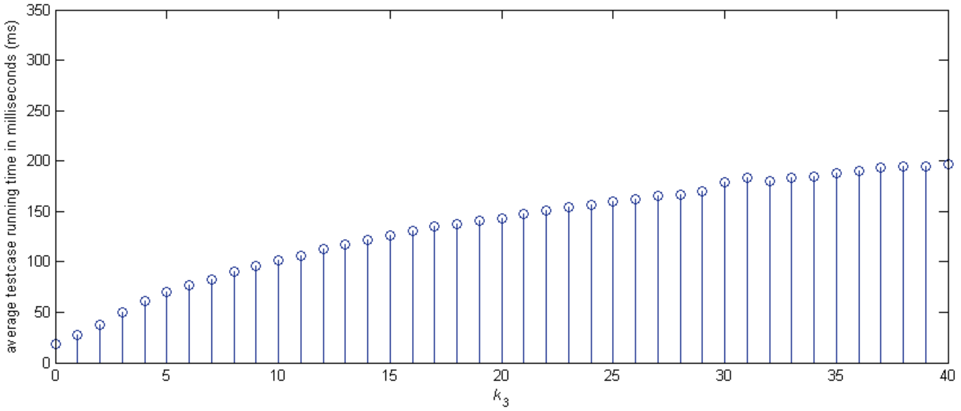


FIG. 9.31. Average testcase runtime versus k_3, using the string similarity

These numbers tell that our phase 1 algorithm is so effective that we may use 5 for k_3 and own a much faster system with almost no loss of performance.

The relationship between k_3 and average testcase runtime is pictured in Figure 9.29, Figure 9.30 and Figure 9.31. All the figures exhibit a relationship close to linear. The intercept at $k_3 = 0$ represents the time that the phase 1 algorithm takes. Thus, majority of time is consumed by the phase 2 algorithm unless k_3 is a very small number. This shows that having an effective phase 1 algorithm is crucial to make the system run faster. The intercept or the time spent by the hybrid, LSA and string similarity on the phase 1 algorithm is 0.031, 0.042 and 0.018 seconds, respectively. The string similarity took the least time because their class and property candidate lists are the most under-occupied.

9.3.8 Compare improved PMI with standard PMI

In the phase 1 algorithm, we use the improved PMI to measure association degree between schema terms. In order to demonstrate that the improved PMI produces a more effective phase 1 algorithm than the standard one, we carry out the same experiment as that in Figure 9.26 using the hybrid similarity but substitute the standard PMI for the improved PMI. The comparison result is shown in Figure 9.32. The improved PMI has a consistently better MRR performance than the standard PMI on all k_3 levels from 1 to 30. Their MRR difference finds its largest, 0.122, when $k_3 = 2$ and gradually decreases as k_3 becomes increasingly larger. However, there is still a considerable gap, 0.057, even when $k_3 = 30$. This comparison supports that the improved PMI is a better measure for computing association degree than the standard PMI.

9.3.9 Cross-validation

In order to avoid potential overfitting, we perform two-fold cross-validation on the 220 testcases. We randomly split the 64 test questions in half, resulting in 117 testcases in one

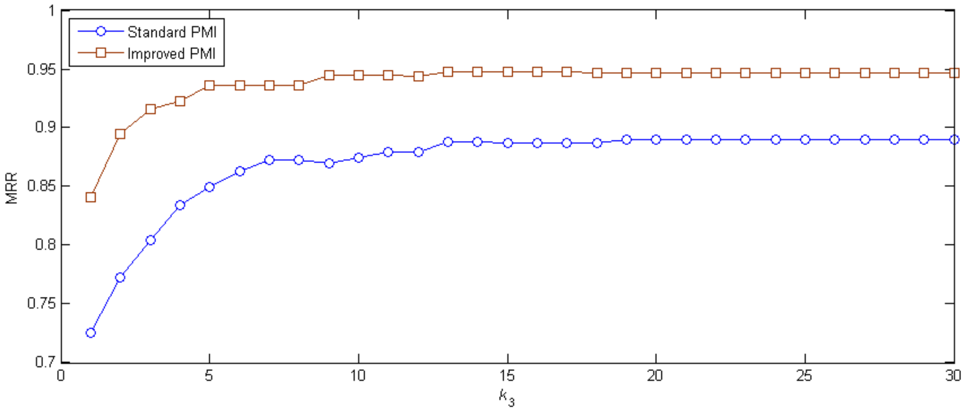


FIG. 9.32. Comparing improved PMI with standard PMI on the DBLP+ dataset

test set A and 103 testcases in the other test set B, as shown in Table 9.8. Our choosing of randomly splitting the test questions rather than the testcases is after careful consideration. Many testcases are rephrasing of the same question. If we randomly break the 220 testcases in half, the resulting datasets will be too similar in terms of the questions they contain and will not well serve the validation purpose that a system trained by a set of questions can successfully answer questions it has not seen.

	number of test questions	number of testcases
test set A	32	117
test set B	32	103

Table 9.8. The split datasets for two-fold cross-validation

The result of two-fold cross-validation using the hybrid similarity is shown in Table 9.9. System A is trained on the testcase collection A and is tested on the other collection B. System B is trained on B and tested on A. Training here means using the same method described earlier to optimize α, γ, β and θ. The MRR test results averaged over the two folds is 0.933, which degrades only 1.37% from 0.946, the optimal MRR score obtained without partitioning the 220 testcases.

	α	γ	β	θ	MRR on test set A	MRR on test set B
System A	0.25	0.4	-1.0	1.5	0.951	0.931
System B	0.25	0.6	0	1.0	0.935	0.955

Table 9.9. The result of two-fold cross-validation

The sets of parameters learned by System A and B are not completely the same, and they also differ from the parameters learned by the optimal system using the entire 220 testcases shown in Table 9.5. The parameter α of 0.25 is agreed by all the three systems. The parameter γ has two close values 0.4 and 0.6, both of which produce highest mean MRR scores in Figure 9.11. The parameter β has three values -1, 0, 2.25, which, though showing considerable variation, still fall in the range that yields top MRR scores in Figure 9.14. The parameter θ has two values 1.0 and 1.5, both of which produce top MRR scores in Figure 9.17.

9.4 Evaluation on DBpedia

In the previous section, we trained our system using 220 testcases on the DBLP+ dataset. In this section, we first test the trained system on the DBpedia dataset, which we call cross-domain validation. We then carry out performance analysis on some of the parameters. Finally we compare our system with other systems using precision and recall measures and discuss the results.

9.4.1 Data

DBpedia, representing data from Wikipedia, is the key component of the Linked Open Data (LOD) and serves as a microcosm for larger, evolving LOD collections. DBpedia has been regularly updated every several months and the version we use to evaluate our system is DBpedia 3.6. DBpedia provides a broad-based, open domain but shallow ontology. Our evaluation dataset only includes data represented under the DBpedia ontology and excludes data from other ontologies and raw infobox data. More specifically, our dataset consists of two datasets downloaded from the DBpedia website [25], *Ontology Infobox Properties* and *Ontology Infobox Types*. The first dataset contains RDF triples (i.e. relations between the instances) and the second provides all type definitions for the instances. It is worth noting that we did not use the dataset *DBpedia Ontology* from the website that specifies the class hierarchy and human-crafted domain and range definitions for properties. Instead, we obtain such kind of knowledge from our CAK model that is automatically built from data.

Heterogeneity is a problem of the DBpedia ontology because it supplants the categories and attribute names of Wikipedia infoboxes, which were independently designed by different communities. First, Terms having similar linguistic meanings are used for different contexts. For example, the property *locatedInArea* is for mountains and the property *location* is for companies. Second, the same term is loaded with multiple senses. For example, the *author* can mean book author or website creator. Third, there are also synonymous or nearly synonymous properties with the same domain, such as *citizenship* and *nationality*. Last, properties that are not linguistically synonymous can mean the same relation in some contexts. In *a company producing software*, for example, several properties are used: *publisher, developer, designer, product* and *author*. While DBpedia is a single ontology, it is somewhat similar to the situation where independently developed domain-specific ontologies are combined.

Table 9.10 shows the statistical properties of our dataset. 250 classes and 1,099 properties from the DBpedia ontology are actually used in this dataset. Besides, we create 133 attribute types and 140 inferred classes. Five of the attribute types *$\hat{N}$umber*, *$\hat{D}$ate*, *$\hat{Y}$ear*, *$\hat{T}$ext* and *$\hat{L}$iteral* are predefined and the other 128 attribute types are automatically learned from the 515 datatype properties. This is in line with what we do on the DBLP+ dataset. On the other hand, the 140 inferred classes are automatically derived from the 584 object

Statistical property	**Value**
Number of types	523
· ontology classes	250
· inferred classes	140
· attribute types	133
Number of properties	1,099
· object properties	584
· datatype properties	515
Number of relations (triples)	13,465,950
Number of type definitions using ontology classes	6,173,940
· Number of instances defined	1,668,215
Number of type definitions using ontology classes and inferred classes	7,510,729
· Number of instances defined	2,165,296

Table 9.10. DBpedia dataset statistics

properties. We do so for addressing the problem that the DBpedia ontology classes are incomplete for describing the DBpedia data. For example, many instances do not have any type definition because the appropriate classes do not exist in the ontology. Many property names are nouns or noun phrases, which can be used to infer the type of the object instance. For an instance, the object of the *religion* property should be a religion. A few other examples of the inferred classes include *Producer*, *Director*, *Battle* and *Capital*. We derived attribute types and inferred classes only from the properties used in at least 1,000 relations (triples) in order to filter out insignificant and noisy types. For the comparison's purpose, we will show experiment results with and without inferred classes in the following sections.

If we only consider direct relations, the degree of connectivity between the 249 DBpedia ontology classes[7] is low. Figure 9.33 shows the connectivity degrees of 62,001 class pairs resulted from pairing every $\leftarrow C_i$ with every $\rightarrow C_j$ where i $\in$ 1..249 and j $\in$ 1..249. The degree of connectivity between two *directed classes* $\leftarrow C_1$ and $\rightarrow C_2$ is defined as the

[7] The most general class *Thing* is excluded.

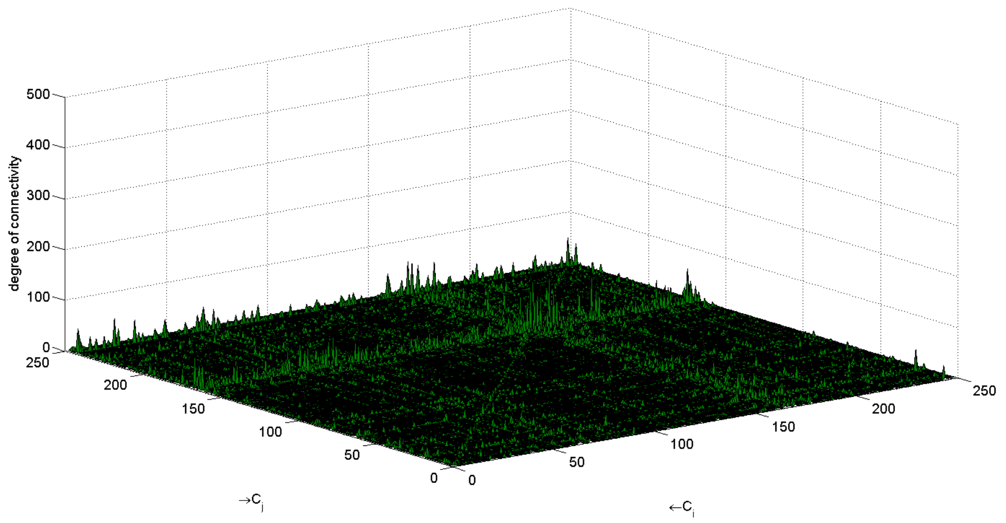


FIG. 9.33. connectivity degree among 249 ontology classes (path distance = 1)

number of distinct schema paths that connect $\leftarrow C_1$ and $\rightarrow C_2$. When it comes to direct relations, it is equivalent to the number of distinct properties that can go from C_1 to C_2 but not including the other way around. Figure 9.33 indicates that the 62,001 *directed class* pairs are either not connected or loosely connected. The highest connectivity degree is 131, found between $\leftarrow Place$ and $\rightarrow Place$.

However, the degree of connectivity increases drastically as we include indirect relations even with very short paths. Figure 9.34 shows the connectivity degrees among the same 62,001 *directed class* pairs with path length no larger than *two*. Inferred classes are not included in the schema path model used to produce the plot. The transfer rate γ of the model is 0.4, which is the optimal parameter we learned on the DBLP+ dataset using hybrid similarity. The highest connectivity degree, found between $\leftarrow Person$ and $\rightarrow Place$, is $59,737$, which is hundreds of time larger than the one produced for direct relations. The size of the statistical model file also increases from 1.7 MB to 39 MB.

If we allow inferred classes, the connectivity degree becomes even much larger gen-

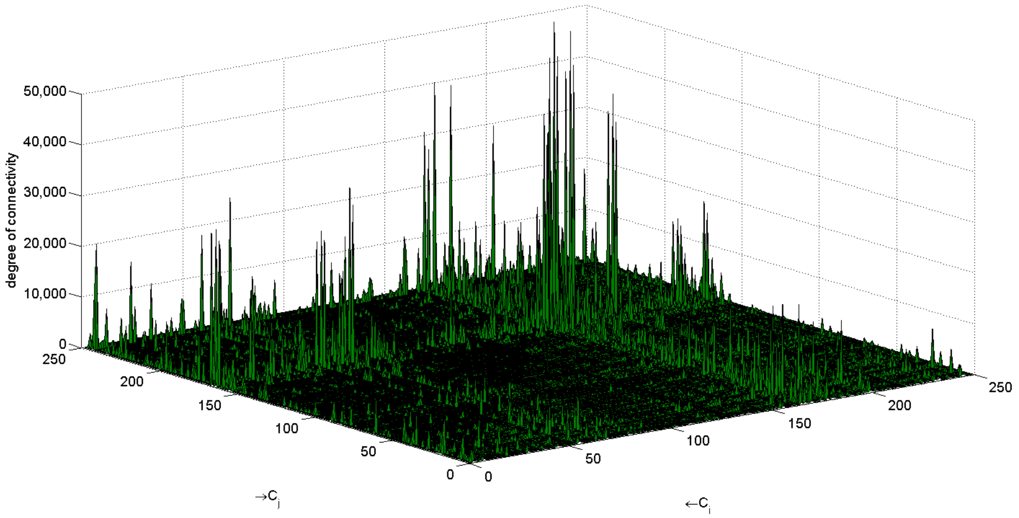


FIG. 9.34. connectivity degree among 249 ontology classes (path distance $\leq$ 2 and inferred classes are *not allowed* in the path)

erally. Figure 9.35 shows the connectivity degrees among the same 62,001 *directed class* pairs. The maximum path length and the transfer rate γ are still *two* and 0.4 but this time the inferred classes are allowed to occur in the path. The highest connectivity degree increases to $274,853$, which is found between $\leftarrow Person$ and $\rightarrow Place$. The size of the statistical model file increases to 220 MB.

The connectivity degree is so high that it demands a lot of computation and could result in slow system response time. We can address this problem by pruning schema paths using a minimum path frequency threshold. Figure 9.36 shows the connectivity degrees among the 62,001 *directed class* pairs when we set the threshold to 20. The connectivity degrees reduce dramatically. The highest connectivity degree decreases to $45,632$, a reduction of 83%. However, we did not apply this pruning in the following experiments in order to show worst case running time.

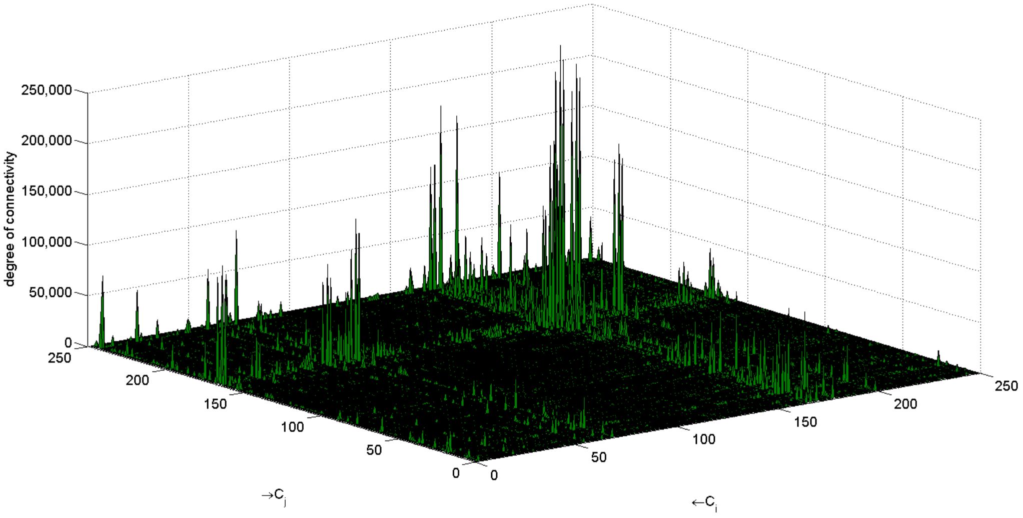


FIG. 9.35. connectivity degree among 249 ontology classes (path distance ≤ 2 and inferred classes are *allowed* in the path)

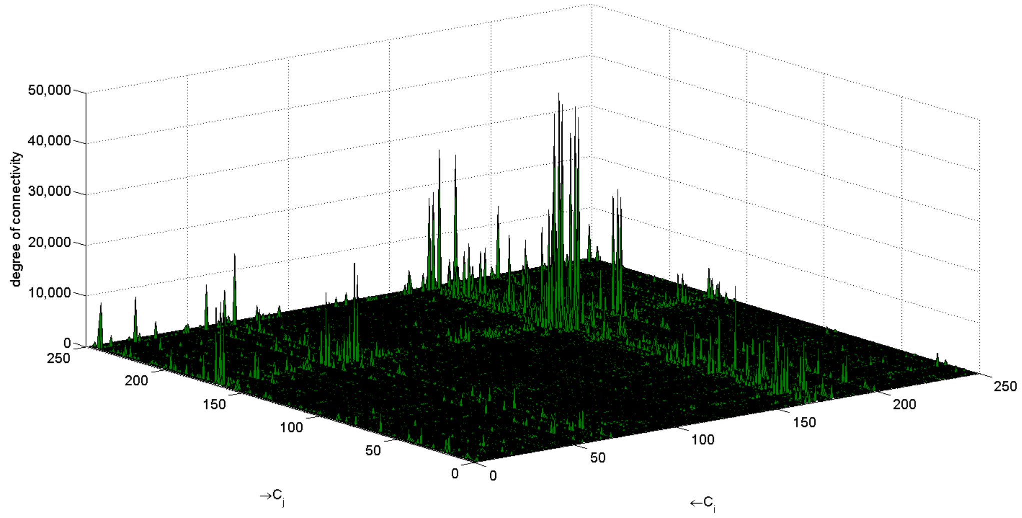


FIG. 9.36. pruned connectivity degree among 249 ontology classes (path distance ≤ 2 and inferred classes are *allowed* in the path)

9.4.2 Query Set

We evaluated our system using a dataset developed for the 2011 Workshop on Question Answering over Linked Data (QALD) [83]. This dataset was designed to evaluate ontology-based question answering (QA) systems and includes 100 natural language (NL) questions (50 training and 50 test) over DBpedia 3.6 along with the corresponding SPARQL queries and ground truth answers produced by running the queries. We only use the 50 test questions to evaluate our system since we have already trained our system using the 64 questions on the DBLP+ dataset.

We selected 33 of the 50 test questions (see Table 9.11)[8] that could be answered using only the DBpedia ontology, i.e., not requiring the YAGO ontology types or raw infobox properties whose corresponding DBpedia ontology properties do not exist. Eight of these were slightly modified and their QALD IDs are tagged with a *. DS 3, DS 5, DS 6, DS 11, DS 12, DS 19, DM 2 and DM 3 required modification because they needed operations currently unsupported by our prototype system: aggregation functions (*Which locations have more than two caves?*) and Boolean answers (*Was U.S. President Jackson involved in a war?*). Our changes included removing the unsupported operations or changing the answer type but preserving the relations. For example, the above two questions were changed to *Give me the location of Ape Cave* and *What wars were U.S. President Jackson involved in?*. Although we introduce an auxiliary entity *Ape Cave* for the first question, the entity name does not affect the mapping process since it runs at the schema level and the entity names are not used. In DM 2, we substituted “Richard Nixon” for “Bill Clinton” because the original question cannot be answered using the DBpedia ontology only but an entity name change makes it answerable.

Among the 33 questions, six are DM questions and the rest are DS questions. There

[8] The questions are numbered by the order they appear in the QALD test file, except that we place all DM questions after DS questions

Category & ID	QALD ID	Question
DS 1	43	Which river does the Brooklyn Bridge cross?
DS 2	7	Where did Abraham Lincoln die?
DS 3	10*	What is the wife of President Obama called?
DS 4	50	In which country does the Nile start?
DS 5	44*	Give me the location of Ape Cave.
DS 6	30*	Give me all proteins.
DS 7	49	How tall is Claudia Schiffer?
DS 8	27	What is the revenue of IBM?
DS 9	13	In which country is the Limerick Lake?
DS 10	32	Which television shows were created by Walt Disney?
DS 11	45*	What is the height of the mountain Annapurna?
DS 12	14*	What wars was U.S. President Jackson involved in?
DS 13	40	Who is the author of WikiLeaks?
DS 14	19	What is the currency of the Czech Republic?
DS 15	11	What is the area code of Berlin?
DS 16	16	Who is the owner of Universal Studios?
DS 17	34	Through which countries does the Yenisei river flow?
DS 18	8	When was the Battle of Gettysburg?
DS 19	24*	What mountains are in Germany?
DS 20	26	Give me all soccer clubs in Spain.
DS 21	5	What are the official languages of the Philippines?
DS 22	6	Who is the mayor of New York City?
DS 23	41	Who designed the Brooklyn Bridge?
DS 24	46	What is the highest place of Karakoram?
DS 25	25	Give me the homepage of Forbes.
DS 26	1	Which companies are in the computer software industry?
DS 27	47	What did Bruce Carver die from?
DM 1	3	Give me the official websites of actors of the television show Charmed.
DM 2	37*	Who is the daughter of Richard Nixon married to?
DM 3	35*	What city is Egypt’s largest city and also its capital?
DM 4	29	In which films directed by Garry Marshall was Julia Roberts starring?
DM 5	42	Which bridges are of the same type as the Manhattan Bridge?
DM 6	2	Which telecommunications organizations are located in Belgium?

Table 9.11. 33 test questions on DBpedia

are no IS or IM questions because dealing with indirect relations goes out of the scope of the QALD 2011 workshop[9]. In fact, all of the QALD questions have the following patterns that are customized for ontology-based NLI systems: (i) most contain one relation and no more than two; (ii) single answer type or variable; and (iii) no anaphora used. They pose less challenge to NLP parsers but do not fully explore the advantages of graph query. Since the QALD query set contains no indirect relations, it cannot work as a training dataset for our approach.

Three computer science graduate students who were unfamiliar with DBpedia and its ontology independently translated the 33 test questions into SFQs. We first familiarized the subjects with the SFQ concept and its rules as specified in Section 3 and then trained them with ten questions from the QALD training dataset. We asked them to first identify the entities in a natural language query and their types and then link the entities with the relations given by the query. We also gave them a few simple constraints, e.g., if the entity value is a number, use “Number” as the type of the entity. However, the major force of learning to create the structural queries is by examples. The subjects quickly learned from the ten examples and found the concepts intuitive and easy to understand. The entire learning process took less than half an hour. Finally, we asked each subject to create SFQs for the 33 test questions in which the DM questions are mixed with the DS questions. Because our graphical web interface was under development, the users drew the queries on paper. None of the subjects had difficulty in constructing the SFQs. We then serialized the three versions of the 33 graphical SFQs into text format and formed a test dataset of 99 testcases, which can be referenced in Appendix C.

[9] DM 5 could be thought as a IS question. However, when it was translated into SFQ by three human subjects in our experiment all of them decomposed the relation “the same type as” to two relations linking to the same “Type” entity. Therefore, at least in the SFQ form it is a DM question.

	α	γ	β	θ	MRR
hybrid semantic similarity	0.25	0.4	2.25	1.0	0.813
LSA semantic similarity	0.25	0.6	2.0	1.5	0.720
string similarity	1.0	1.0	2.0	1.0	0.464

Table 9.12. MRR performance on DBpedia for three similarity measures when inferred classes are used

	α	γ	β	θ	MRR
hybrid semantic similarity	0.25	0.4	2.25	1.0	0.782

Table 9.13. MRR performance on DBpedia for hybrid semantic similarity when inferred classes are *not used*

9.4.3 Results using the learned parameters

We learned the optimal parameters of our system using 220 testcases on the DBLP+ dataset and the same parameters, as shown in Table 9.5, are used in testing our system on the 99 DBpedia testcases. The parameters k_1 and k_2 are still set to be 10 and 20, respectively. However the parameter k_3 is changed from 40 to 10 because it allows our system to run faster and response in real time. The value of k_3 realizes the tradeoff between efficiency and effectiveness. According to our DBLP+ experiment of analyzing the impact of k_3 in Section 9.3.7, we find that 40 is a quite large number for k_3 and even 5 can work well.

Table 9.12 shows the MRR performance of the trained system on the 99 DBpedia testcases for three different similarity measures. Inferred classes are enabled in the statistical model used by the system. The hybrid semantic similarity and LSA semantic similarity achieve MRR performance 0.813 and 0.720, which improves upon the string similarity by 75.2% and 55.2% respectively. All the three measures have lowered MRR scores compared with their performance on the DBLP+ dataset.

For the hybrid semantic similarity measure, we carried out an additional experiment – test the trained system in which inferred classes are not used. We employ a different approach to address the problem of the incomplete DBpedia type system. We simply add the

Rank	**hybrid**	**LSA**	**string**	**hybrid w/o inferred classes**
1	75	64	43	72
2	8	10	3	5
3	3	3	2	8
4	2	5	3	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	2
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
> 10	11	17	48	12

Table 9.14. Distribution of ranks for four system scenarios

most general type, *Thing*, with a designated similarity score 0.15 to every class candidate list. The result of this approach is shown in Table 9.13. This simple approach yielded a worse but still good MRR score 0.782.

We can have more insight into the performance by examining the distribution of the ranks over the 99 testcases that the MRR score combines. The distribution is provided in Table 9.14 for four system scenarios – the three similarity measures using inferred classes and the hybrid semantic similarity measure without using inferred classes. The top-10 coverage of hybrid semantic similarity w/ and w/o using inferred classes are 88.9% and 87.9%. The LSA semantic similarity works worse, with a 82.8% top-10 coverage, and the string similarity exhibits the worst performance, with a 51.5% top-10 coverage.

However, both the top-10 coverage and the MRR score have a upper limit 91.9% because there are eight testcases that are not possible to be correctly answered. Five of them are because the human subjects understand two questions differently from our gold standard due to ambiguity in the questions. In DS 16, all of three subjects interpret “Who” as a *Person* type. However, the type that leads to the gold standard answer is *Organization*. In DS 25, two of three subjects interpret the type of “Forbes” to be *Company* but the one leads to the gold standard answer is *Magazine*. The other three testcases are due to an inappropriate property name involved in answering DM 5. In order to answer this question, we need map “type” to *architecturalBureau*, which is not an appropriate name for describing the relation “type of bridge”[10].

The individual reciprocal ranks of the 99 testcases are pictured as bars in Figure 9.37, Figure 9.38, Figure 9.39 and Figure 9.40 for the four system scenarios. The testcases are numbered in the same order as they appear in Appendix C. The bars of the testcases in two categories (DS and DM) are marked with two different colors. Since most of testcases for the hybrid and LSA semantic similarity measures have perfect reciprocal ranks, white slots in the figures indicates the testcases whose correct interpretations failed to be ranked at 1st place. Many of white slots in Figure 9.37 and Figure 9.38 are overlapped, a reflection of the fact that the hybrid similarity is built upon the LSA semantic similarity.

Figure 9.41, Figure 9.42, Figure 9.43 and Figure 9.44 show runtime of each of the 99 testcases for the four system scenarios. In general, the testcases in DM requires more execution time than those in DS because there are more than one relation to deal with in each query in the DM categories. However, there are many DS testcases that took more time to run than many of DM testcases. We can also observe high variance of the runtime within every category. These phenomena demonstrate another important factor influencing the runtime – the connectivity degree among the classes. If a query interpretation requires checking paths between two heavily connected classes, it will take much more time than those who do not.

Table 9.15 shows the average time to run a testcase for the hybrid, LSA and string similarity with inferred classes and the hybrid similarity without inferred classes. The one not using inferred classes takes generally more execution time than the others using

[10] In DBpedia 3.7, the property *architecturalBureau* is renamed to *type*

	average testcase runtime
hybrid semantic similarity and inferred classes	0.648 seconds
LSA semantic similarity and inferred classes	0.660 seconds
string similarity and inferred classes	0.478 seconds
hybrid semantic similarity and *no* inferred classes	1.073 seconds

Table 9.15. Average testcase runtime for four system scenarios

inferred classes. This is because when inferred classes are not used the most general type *Thing* is added to every class candidate list. Consequently, *Thing* is more likely to appear in the concept mapping hypothesis list. Since *Thing* is very intensively connected to some other common classes, it takes much time to check all the schema paths between them. However, we also observe that several testcases in the scenario using inferred classes take longer time to run than the corresponding testcases in the scenarios not using inferred classes. That is because some inferred classes are common classes and heavily connected to other common classes, for example, *L̃ocation*. More than one such inferred classes may appear, possibly along with *Thing*[11], in the concept mapping hypothesis list, resulting in a prolonged processing time.

For those who want to have more complete view on the performance, we provide the top 10 interpretations of all 99 testcases using the hybrid similarity in Appendix C. It is very informative to see what else interpretations are ranked at top places besides the correct one. It helps understand what ambiguity confuses our system and what problems our approach is challenged by.

Discussion

Mapping relation is much more difficult than mapping concepts. Equivalent relations can go beyond synonyms, they can be context-dependent and many of them involve *default*

[11] *Thing* itself can be semantically similar to some classes in the DBpedia ontology. For example, our LSA model tells a similarity score 0.30 between *Thing* and *Place*.

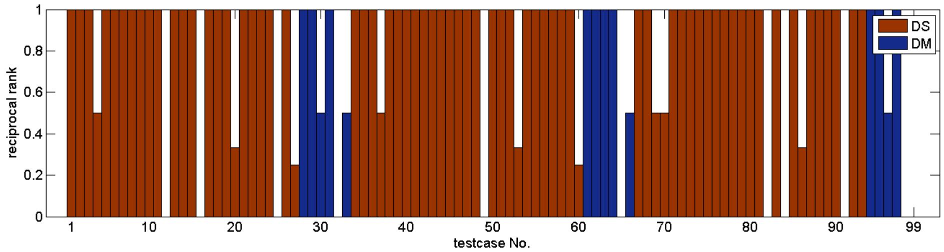


FIG. 9.37. Reciprocal ranks of 99 testcases using the hybrid similarity and inferred classes

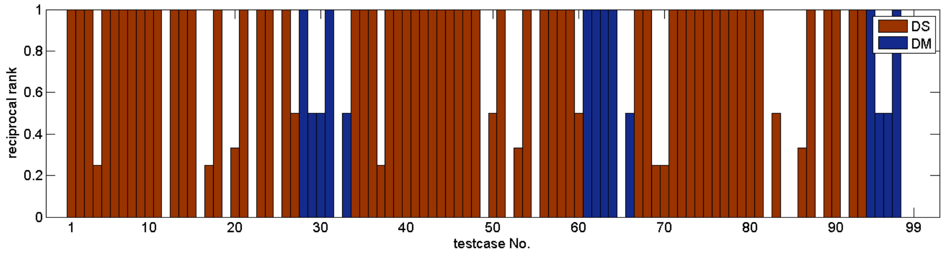


FIG. 9.38. Reciprocal ranks of 99 testcases using the LSA similarity and inferred classes

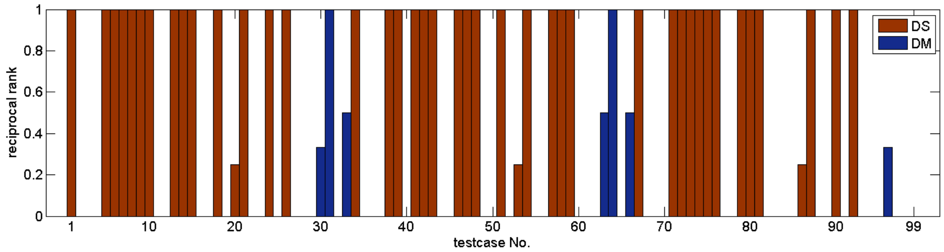


FIG. 9.39. Reciprocal ranks of 99 testcases using the string similarity and inferred classes

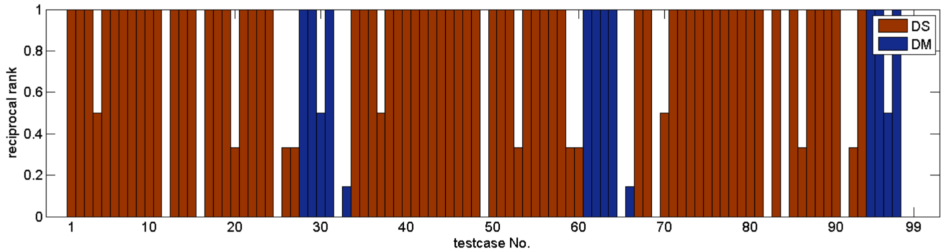


FIG. 9.40. Reciprocal ranks of 99 testcases using the hybrid similarity and *no* inferred classes

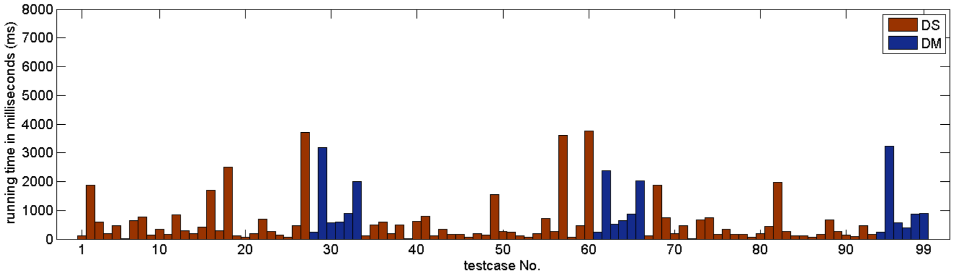


FIG. 9.41. Runtime of 99 testcases using the hybrid similarity and inferred classes

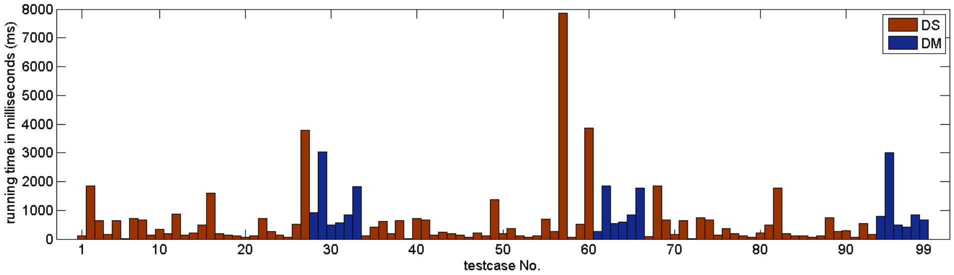


FIG. 9.42. Runtime of 99 testcases using the LSA similarity and inferred classes

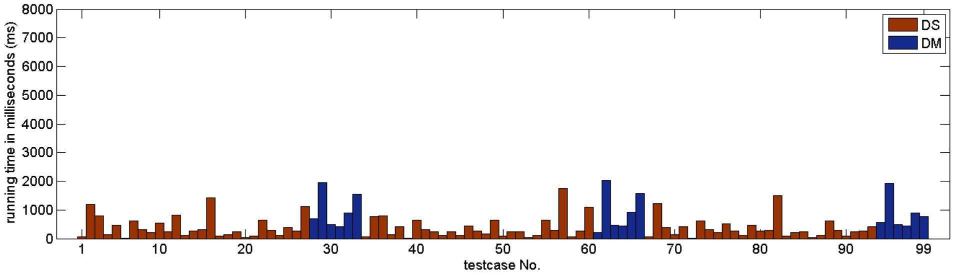


FIG. 9.43. Runtime of 99 testcases using the string similarity and inferred classes

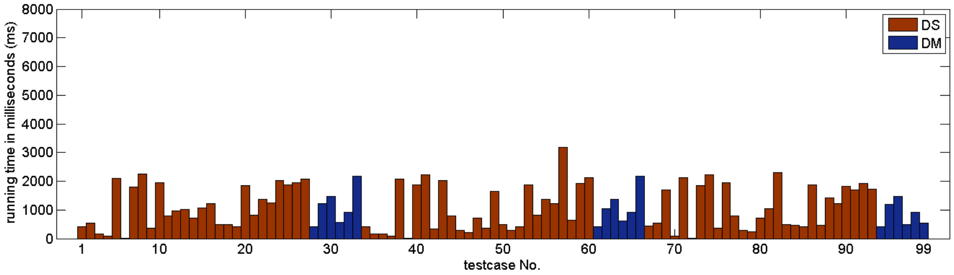


FIG. 9.44. Runtime of 99 testcases using the hybrid similarity and *no* inferred classes

relations. Examples include mapping “actor” to *starring*, “marry” to *spouse*, “died” to *deathPlace*, “mayor” to *leaderName*, “tall” to *height*, “start” to *sourceCountry* and “flows through” to *country*. Thanks to our semantic similarity measure and the context-dependent relation mapping algorithm, we are still able to recognize and map them.

The QALD workshop only provides a single SPARQL query or mapping as the translation for a natural language question. However, our system found some other mappings or interpretations, which also leads to correct answers. One category of these alternative mappings is because of the inverse relation. Consider two examples. When querying about the daughter relation in DM 2, our system is able to map it to not only the *child* property but also the *parent* property in its different interpretations. When querying about the languages of a country in DS 21, our system finds not only the *officialLanguage* property but also the *spokenIn* property. Another category is about the indirect relation. Although the QALD workshop treats all relations in the questions as *direct relation*, it does not prevent our system to interpret them as *indirect relation*. For example, DS 4 asks about the relation which country a river starts in. The QALD interpretation is $River \overset{sourceCountry}{\rightarrow} Country$ but besides this one our system finds an indirect one $River \overset{sourcePlace}{\rightarrow} Place \overset{country}{\rightarrow} Country$, which also answers the query successfully. For another instance, DS 20 asks about the soccer clubs in a country. The QALD interpretation is $SoccerClub \overset{ground}{\rightarrow} Country$ while our system finds an indirect but also correct one $SoccerClub \overset{league}{\rightarrow} SoccerLeague \overset{country}{\rightarrow} Country$. These additional correct interpretations are added to our gold standard.

We observed that our users differed in whether a nominal compound [30] should be entered as a phase or decomposed, leading to another category of structure mismatch other than *indirect relation*. For example, two subjects kept the noun phrase “U.S. President” as a single unit while the other decomposed it into two units *President* and *Country* which are linked by the relation *in*. In the DBpedia ontology, however, there are no links between

U.S. Presidents and the country United States[12]. Therefore, our gold standard judges all the interpretations produced for the decomposed noun phrase version are wrong. DM 6 and DS 12 fall in this category. We will present future work dealing with this kind of structure mismatch in the last chapter.

Our semantic similarity measures sometimes failed due to the flexibility of human expressions. For example, one subject translated DS 18 into a “Battle” entity and a “Year” entity which are connected by the relation “took place”. Our system was misled by “took place” because it is much more similar to the property *place* than to *date*[13].

In the coming sections, we will conduct a series of experiments to analyze how performance is impacted by each individual parameter. Unless specified otherwise, all parameters, except the one being analyzed, have the same values as the ones used in this section.

9.4.4 Performance analysis on β and θ

The values of the parameters α, γ, β and θ are learned from the experiments on the DBLP+ dataset and we directly apply them to the DBpedia dataset. It is of interest to know how good these parameters are for the DBpedia dataset and if any better choices exist. However, the DBpedia query set is not appropriate for tuning α and γ because the queries do not contain indirect relations. Therefore, we only carried out experiments on adjusting β and θ.

We limited our experiments to only two scenarios: hybrid and LSA semantic similarity models with inferred classes enabled. The results of β tuning are shown in Figure 9.45 and Figure 9.46. The range we select for β is $[-5 \mathinner{..} 8]$, with a step of 0.25. The hybrid similarity model finds its highest MRR score 0.826 at β level 1.5 and the LSA similarity model finds

[12] The term “President of United States” appears as the value of a string property of U.S. Presidents, however DBpedia currently does not extract relations from strings

[13] This particular problem can be easily solved by adding the phrase “take place” to our vocabulary list used by the semantic models.

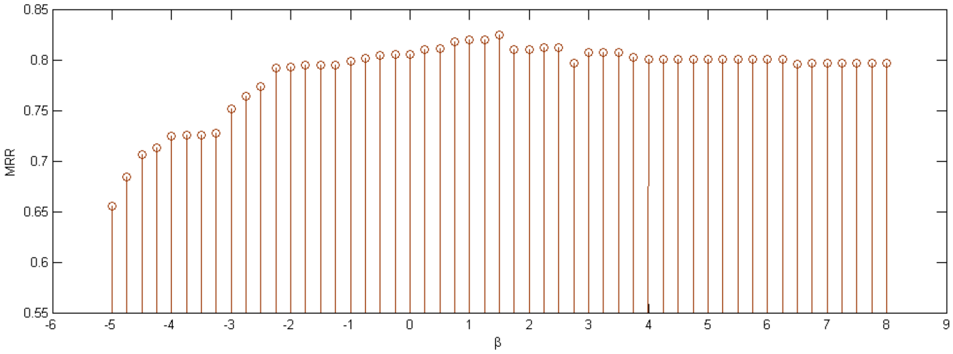


FIG. 9.45. MRR performance versus β, using the hybrid semantic similarity model and inferred classes

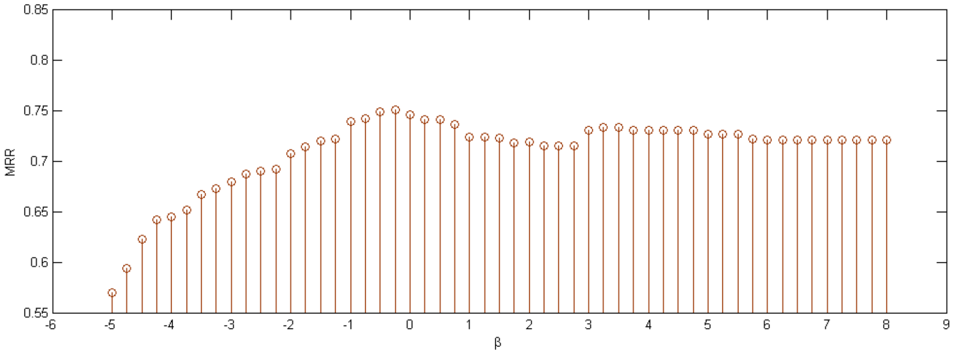


FIG. 9.46. MRR performance versus β, using the LSA semantic similarity model and inferred classes

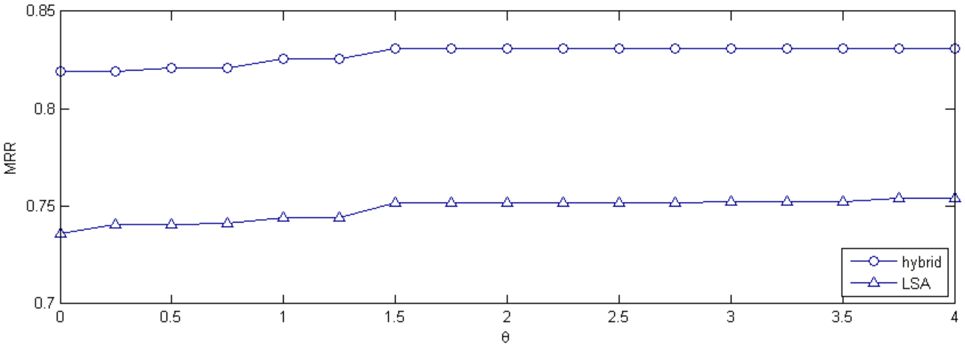


FIG. 9.47. MRR performance versus θ, using the semantic similarity models and inferred classes

its highest MRR score 0.752 at β level -0.25. On the other hand, the β levels we learned on the DBLP+ dataset for the hybrid and LSA similarity models are 2.25 and 2.0, respectively. These levels produce good results on the DBpedia dataset, though not the best.

As we did on the DBLP+ dataset, we also tried some very large β levels but only for the hybrid similarity model. We find that MRR performance goes down to 0.631 when β reaches $1,000$ and keeps that score unchanged as β goes up to $1,000,000$. Compared with the same experiment we did on the DBLP+, we see a greater performance decline on the DBpedia. This shows that *popularity* plays a more important role on the DBpedia than on the DBLP+. This also implies that there are many incorrect schema paths which are “semantically”[14] very competitive to the gold standard paths but can be easily ruled out by their very low popularity.

We further tune θ on the top of the adjusted β levels. The tuning range is $[0 \mathinner{..} 4]$, which is the same as we chose for the DBLP+ experiment. The result is shown in Figure 9.47. The hybrid and LSA similarity models achieve their highest MRR scores 0.831 and 0.754 at θ levels 1.5 and 3.75, respectively.

In sum, after tuning β and θ using the DBpedia’s own query set, we boost MRR performance to 0.831 and 0.754 for the hybrid and LSA semantic similarity models, which are 2.2% and 4.7% improvements over the old scores.

9.4.5 Performance analysis on k_3

We manually set 10 as the value of k_3 in our previous experiments and now we present the results of experimenting with different k_3 values, using our best performing scenario as an example. Figure 9.48 shows the MRR performance is impacted by k_3 using the hybrid similarity model and inferred classes. The range of k_3 is from 1 to 30. The MRR

[14] This is measured by our semantic similarity computation.

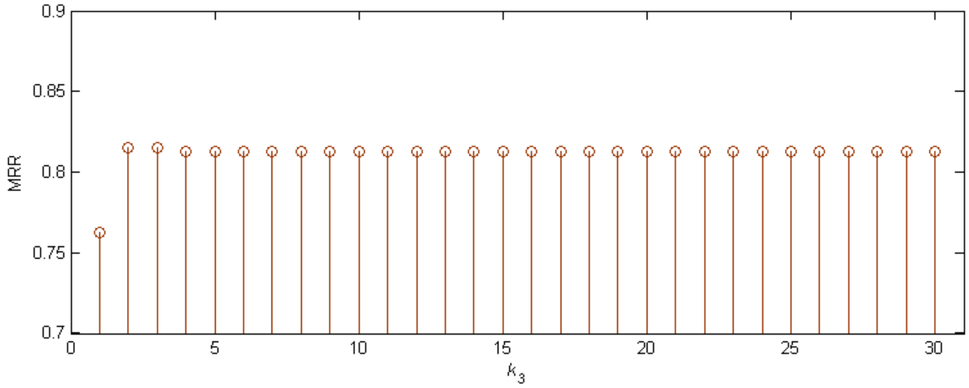


FIG. 9.48. MRR performance versus k_3, using the hybrid similarity and inferred classes

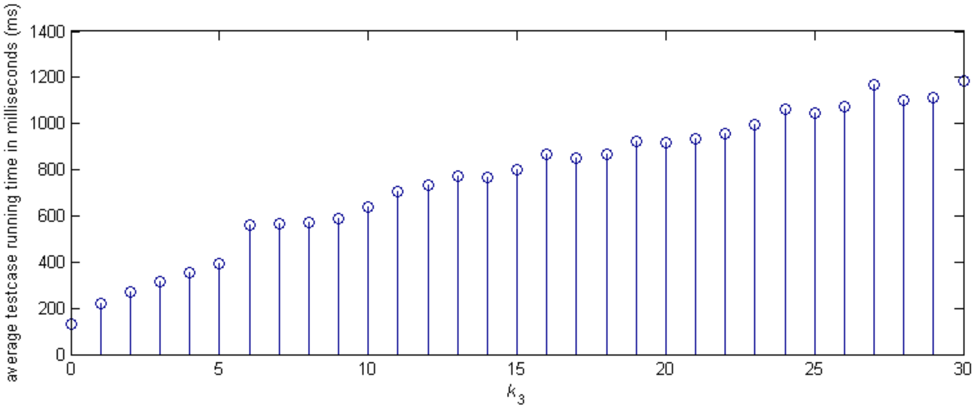


FIG. 9.49. Average testcase runtime versus k_3, using the hybrid similarity and inferred classes

performance quickly reaches its peak, 0.815, when k_3 is only 2 and slightly decreases to 0.813 afterwards. This shows our phase 1 algorithm is very effective to our collection of 99 testcases. The system can run much faster without losing any performance by simply using 2 for the value of k_3.

The average testcase runtime for different k_3 values is presented in Figure 9.49. We can see a close linear relationship is held between the average testcase runtime and k_3 values. The intercept at $k_3 = 0$ represents the time, 0.132 seconds, that the phase 1 algorithm takes. Majority of time is consumed by the phase 2 algorithm unless k_3 is a very small

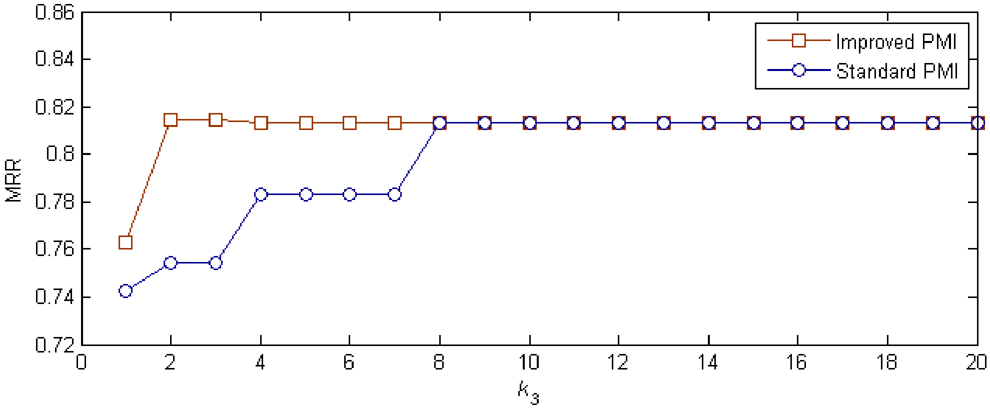


FIG. 9.50. Comparing improved PMI with standard PMI on the DBpedia dataset

number. The figure illustrates that an effective phase 1 algorithm is crucial to make the system efficient.

9.4.6 Compare improved PMI with standard PMI

In the phase 1 algorithm, we use the improved PMI to measure association degree between schema terms. In order to demonstrate that the improved PMI produces a more effective phase 1 algorithm than the standard one, we carry out the same experiment as that in Figure 9.48 but substitute the standard PMI for the improved PMI. The comparison result is shown in Figure 9.50. The improved PMI has a better performance than the standard PMI until $k_3 = 8$ and their performance is tied thereafter. The improved PMI climbs much faster to its ceiling than the standard PMI. This shows that improved PMI produces a more effective phase 1 algorithm than the standard one. This also provides an evidence that the improved PMI is a better measure for computing association degree than the standard PMI.

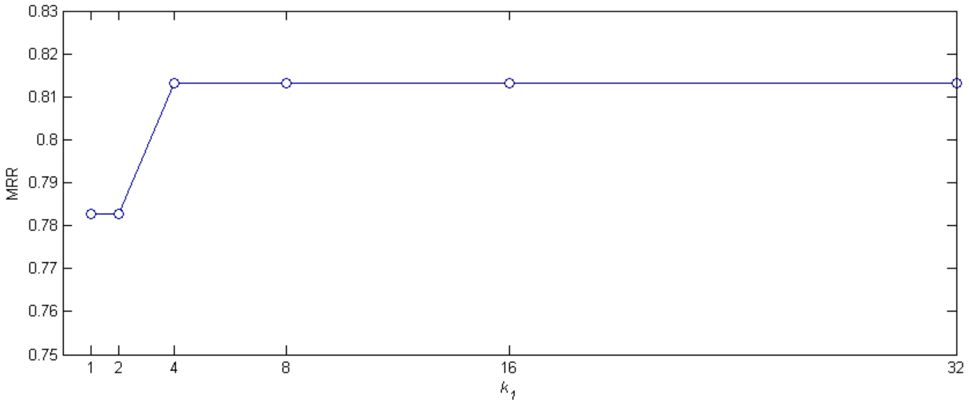


FIG. 9.51. MRR performance versus k_1, using the hybrid similarity and inferred classes

9.4.7 Performance analysis on k_1

The parameter k_1 has a fixed value 10 in all the previous experiments. In this section, we experiment on varying k_1 and see the impact on both efficacy and efficiency. These experiments use the hybrid similarity model and inferred classes. Figure 9.51 shows how the MRR performance changes over a number of k_1 values from 1 to 32. We do not use a larger right boundary, such as 64, because that would make the class candidate lists mostly under-occupied. The MRR performance already reaches its highest when k_1 is only 4. This demonstrates our hybrid semantic similarity measure is effective in finding appropriate class candidates for the concepts in our collection of 99 SFQ testcases.

The average testcase runtime against a number of different k_1 values is presented in Figure 9.52. When k_1 is between 1 to 4, the number of the concept mapping candidates actually produced, represented by k_3', is rather determined by k_1 than k_3. k_3' is positively correlated with k_1 and is often less than 10, the value we set for k_3. In this period of k_1, both the time used by the phase 1 and phase 2 algorithms grows with k_1. When k_1 is between 4 to 16, k_3' is constantly 10. In this period of k_1, the phase 2 algorithm takes almost the same amount of time to run and the growth in runtime can be mostly ascribed to the phase

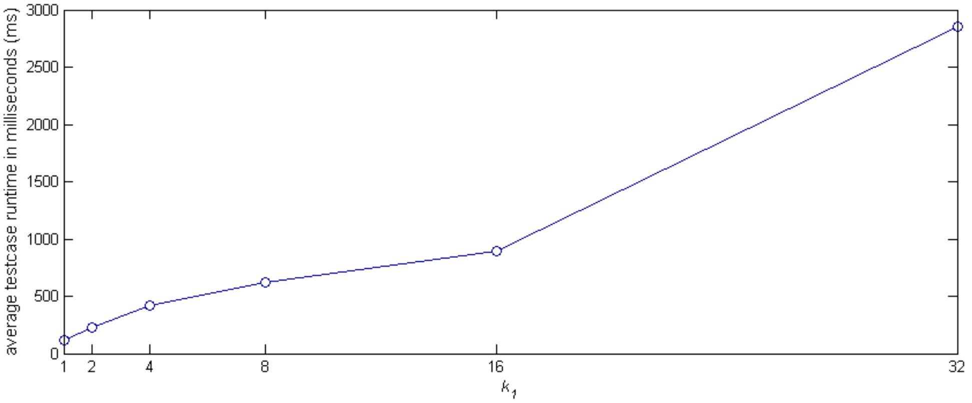


FIG. 9.52. Average testcase runtime versus k_1, using the hybrid similarity and inferred classes

1 algorithm. The growth tends to be slow because majority of time is still consumed by the phase 2 algorithm. When k_1 is between 16 to 32, the time used by the phase 2 algorithm remains the same but the time used by the phase 1 algorithm increases rapidly and takes the lead.

In Section 5.2.2, we theoretically show that the concept mapping optimization algorithm has a computation complexity of $k_1^n\left[O(\frac{2m}{n}k_2)+O(\log k_3)\right]$. Here, we will experimentally examine it. For doing this, we first define $t(k_1)$, a function of k_1, be the time used by the optimization algorithm. Then, we can deduce $log(t)$ is a linear function of $log(k_1)$ from the Equation 9.4 where n, the number of nodes in the query, can be solved as the slope of the linear function.

$$\begin{aligned}
log(t(k_1)) &= log\left(k_1^n\left[O(\frac{2m}{n}k_2)+O(\log k_3)\right]\right)\\
&= n\cdot log(k_1)+log\left(O(\frac{2m}{n}k_2)+O(\log k_3)\right)\\
&= n\cdot log(k_1)+C \qquad (9.4)
\end{aligned}$$

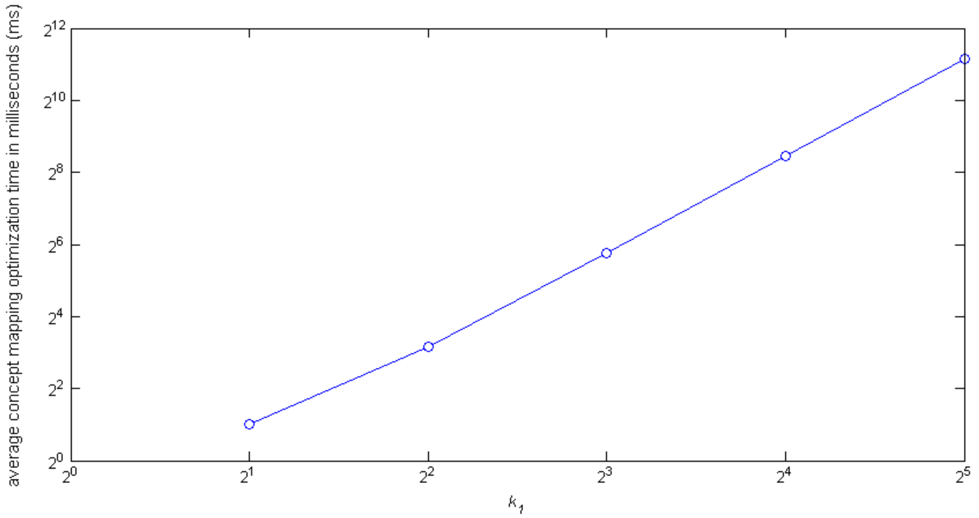


FIG. 9.53. Average concept mapping optimization time versus k_1, using the hybrid similarity and inferred classes

The actual relationship between average concept mapping optimization time and k_1, both in logarithmic scale, is shown in Figure 9.53. We exclude the case when k_1 is 1 because the optimization took less than 1 milliseconds, which cannot be measured. The relationship in the figure is very close to a straight line. The slope of the line, 2.5, fits fairly well with our collection of 99 testcases, which contains mostly 2-node queries and some 3-node queries.

9.4.8 Performance analysis on k_2

The parameter k_2 has a pre-selected value 20 in all the previous experiments. In this section, we experiment on varying k_2 and see the impact on both efficacy and efficiency. These experiments use the hybrid similarity model and inferred classes. Figure 9.54 shows the MRR performance for a number of k_2 values from 2 to 32. We make test value of k_2 start from 2 but not 1 because *default relations* need at least two candidates. The MRR performance always stays at the top level regardless of the selecting of k_2. This shows we can use a small number for k_2 on this DBpedia experiment where k_1 is set to 10 and k_3 is set to 10. However, if we make k_3 very small, the value of k_2 will impact on the performance.

Figure 9.55 shows how the value of k_2 affects the average concept mapping optimization time. A clear linear relationship exists between the mapping optimization time and k_2, which is in accordance with the time complexity $k_1^n\left[O(\frac{2m}{n}k_2)+O(\log k_3)\right]$ that we theoretically calculate in Section 5.2.2.

Figure 9.56 shows the average testcase runtime versus a number of different k_2 values. By comparing Figure 9.56 with Figure 9.55, we can find that the testcase runtime and the mapping optimization time share a very similar way to grow with k_2. This implies that k_2 has little effect on the runtime of the phase 2 algorithm under the current settings and the increase of the average testcase runtime is mainly caused by the mapping optimization step in the phase 1 algorithm.

9.4.9 Formal query generating and entity matching

In this section, we will show how to generate formal queries, namely SPARQL, from the top interpretations of a SFQ query and run the SPARQL queries to get answers. We also present results of evaluating the answers produced for the 99 testcases, in terms of precision and recall, against our gold standard.

Translating a SFQ interpretation to a SPARQL query is largely straightforward because they have exactly the same structure and use exactly the same ontology terms. However, we still need to deal with several issues before we can carry out the translation, which are listed as below.

1. **Matching entity name.** Ideally, we assume that a big index exists that maps all the names of an entity to the id or URI of the entity. In DBpedia, since every entity has the *label* property that refers to its name, we can build such an index on the *label*

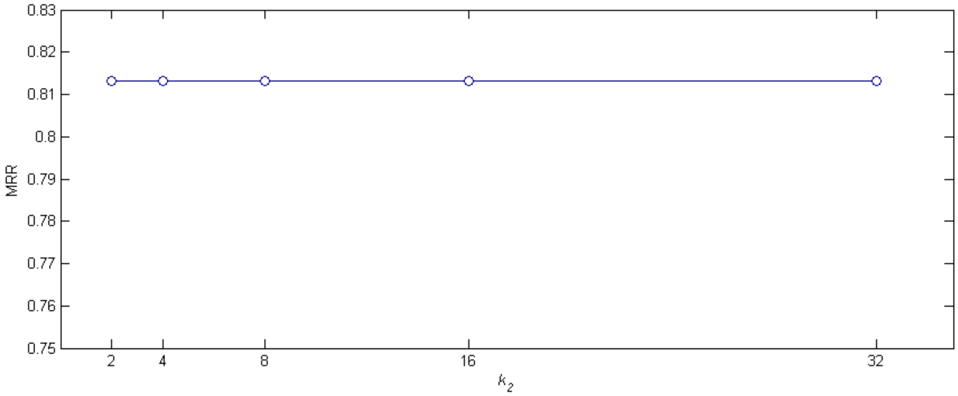


FIG. 9.54. MRR performance versus k_2, using the hybrid similarity and inferred classes

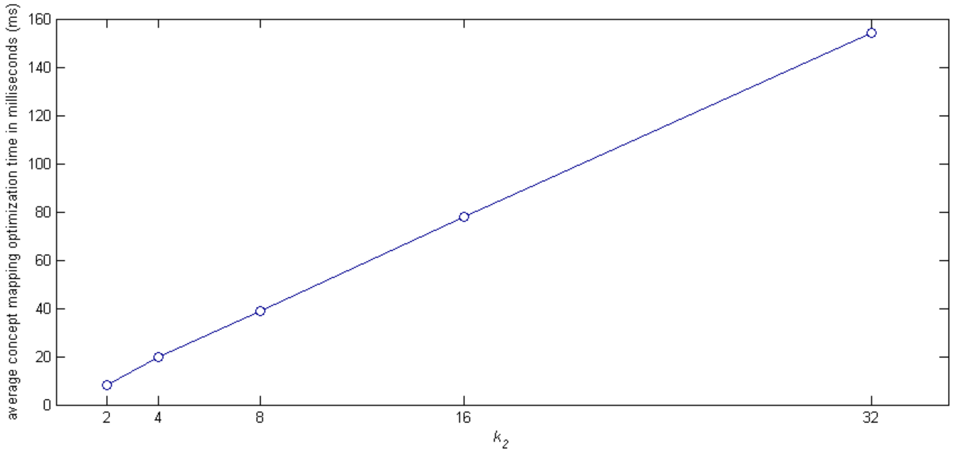


FIG. 9.55. Average concept mapping optimization time versus k_2, using the hybrid similarity and inferred classes

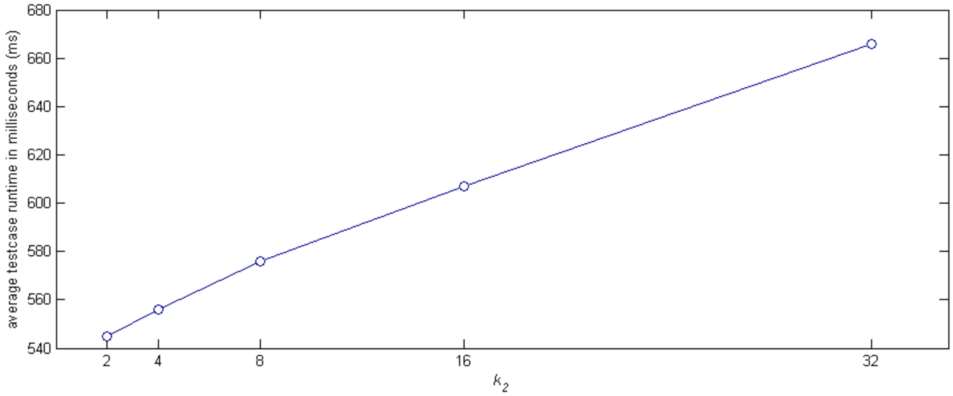


FIG. 9.56. Average testcase runtime versus k_2, using the hybrid similarity and inferred classes

property. However, this index is incomplete because not all the name variations are included. Although names work as identifiers for entities, certain variations do exist. For example, both “Barack Obama” and “Obama” are the common ways to refer to the same person. We use a three-step pragmatic approach to handle this problem, which is described as follows.

At the first step, we produce a SPARQL query by doing an exact match on the entity name and run the query. If the query returns a non-empty result, we present the result as the answer and halt the program. By doing this, we assume that the exact match on an ambiguous name will return the most popular entity under that name. The assumption is well satisfied on the DBpedia dataset. For example, if we ask “*Who is the wife of Michael Jordan?*”, the exact match on “Michael Jordan” will lead to the American basketball player, which is often the expected answer under this general context. If the query returns an empty result, this entails either the exact match failed to match any entity or the most popular entity does not meet the conditions in the query. In both cases, we proceed to the next step. For example, consider the query “*Who is the thesis advisor of Michael Jordan?*”. The SPARQL query using the exact name match will return an empty result because the basketball player does not have a thesis advisor.

At the second step, we deal with a very common category of name variation. That is, whether appending the type of the entity in its name. Consider examples like “New York City” and “Nile River”. “New York” is also a common way to refer to the city and so is “Nile” to the river. This problem is especially prevalent in our circumstance because our SFQ interface asks the user to input both name and type of an entity. We address this name variation by first comparing the entity name to the type of the entity, which is known to us in our circumstance. If the entity name ends with the type, we remove the type to create a variation of the name; otherwise we append the type. Then, we produce a SPARQL query using an exact match on this new entity name. If a non-empty result is returned, we output it as answers and halt the program; otherwise we proceed to the third step.

At the third step, we no longer do exact match on the entity name. Instead, we apply full-text search or keyword search to matching entity name. Virtuoso [29], the SPARQL query engine we used, supports a built-in text search function *bif:contains* that finds literals containing specified text. Full-text search greatly improves the recall of locating the target entity but lowers the precision. For example, both “Barack Obama” and “Michelle Obama” are the results for matching “Obama”.

2. **Missing types.** We need to address a deficiency in DBpedia: many instances do not have all of the appropriate type assertions. *Bill Clinton* is not asserted to be of type *President, Beijing* is not of type *City* and many person instances even lack the *Person* type. The way we deal with this problem is that, each time we remove the most general type[15] constraint from the SPARQL query and run the modified query again. This continues until an non-empty result is returned or there is no type constraint left in the query. The more general a type is, the less information it contains. Dropping the most general type maximizes the information remaining in the query in each iteration.

3. **Missing relations.** Many common relations are missing in the DBpedia dataset. For example, there is no any relation linking the President Obama and the United States. For a multiple-relation query which returns an empty result, we can remove the relations that are not critical to the query in order to improve recall. Consider the SFQ example $\frac{?}{Place} \xleftarrow{born\,in} \frac{Obama}{President} \xrightarrow{of} \frac{United\,States}{Country}$. The relation $\frac{Obama}{President} \xrightarrow{of} \frac{United\,States}{Country}$

[15] How general a type is is measured by the number of instances it has.

can be dropped without affecting the meaning of the query very much. This relation exemplifies the pattern we use to identify non-critical relations – none of the nodes in the relation are variables and one of the nodes is a leaf node.

We use a non-empty strategy to automatically generate answers for a SFQ query. We start from the best interpretation of a SFQ query. We issue a sequence of SPARQL queries, which are resulted by the modules that deal with three-step entity matching, missing relations and missing types in order. If any of the SPARQL queries in the sequence produces a non-empty result, we present it as answers and quit the program. If still no result is found, we accept for the second best interpretation and so on.

We installed a SPARQL endpoint, which uses Virtuoso Opensource 6.1 query engine, in one of our lab servers. The machine is equipped with Intel Xeon 1.80GHz CPU and 64 GB memory. However, multiple tasks from different users were sharing the computing resource when we ran our experiments on it. We loaded the Virtuoso engine with not only the DBpedia ontology but also the YAGO ontology and raw infobox data to reproduce the QALD SPARQL endpoint. The SPARQL queries are generated in our testing machine, then sent to the remote Virtuoso server via HTTP calls for execution and get answers back.

Table 9.16 shows the evaluation results on the 99 testcases for four system scenarios. The parameters of the four scenarios are the same as in Table 9.12 and Table 9.13, which are learned from the experiments on the DBLP+ dataset. Both the hybrid and LSA semantic similarity measures achieve good precision and recall, which are much better than what the string similarity obtains. Using inferred classes helps improve the performance for the hybrid semantic similarity measure, but not very much. The average testcase runtime for the semantic similarity measures, including both query interpretation and execution time, was all around one second, which shows our system can response in real time. The string similarity measure took much longer response time because it issued more SPARQL

system scenario	mean precision	mean recall	average runtime	average issued SPARQL queries
hybrid sim. and inferred classes	0.909	0.889	0.721 sec.	2.8
LSA sim. and inferred classes	0.814	0.801	0.766 sec.	3.6
string sim. and inferred classes	0.549	0.545	16.866 sec.	13.5
hybrid sim. and *no* inferred classes	0.889	0.869	1.138 sec.	3.9

Table 9.16. SPARQL evaluation results on 99 testcases for four system scenarios

queries per testcase. By comparing runtime in Table 9.16 with that in Table 9.15 which only includes query interpretation time, we can find it took trivial time to execute the SPARQL queries on the Virtuoso server, except for the case of string similarity.

Discussion

The problems arising in query interpretation have been discussed in Section 9.4.3. Here we will discuss the problems occurring in SPARQL generating and query execution through one example. Consider the SFQ $\frac{Obama}{President} \overset{wife}{\rightarrow} \frac{?}{Person}$. Because the entity Barack Obama lacks the type *President* in DBpedia, the produced SPARQL queries return empty results until the type *President* is dropped. Since “wife” is mapped to *spouse* in the top interpretation of the SFQ, the query is reduced to “Who is the spouse of Obama?”. When the keyword search is used to match “Obama”, several instances are matched, of which only “Barack Obama” and “Michelle Obama” have the *spouse* property. Therefore, our program returns both “Michelle Obama” and “Barack Obama” as answers, which have a precision 0.5. An interesting question is, whether we could make an assumption that an entity name in a SFQ query always refers to a single entity. If yes, we can further choose from the answers the ones that correspond to the most popular entity[16] matched by the name. In this way, our system can precisely answer the above question. However, this assumption may hold most of times but not always. For example, in the SFQ query

[16] The popularity of an entity can be measured by its incoming links in DBpedia.

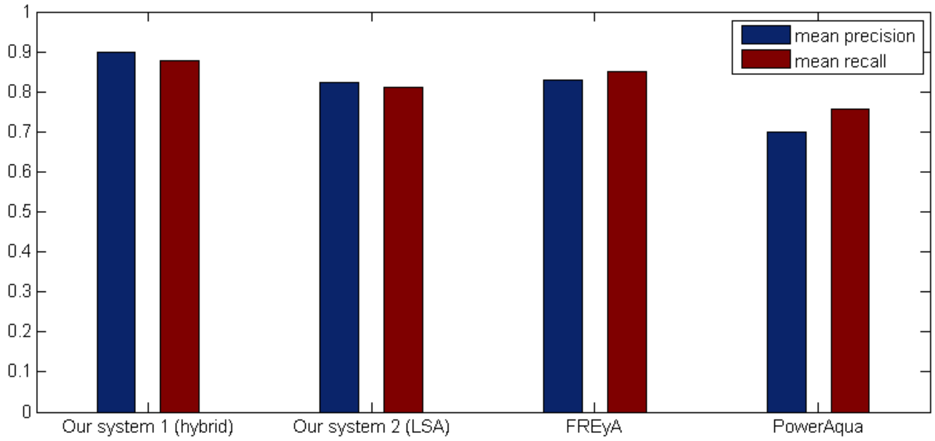


FIG. 9.57. Comparing with two QALD systems on 30 test questions

$\frac{Harry\,Potter}{Book} \overset{published\,in}{\longrightarrow} \frac{?}{Year}$, the name “Harry Potter” is probably used to match a series of books by the user.

9.4.10 Comparison with existing systems

The QALD 2011 report [82] showed results of two systems, FREyA and PowerAqua, on the 50 test questions. Both systems modified or reformulated some of the questions that their NLP parsers have difficulties in understanding. We compared our system with them using 30 questions (90 testcases) in Table 9.11. Three questions, DS 5, DS 11 and DS 19, were excluded because we had simplified them by removing aggregation operations. Among the 30 questions, FREyA modified four question (DS 4, DS 26, DM 2 and DM 6) and PowerAqua eight (DS 3, DS 4, DS 12, DS 17, DS 18, DS 23, DS 24 and DS 26).

Figure 9.57 shows mean precision and recall of FREyA, PowerAqua and our system in two settings over the 30 questions. The two settings are the first two scenarios in Table 9.16, one using the hybrid semantic similarity and the other the LSA semantic similarity. The best performing system is our system using the hybrid semantic similarity. Our

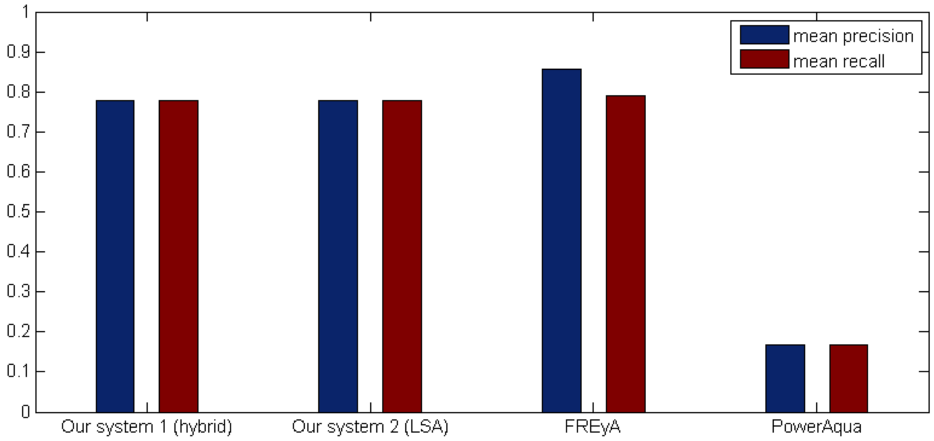


FIG. 9.58. Comparing with two QALD systems on 6 two-relation questions

LSA system performs a little bit worse than FREyA but significantly better than PowerAqua. We also present their performance on the six questions consisting of two relations in Figure 9.58 where FREyA shows the best performance. Our systems yield a performance close to FREyA but PowerAqua exhibits a much worse performance. FREyA has an overall performance close to our systems but it is an interactive system incorporating dialogs to disambiguate questions [24]. FREyA needs the user to manually specify the mapping between a user term and its corresponding ontology term by searching into the ontology when the user term is outside the system's linguistic coverage. Both FREyA and PowerAqua used the 50 DBpedia training questions to tune their systems. In contrast, we do not use the training questions at all and our systems are learned from the DBLP+ data, which is in a totally different domain from DBpedia. It is also worth mentioning that our LSA system is built on a purely computational and statistical approach. It does not use WordNet or any other human-crafted knowledge and works almost as well as the other systems.

PowerAqua's performance dropped dramatically on the six two-relation questions while FREyA and our system remained the same. There are two reasons why our system

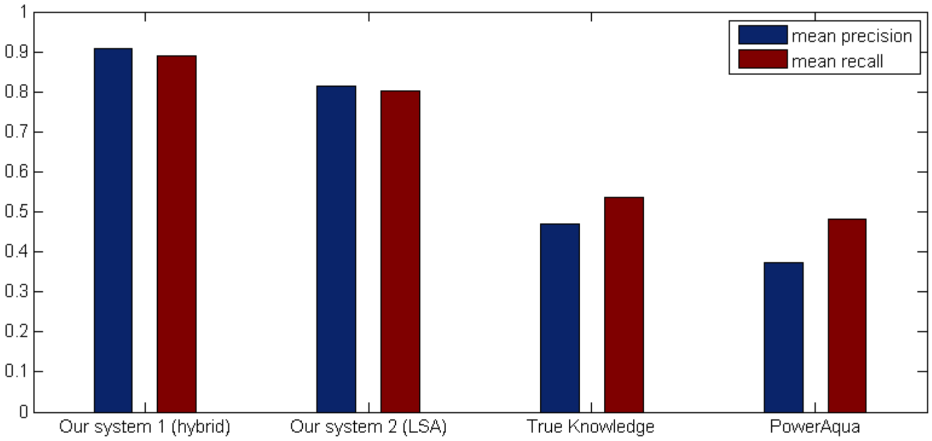


FIG. 9.59. Comparing with two online systems on 33 test questions

yields almost he same performance on six two-relation queries as on other single-relation queries. First, we relied on humans to create the relational structure of the queries but PowerAqua uses NLP techniques. Second, two-relation queries give more information and therefore have less ambiguity than single relation queries.

We also evaluated all 33 test questions on two online systems, PowerAqua [81] and True Knowledge [101] in August 2011. Both include DBpedia as part of their knowledge bases. The true answers of most of the test questions are complete but some are not, which means that PowerAqua and True Knowledge can return correct answers that are not in the true answers of some questions. For these cases, we manually checked the results to identify all correct answers in computing precision. PowerAqua shows the dataset used to derive answers, allowing us to use answers only from DBpedia and ignored others. The results are presented in Figure 9.59 and Figure 9.60.

Ontology-based open domain QA is a new research area and the QALD workshop is the first known to us to provide an evaluation dataset. A direct comparison of our system against others is difficult due to different settings. Systems in the comparisons used slightly

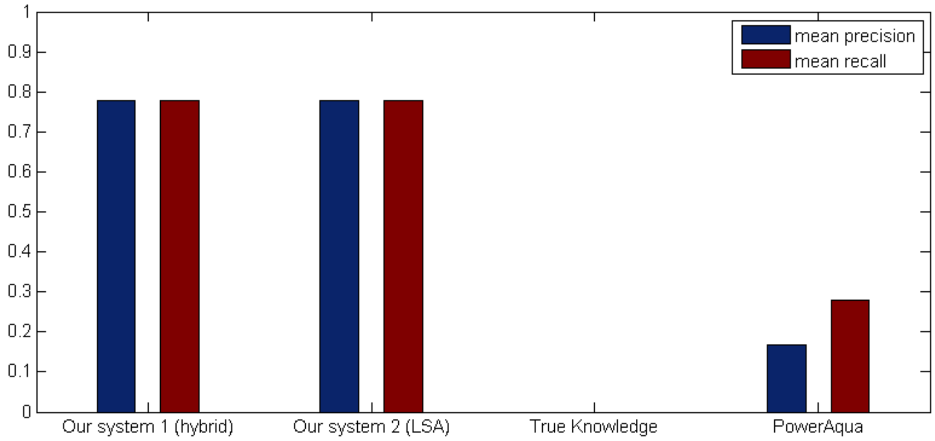


FIG. 9.60. Comparing with two online systems on 6 two-relation questions

different query sets and ran on datasets not completely the same. The two online systems have not been tuned using QALD training questions. Moreover, our user interface differs from others. Some people may think either NLI or SFQ interface is just a means to allowing users to describe their information needs and we can directly compare their results. Others may believe the comparison is biased because our system benefits from user interpretation of NL questions.

Nevertheless, the comparisons with top systems show our approach works well. Our system also has three desirable features that others lack. First, our approach saves expensive human effort in crafting schema of data and the mapping lexicon. True Knowledge, FREyA and PowerAqua all depend on such knowledge in performing disambiguation and addressing vocabulary mismatch problem that cannot be solved by synonym expansion [102, 24, 67]. In contrast, our systems learned such kind of knowledge statistically and automatically from data itself. Second, our system has the advantage over automatic NLI systems in answering questions containing two or more relations. It can even handle more complicated queries, such as the one in Figures 5.1, while the corresponding NL question would involve multiple answer types and anaphora. Third, our system is fast. Both of our two systems have an average response time less than one second. FREyA reported 36 seconds on average in answering a question [24]. PowerAqua did not report execution time on QALD questions but our experiment of testing 33 questions on its website showed an average of 143.7 seconds.

Chapter 10

CONCLUSION AND FUTURE WORK

Large collections of structured semantic data like DBpedia provide essential knowledge for many applications and potentially for end users, but are difficult for non-experts to query and explore. The schema-free structured query approach allows people to query semantic data without mastering formal query languages or acquiring detailed knowledge of the classes, properties and individuals in the underlying ontologies and the URIs/IDs that denote them. Our system uses statistical data about lexical semantics and the target RDF datasets to generate plausible SPARQL queries from a user's intuitive query. We obtained promising results, precision of 0.909 and recall of 0.889, in an evaluation on DBpedia with users who sought answers for 33 QALD test questions.

The core contribution of this thesis is to provide a fully-automatic approach to interpret and disambiguate user queries, which means mapping the concepts, relations and entities in a user query to the concepts, paths and entities in a given knowledge base. Our approach does not require any human-crafted knowledge, such as thesauri, lexicons and mapping rules. This distinguishes it from all existing approaches. Our system that was purely built on data statistics obtained precision 0.814 and recall 0.801 on the 33 QALD test questions. The performance is worse than our best system that exploits WordNet knowledge but not much and it is still comparable to other existing systems. Our approach is also very efficient. It took less than one second for answering questions on the DBpedia dataset by average. Our experiments also show that it can be even faster while maintaining the same accuracy if other parameters are used. These nice properties make us believe that our approach can be scaled to even larger and more diverse datasets such as Freebase.

Although we have already seen many applications of lexical semantic similarity in NLP, AI and information retrieval, it is almost always incorporated and stay as a shallow feature. In this thesis, we use lexical semantic similarity as units for building more complicated structural semantic similarity. We show the disambiguation problems that arise in this process as well as algorithms that we use to deal with the problems. The good performance of our system implies that it is possible to have compositional semantic similarity built on the top of lexical semantic similarity that takes into account the structural features of two graphs.

In this thesis we propose some new and useful concepts, including schema network, schema path, schema path probability and semantic stretch. We also develop novel metrics and algorithms to work with these concepts. We believe these concepts have general natures and can be applied to other problems different than building schema-free query interfaces.

Our current query interpretation mechanism has difficulties in understanding mathematical operations, such as *larger than* and *not equal*. However, since the number of these operations is finite and small, we can implement them as a toolbar in the graphical interface for drawing SFQs. Users can select an operation by clicking an icon in the toolbar and apply it to the nodes in the SFQ query.

A key challenge in our ongoing and future work is to move beyond DBpedia and make it easier to apply our SFQ approach to new RDF data collection and to a large LOD cloud. In addition, we are currently working on a few more modest extensions. The first extension makes entering terms for concepts optional. Consider the SFQ in Figure 5.1, where the user might omit the concept name for the named entity “Woody Allen”. Our solution is to find all possible types of entities lexically matching “Woody Allen”, put the classes into the candidate list of *Woody Allen* and run the same algorithm to identify the right class. The second incorporates user interaction to create SFQs. Information in our CAK models can help users explore the application domain and pick appropriate concepts and relations from dynamically pop-up windows. The third extension is to further improve the accuracy of our lexical semantic similarity model, particularly in dealing with words of many senses.

Appendix A

CAK OF DBLP+

This appendix supplies the Concept Association Knowledge (CAK) of the DBLP+ dataset that is automatically computed from data, using the approach discussed in Section 4.4.

A.1 Association knowledge between classes and properties

Below is a list of classes with their associated properties. The association degree, computed using PMI, is also presented.

- ←**Article**: journal 5.8, @volumeNumber 5.8, @issueNumber 5.8, author 5.1, primaryAuthor 5.1, @DOI 5.1, venue 5.0, firstAuthor 5.0, @publicationYear 5.0, cites 5.0, @pageNumbers 5.0, secondAuthor 4.9, @subject 4.8, @abstract 4.7, @numberOfCitations 4.6, @name 4.6, institution 3.9, editor -3.1, secondEditor -3.1, firstEditor -3.3
- →**Article**: cites 7.0
- ←**Author**: @numberOfPublications 7.0, @e-mail 6.6, @homepage 6.4, @numberOfCitations 6.1, @name 6.1, institution 5.5, @subject 5.2
- →**Author**: author 6.2, primaryAuthor 6.1, firstAuthor 6.0, secondAuthor 6.0, editor 5.4, firstEditor 5.2, secondEditor 5.2
- ←**Book**: @ISBN 8.4, publisher 8.4, series 7.9, editor 6.0, firstEditor 5.9, secondEditor 5.8, @publicationYear 4.8, firstAuthor 4.7, @pageNumbers 4.6, primaryAuthor 4.5, @numberOfCitations 4.4,

@name 4.4, author 4.3, secondAuthor 4.0, cites 3.9, @volumeNumber 3.7, institution 0.1

- →**Book**: book 9.3, cites 6.4, proceedings 0.2
- ←**Conference**: @numberOfPublications 6.0, @subject 5.3, @numberOfCitations 5.2, @name 5.2
- →**Conference**: conference 7.5, venue 7.0
- ←**Country**: @numberOfPublications 5.1, @numberOfCitations 4.2, @name 4.2
- →**Country**: country 10.9
- ←**Editor**: @homepage 8.1, @e-mail 7.4, institution 6.4, @numberOfPublications 6.0, @subject 5.4, @numberOfCitations 5.1, @name 5.1
- →**Editor**: editor 6.9, firstEditor 6.7, secondEditor 6.6, author 6.0, secondAuthor 5.9, primaryAuthor 5.9, firstAuthor 5.6
- ←**InBook**: book 8.7, @DOI 4.9, @publicationYear 4.9, @pageNumbers 4.9, cites 4.6, firstAuthor 4.6, primaryAuthor 4.5, @subject 4.5, @numberOfCitations 4.5, @name 4.5, author 4.4, secondAuthor 4.2, @abstract 3.1, institution 2.7
- →**InBook**: cites 6.3
- ←**InProceedings**: proceedings 5.5, conference 5.5, cites 5.2, author 5.2, primaryAuthor 5.1, secondAuthor 5.1, venue 5.1, @pageNumbers 5.1, firstAuthor 5.0, @abstract 5.0, @publicationYear 5.0, @DOI 5.0, @subject 4.9, institution 4.8, @numberOfCitations 4.6, @name 4.6, secondEditor -3.4, editor -3.4, firstEditor -3.5
- →**InProceedings**: cites 7.0, proceedings -1.1
- ←**Institution**: country 9.6, @numberOfPublications 6.3, @numberOfCitations 5.4, @name 5.4
- →**Institution**: institution 8.7
- ←**Journal**: @numberOfPublications 5.6, @subject 5.1, @numberOfCitations 4.8, @name 4.8
- →**Journal**: journal 7.7, venue 7.0
- ←**Paper**: author 5.2, cites 5.2, primaryAuthor 5.2, venue 5.1, firstAuthor 5.1, @publicationYear 5.1, @pageNumbers 5.1, secondAuthor 5.1, @DOI 5.1, proceedings 5.0, conference 5.0, journal 5.0, @volumeNumber 5.0, @issueNumber 5.0, @abstract 5.0, @subject 5.0, @numberOfCitations 4.7, @name 4.7, institution 4.6, editor -3.2, secondEditor -3.2, firstEditor -3.3

- →**Paper**: cites 7.1, proceedings -1.6
- ←**Person**: @numberOfPublications 7.0, @e-mail 6.6, @homepage 6.4, @numberOfCitations 6.1, @name 6.1, institution 5.5, @subject 5.2
- →**Person**: author 6.2, primaryAuthor 6.1, firstAuthor 6.0, secondAuthor 6.0, editor 5.4, firstEditor 5.2, secondEditor 5.2
- ←**Proceedings**: editor 8.9, firstEditor 8.7, secondEditor 8.7, publisher 8.5, series 8.4, @ISBN 8.4, conference 5.4, @numberOfPublications 5.3, @volumeNumber 4.9, @publicationYear 4.9, @numberOfCitations 4.5, @name 4.5, cites 0.0
- →**Proceedings**: proceedings 8.1, cites 0.4
- ←**Publication**: author 5.2, cites 5.2, primaryAuthor 5.2, @publicationYear 5.1, firstAuthor 5.1, venue 5.1, @pageNumbers 5.1, secondAuthor 5.1, @DOI 5.1, conference 5.0, proceedings 5.0, @volumeNumber 5.0, journal 5.0, @issueNumber 5.0, @abstract 5.0, @subject 4.9, @numberOfCitations 4.7, @name 4.7, institution 4.6, editor 4.5, publisher 4.4, @ISBN 4.4, book 4.3, firstEditor 4.3, secondEditor 4.2, series 4.2, @numberOfPublications 0.9
- →**Publication**: cites 6.9, proceedings 6.7, book 6.0
- ←**Publisher**: @name 5.6
- →**Publisher**: publisher 10.5
- ←**Series**: @name 5.4
- →**Series**: series 11.0
- ←**Thesis**: institution 6.2, firstAuthor 4.9, @publicationYear 4.9, @pageNumbers 4.7, @numberOfCitations 4.5, @name 4.5, @subject 4.4, primaryAuthor 4.4, author 4.0, cites 3.4, @abstract 2.5, secondAuthor -3.0
- →**Thesis**: cites 5.7
- ←**Thing**: author 5.1, cites 5.1, primaryAuthor 5.1, @name 5.1, @numberOfCitations 5.1, @publicationYear 5.0, firstAuthor 5.0, @subject 5.0, venue 5.0, @pageNumbers 5.0, secondAuthor 5.0, @DOI 5.0, @numberOfPublications 4.9, conference 4.9, proceedings 4.9, @volumeNumber 4.9, journal 4.9, @issueNumber 4.9, @abstract 4.9, institution 4.8, @e-mail 4.6, editor 4.4, @homepage 4.3, publisher 4.3, @ISBN 4.3, book 4.2, firstEditor 4.2, country 4.2, secondEditor 4.1, series 4.1

- →**Thing**: author 5.1, cites 5.1, primaryAuthor 5.1, @name 5.1, @numberOfCitations 5.1, @publicationYear 5.0, firstAuthor 5.0, @subject 5.0, venue 5.0, @pageNumbers 5.0, secondAuthor 5.0, @DOI 5.0, @numberOfPublications 4.9, conference 4.9, proceedings 4.9, @volumeNumber 4.9, journal 4.9, @issueNumber 4.9, @abstract 4.9, institution 4.8, @e-mail 4.6, editor 4.4, @homepage 4.3, publisher 4.3, @ISBN 4.3, book 4.2, firstEditor 4.2, country 4.2, secondEditor 4.1, series 4.1
- ←**Venue**: @numberOfPublications 6.1, @subject 5.4, @numberOfCitations 5.2, @name 5.2
- →**Venue**: venue 7.1, conference 7.0, journal 6.9

A.2 Association knowledge between classes themselves with distance = 1

Below is a list of classes with their associated classes by a path length of one. The association degree, computed using PMI, is also presented.

- ←**Article**: →Journal 5.9, →Thing 5.3, →N̂umber 5.2, →Author 5.2, →Person 5.2, →L̂iteral 5.2, →Venue 5.1, →Ŷear 5.1, →Article 5.1, →D̂OI 5.1, →D̂ate 5.0, →Paper 5.0, →T̂ext 4.9, →Ŝubject 4.8, →Editor 4.8, →Publication 4.8, →InProceedings 4.8, →Âbstract 4.7, →N̂ame 4.6, →Book 4.3, →InBook 4.2, →Institution 3.9, →Thesis 3.4, →Proceedings -2.0
- →**Article**: ←Paper 5.1, ←Publication 5.1, ←Article 5.1, ←Thing 5.0, ←InProceedings 5.0, ←InBook 4.6, ←Book 3.8, ←Thesis 3.3, ←Proceedings 0.1
- ←**Author**: →Ê-mail 6.6, →Ĥomepage 6.4, →N̂ame 6.1, →N̂umber 6.1, →L̂iteral 5.8, →T̂ext 5.7, →Institution 5.5, →Ŝubject 5.2, →Thing 5.1
- →**Author**: ←Publication 5.3, ←Paper 5.3, ←InProceedings 5.3, ←Thing 5.2, ←Article 5.2, ←Book 4.6, ←InBook 4.6, ←Proceedings 4.6, ←Thesis 4.4
- ←**Book**: →ÎSBN 8.4, →Publisher 8.4, →Series 7.9, →Editor 4.9, →Ŷear 4.9, →D̂ate 4.8, →Author 4.6, →Person 4.6, →Thing 4.6, →L̂iteral 4.5, →N̂umber 4.4, →N̂ame 4.4, →T̂ext 4.2, →Book 4.0, →InBook 4.0, →Paper 3.8, →Article 3.8, →InProceedings 3.7, →Publication 3.7, →Thesis 2.6, →Institution 0.1, →Proceedings -0.2
- →**Book**: ←InBook 7.3, ←Publication 4.7, ←Thing 4.6, ←Paper 4.5, ←InProceedings 4.5, ←Article 4.3, ←Book 4.0, ←Thesis 3.1, ←Proceedings -0.1

- ←**Conference**: →Ŝubject 5.3, →N̂ame 5.2, →N̂umber 5.1, →T̂ext 5.0, →L̂iteral 5.0, →Thing 4.2

- →**Conference**: ←InProceedings 5.6, ←Paper 5.1, ←Publication 5.1, ←Thing 5.0, ←Proceedings 4.8

- ←**Country**: →N̂ame 4.2, →N̂umber 4.2, →L̂iteral 3.8, →T̂ext 3.5, →Thing 3.1

- →**Country**: ←Institution 9.6, ←Thing 4.2

- ←**Editor**: →Ĥomepage 8.1, →Ê-mail 7.4, →Institution 6.4, →T̂ext 5.4, →Ŝubject 5.4, →L̂iteral 5.2, →N̂ame 5.1, →N̂umber 5.1, →Thing 4.6

- →**Editor**: ←Proceedings 6.1, ←InProceedings 5.2, ←Publication 5.1, ←Paper 5.1, ←Thing 5.0, ←InBook 5.0, ←Book 4.9, ←Article 4.8, ←Thesis 3.4

- ←**InBook**: →Book 7.3, →InBook 5.1, →Editor 5.0, →Ŷear 5.0, →Publication 4.9, →D̂OI 4.9, →D̂ate 4.9, →Thing 4.7, →T̂ext 4.7, →L̂iteral 4.7, →Author 4.6, →Person 4.6, →Paper 4.6, →Article 4.6, →Ŝubject 4.5, →N̂ame 4.5, →InProceedings 4.5, →N̂umber 4.4, →Thesis 3.7, →Âbstract 3.1, →Institution 2.7, →Proceedings -2.4

- →**InBook**: ←InBook 5.1, ←Publication 4.4, ←Paper 4.4, ←InProceedings 4.4, ←Thing 4.3, ←Article 4.2, ←Book 4.0, ←Thesis 3.2, ←Proceedings -0.3

- ←**InProceedings**: →Conference 5.6, →Proceedings 5.5, →Publication 5.4, →Author 5.3, →Person 5.3, →Thing 5.3, →InProceedings 5.3, →Paper 5.2, →Editor 5.2, →Venue 5.1, →Ŷear 5.1, →Âbstract 5.0, →D̂ate 5.0, →Article 5.0, →T̂ext 5.0, →D̂OI 5.0, →L̂iteral 4.9, →Ŝubject 4.9, →Institution 4.8, →N̂ame 4.6, →N̂umber 4.6, →Book 4.5, →InBook 4.4, →Thesis 3.9

- →**InProceedings**: ←InProceedings 5.3, ←Paper 5.1, ←Publication 5.1, ←Thing 5.0, ←Article 4.8, ←InBook 4.5, ←Book 3.7, ←Thesis 3.3, ←Proceedings -0.3

- ←**Institution**: →Country 9.6, →N̂ame 5.4, →N̂umber 5.4, →L̂iteral 5.0, →T̂ext 4.7, →Thing 4.5

- →**Institution**: ←Editor 6.4, ←Thesis 6.1, ←Author 5.5, ←Person 5.5, ←InProceedings 4.8, ←Thing 4.8, ←Publication 4.6, ←Paper 4.6, ←Article 3.9, ←InBook 2.7, ←Book 0.1

- ←**Journal**: →Ŝubject 5.1, →N̂ame 4.8, →N̂umber 4.7, →T̂ext 4.7, →L̂iteral 4.6, →Thing 3.9

- →**Journal**: ←Article 5.9, ←Paper 5.1, ←Publication 5.1, ←Thing 5.0

- ←**Paper**: →Thing 5.4, →Author 5.3, →Person 5.3, →Publication 5.2, →Venue 5.2, →Paper 5.2, →Ŷear 5.2, →InProceedings 5.1, →Conference 5.1, →L̂iteral 5.1, →Journal 5.1, →D̂ate 5.1,

→Editor 5.1, →Article 5.1, →D̂OI 5.1, →Proceedings 5.0, →T̂ext 5.0, →N̂umber 5.0, →Âbstract 5.0, →Ŝubject 5.0, →N̂ame 4.7, →Institution 4.6, →Book 4.5, →InBook 4.4, →Thesis 3.8

- →**Paper**: ←InProceedings 5.2, ←Paper 5.2, ←Publication 5.2, ←Thing 5.1, ←Article 5.0, ←InBook 4.6, ←Book 3.8, ←Thesis 3.4, ←Proceedings -0.0
- ←**Person**: →Ê-mail 6.6, →Ĥomepage 6.4, →N̂ame 6.1, →N̂umber 6.1, →L̂iteral 5.8, →T̂ext 5.7, →Institution 5.5, →Ŝubject 5.2, →Thing 5.1
- →**Person**: ←Publication 5.3, ←Paper 5.3, ←InProceedings 5.3, ←Thing 5.2, ←Article 5.2, ←Book 4.6, ←InBook 4.6, ←Proceedings 4.6, ←Thesis 4.4
- ←**Proceedings**: →Publisher 8.5, →Series 8.4, →ÎSBN 8.4, →Editor 6.1, →Ŷear 5.0, →D̂ate 4.9, →N̂umber 4.8, →Conference 4.8, →Thing 4.7, →L̂iteral 4.7, →Author 4.6, →Person 4.6, →N̂ame 4.5, →Venue 4.3, →T̂ext 3.8, →Article 0.1, →Paper -0.0, →Book -0.1, →Publication -0.2, →InProceedings -0.3, →InBook -0.3, →Proceedings -3.7
- →**Proceedings**: ←InProceedings 5.5, ←Paper 5.0, ←Publication 5.0, ←Thing 4.9, ←Book -0.2, ←Article -2.0, ←InBook -2.4, ←Proceedings -3.7
- ←**Publication**: →Thing 5.4, →Author 5.3, →Person 5.3, →Publication 5.2, →Paper 5.2, →Ŷear 5.2, →Venue 5.2, →InProceedings 5.1, →Editor 5.1, →Conference 5.1, →L̂iteral 5.1, →D̂ate 5.1, →Article 5.1, →Journal 5.1, →D̂OI 5.1, →T̂ext 5.0, →Proceedings 5.0, →N̂umber 5.0, →Âbstract 5.0, →Ŝubject 4.9, →N̂ame 4.7, →Book 4.7, →Institution 4.6, →Publisher 4.4, →InBook 4.4, →ÎSBN 4.4, →Series 4.2, →Thesis 3.8
- ←**Publication**: →Thing 5.4, →Author 5.3, →Person 5.3, →Publication 5.2, →Paper 5.2, →Ŷear 5.2, →Venue 5.2, →InProceedings 5.1, →Editor 5.1, →Conference 5.1, →L̂iteral 5.1, →D̂ate 5.1, →Article 5.1, →Journal 5.1, →D̂OI 5.1, →T̂ext 5.0, →Proceedings 5.0, →N̂umber 5.0, →Âbstract 5.0, →Ŝubject 4.9, →N̂ame 4.7, →Book 4.7, →Institution 4.6, →Publisher 4.4, →InBook 4.4, →ÎSBN 4.4, →Series 4.2, →Thesis 3.8
- →**Publication**: ←InProceedings 5.4, ←Paper 5.2, ←Publication 5.2, ←Thing 5.1, ←InBook 4.9, ←Article 4.8, ←Book 3.7, ←Thesis 3.3, ←Proceedings -0.2
- ←**Publisher**: →N̂ame 5.6, →T̂ext 4.9, →L̂iteral 4.1, →Thing 3.3
- →**Publisher**: ←Proceedings 8.5, ←Book 8.4, ←Publication 4.4, ←Thing 4.3

- ←**Series**: →N̂ame 5.4, →T̂ext 4.7, →L̂iteral 3.9, →Thing 3.2
- →**Series**: ←Proceedings 8.4, ←Book 7.9, ←Publication 4.2, ←Thing 4.1
- ←**Thesis**: →Institution 6.1, →Ŷear 5.0, →D̂ate 4.9, →Thesis 4.6, →N̂ame 4.5, →L̂iteral 4.5, →Thing 4.5, →Ŝubject 4.4, →Author 4.4, →Person 4.4, →N̂umber 4.3, →T̂ext 4.3, →Editor 3.4, →Paper 3.4, →InProceedings 3.3, →Article 3.3, →Publication 3.3, →InBook 3.2, →Book 3.1, →Âbstract 2.5
- →**Thesis**: ←Thesis 4.6, ←InProceedings 3.9, ←Publication 3.8, ←Paper 3.8, ←InBook 3.7, ←Thing 3.7, ←Article 3.4, ←Book 2.6
- ←**Thing**: →Thing 5.4, →L̂iteral 5.3, →N̂umber 5.3, →Author 5.2, →Person 5.2, →T̂ext 5.2, →Publication 5.1, →Paper 5.1, →Ŷear 5.1, →Venue 5.1, →N̂ame 5.1, →InProceedings 5.0, →Editor 5.0, →Conference 5.0, →D̂ate 5.0, →Ŝubject 5.0, →Article 5.0, →Journal 5.0, →D̂OI 5.0, →Proceedings 4.9, →Âbstract 4.9, →Institution 4.8, →Ê-mail 4.6, →Book 4.6, →Ĥomepage 4.3, →Publisher 4.3, →InBook 4.3, →ÎSBN 4.3, →Country 4.2, →Series 4.1, →Thesis 3.7
- →**Thing**: ←Thing 5.4, ←Publication 5.4, ←Paper 5.4, ←InProceedings 5.3, ←Article 5.3, ←Author 5.1, ←Person 5.1, ←InBook 4.7, ←Proceedings 4.7, ←Editor 4.6, ←Book 4.6, ←Institution 4.5, ←Thesis 4.5, ←Venue 4.3, ←Conference 4.2, ←Journal 3.9, ←Publisher 3.3, ←Series 3.2, ←Country 3.1
- ←**Venue**: →Ŝubject 5.4, →N̂ame 5.2, →N̂umber 5.2, →T̂ext 5.0, →L̂iteral 5.0, →Thing 4.3
- →**Venue**: ←Paper 5.2, ←Publication 5.2, ←InProceedings 5.1, ←Article 5.1, ←Thing 5.1, ←Proceedings 4.3

A.3 Association knowledge between classes themselves with distance ≤ 3

Below is a list of classes with their associated classes by a maximum path length of three. The association degree, computed using PMI, is also presented.

- ←**Article**: →Journal 5.9, →Thing 5.3, →N̂umber 5.2, →Author 5.2, →Person 5.2, →L̂iteral 5.2, →Venue 5.1, →Ŷear 5.1, →Article 5.1, →D̂OI 5.1, →D̂ate 5.0, →Paper 5.0, →T̂ext 4.9, →Ŝubject 4.8, →Editor 4.8, →Publication 4.8, →InProceedings 4.8, →Âbstract 4.7, →N̂ame 4.6, →Book

4.3, →InBook 4.2, →Institution 3.9, →Thesis 3.4, →Ê-mail 3.4, ←Publication 3.1, ←Paper 3.1, →Ĥomepage 3.1, ←InProceedings 3.1, ←Article 3.0, ←Thing 3.0, ←Book 2.4, ←Proceedings 2.4, ←InBook 2.4, ←Thesis 2.2, →Country 2.1, →Conference 1.8, →Proceedings 1.7, →ÎSBN 1.4, →Publisher 1.4, →Series 0.9, ←Editor 0.5, ←Institution -0.1, ←Author -0.4, ←Person -0.4

- →**Article**: ←Paper 5.1, ←Publication 5.1, ←Article 5.1, ←Thing 5.0, ←InProceedings 5.0, ←InBook 4.6, →Thing 4.0, →Author 3.9, →Person 3.9, →Publication 3.8, →Venue 3.8, →Paper 3.8, ←Book 3.8, →Ŷear 3.8, →Journal 3.8, →InProceedings 3.7, →Conference 3.7, →Editor 3.7, →L̂iteral 3.7, →D̂ate 3.7, →Article 3.7, →D̂OI 3.7, →Proceedings 3.6, →T̂ext 3.6, →N̂umber 3.6, →Âbstract 3.6, →Ŝubject 3.6, →N̂ame 3.3, ←Thesis 3.3, →Book 3.3, →Institution 3.2, →Publisher 3.0, →InBook 3.0, →ÎSBN 3.0, →Series 2.8, →Thesis 2.4, →Ê-mail 2.1, →Ĥomepage 1.8, →Country 1.5, ←Proceedings 1.3, ←Editor -0.2, ←Author -1.1, ←Person -1.1

- ←**Author**: →Ê-mail 6.6, →Ĥomepage 6.4, →N̂ame 6.1, →N̂umber 6.1, →L̂iteral 5.8, →T̂ext 5.7, →Institution 5.5, →Ŝubject 5.2, →Thing 5.1, →Country 3.8, ←Editor 2.1, ←Thesis 1.9, ←Institution 1.5, ←Author 1.2, ←Person 1.2, ←InProceedings 0.6, ←Thing 0.5, ←Publication 0.3, ←Paper 0.3, ←Article -0.4, →Conference -0.7, →Proceedings -0.8, →Author -0.8, →Person -0.8, →Publication -0.9, →Venue -0.9, →Paper -0.9, →Ŷear -1.0, →InProceedings -1.0, →Editor -1.0, →D̂ate -1.0, →Journal -1.0, →Article -1.1, →D̂OI -1.1, →Âbstract -1.2, →Book -1.5, ←InBook -1.6, →Publisher -1.8, →InBook -1.8, →ÎSBN -1.8, →Series -1.9, →Thesis -2.4, ←Book -2.6, ←Proceedings -2.7

- →**Author**: ←Publication 5.3, ←Paper 5.3, ←InProceedings 5.3, ←Thing 5.2, ←Article 5.2, ←Book 4.6, ←InBook 4.6, ←Proceedings 4.6, ←Thesis 4.4, →Thing 4.2, →Author 4.2, →Person 4.2, →Publication 4.1, →Venue 4.1, →Paper 4.1, →Ŷear 4.0, →Conference 4.0, →InProceedings 4.0, →Editor 4.0, →L̂iteral 4.0, →Journal 3.9, →D̂ate 3.9, →Article 3.9, →D̂OI 3.9, →Proceedings 3.9, →T̂ext 3.9, →N̂umber 3.9, →Âbstract 3.8, →Ŝubject 3.8, →N̂ame 3.6, →Book 3.5, →Institution 3.5, →Publisher 3.2, →InBook 3.2, →ÎSBN 3.2, →Series 3.1, →Thesis 2.6, →Ê-mail 2.3, →Ĥomepage 2.1, →Country 1.7, ←Editor 0.0, ←Author -0.8, ←Person -0.8

- ←**Book**: →ÎSBN 8.4, →Publisher 8.4, →Series 7.9, ←Proceedings 5.3, ←Book 5.3, →Editor 4.9, →Ŷear 4.9, →D̂ate 4.8, →Author 4.6, →Person 4.6, →Thing 4.6, →L̂iteral 4.5, →N̂umber 4.4, →N̂ame 4.4, →T̂ext 4.2, →Book 4.0, →InBook 4.0, →Paper 3.8, →Article 3.8, →InProceedings 3.7, →Publication 3.7, →Ê-mail 2.8, →Thesis 2.6, ←Publication 2.5, ←Paper 2.5, →Ĥomepage 2.5, ←InProceedings 2.5, ←Thing 2.4, ←Article 2.4, ←InBook 1.8, →Institution 1.6, ←Thesis

1.6, →Ŝubject 1.3, →Venue 1.2, →Conference 1.2, →Journal 1.1, →D̂OI 1.1, →Proceedings 1.1, →Âbstract 1.0, →Country -0.1, ←Editor -1.8, ←Author -2.6, ←Person -2.6, ←Institution -3.9

- →**Book**: ←InBook 7.3, ←Publication 4.7, ←Thing 4.6, ←Paper 4.5, ←InProceedings 4.5, →Book 4.4, ←Article 4.3, ←Book 4.0, →Thing 3.5, →Author 3.5, →Person 3.5, →Publication 3.4, →Paper 3.4, →Ŷear 3.4, →Venue 3.4, →InProceedings 3.3, →Editor 3.3, →Conference 3.3, →L̂iteral 3.3, →D̂ate 3.3, →Article 3.3, →Journal 3.3, →D̂OI 3.3, →T̂ext 3.2, →Proceedings 3.2, →N̂umber 3.2, ←Thesis 3.1, →Âbstract 3.1, →Ŝubject 3.1, →N̂ame 2.9, →Institution 2.8, →Publisher 2.6, →InBook 2.6, →ÎSBN 2.5, →Series 2.4, →Thesis 2.0, →Ê-mail 1.7, →Ĥomepage 1.4, →Country 1.1, ←Proceedings 0.9, ←Editor -0.6, ←Author -1.5, ←Person -1.5
- ←**Conference**: →Ŝubject 5.3, →N̂ame 5.2, →N̂umber 5.1, →T̂ext 5.0, →L̂iteral 5.0, →Thing 4.2
- →**Conference**: ←InProceedings 5.6, ←Paper 5.1, ←Publication 5.1, ←Thing 5.0, ←Proceedings 4.8, →Conference 4.3, →Proceedings 4.2, →Publication 4.1, →Author 4.0, →Person 4.0, →Thing 4.0, →InProceedings 4.0, →Paper 3.9, →Editor 3.9, →Venue 3.9, →Ŷear 3.8, →L̂iteral 3.8, →Âbstract 3.8, →Journal 3.8, →D̂ate 3.7, →Article 3.7, →D̂OI 3.7, →T̂ext 3.7, →N̂umber 3.7, →Ŝubject 3.6, →Institution 3.6, →N̂ame 3.4, →Book 3.3, →InBook 3.1, →Publisher 3.0, →ÎSBN 3.0, →Series 2.9, →Thesis 2.6, →Ê-mail 2.2, →Ĥomepage 1.9, ←InBook 1.9, →Country 1.8, ←Article 1.8, ←Book 1.2, ←Thesis 1.0, ←Editor 0.1, ←Author -0.7, ←Person -0.7
- ←**Country**: →N̂ame 4.2, →N̂umber 4.2, →L̂iteral 3.8, →T̂ext 3.5, →Thing 3.1
- →**Country**: ←Institution 9.6, →Country 6.3, ←Editor 4.6, ←Thesis 4.4, ←Thing 4.2, ←Author 3.8, ←Person 3.8, ←InProceedings 3.1, ←Publication 2.9, ←Paper 2.9, →N̂ame 2.2, ←Article 2.1, →N̂umber 2.1, →Conference 1.8, →Thing 1.8, →L̂iteral 1.7, →Proceedings 1.7, →Author 1.7, →Person 1.7, →Publication 1.6, →Venue 1.6, →Paper 1.6, →Ŷear 1.6, →InProceedings 1.5, →Editor 1.5, →D̂ate 1.5, →Journal 1.5, →Article 1.5, →D̂OI 1.5, →T̂ext 1.4, →Âbstract 1.4, →Ŝubject 1.3, →Book 1.1, →Institution 1.1, ←InBook 1.0, →Ê-mail 0.9, →Publisher 0.8, →InBook 0.8, →ÎSBN 0.8, →Ĥomepage 0.6, →Series 0.6, →Thesis 0.2, ←Book -0.1, ←Proceedings -0.1
- ←**Editor**: →Ĥomepage 8.1, →Ê-mail 7.4, →Institution 6.4, →T̂ext 5.4, →Ŝubject 5.4, →L̂iteral 5.2, →N̂ame 5.1, →N̂umber 5.1, →Country 4.6, →Thing 4.6, ←Editor 2.9, ←Thesis 2.7, ←Institution 2.4, ←Author 2.1, ←Person 2.1, ←InProceedings 1.4, ←Thing 1.4, ←Publication 1.2, ←Paper 1.2, ←Article 0.5, →Conference 0.1, →Proceedings 0.0, →Author 0.0, →Person 0.0, →Publication -0.1,

→Venue -0.1, →Paper -0.1, →Ŷear -0.1, →InProceedings -0.2, →Editor -0.2, →D̂ate -0.2, →Journal -0.2, →Article -0.2, →D̂OI -0.2, →Âbstract -0.3, →Book -0.6, ←InBook -0.7, →Publisher -0.9, →InBook -0.9, →ÎSBN -0.9, →Series -1.1, →Thesis -1.5, ←Book -1.8, ←Proceedings -1.8

- →**Editor**: ←Proceedings 6.1, ←InProceedings 5.2, ←Publication 5.1, ←Paper 5.1, ←Thing 5.0, ←InBook 5.0, ←Book 4.9, ←Article 4.8, →Thing 4.0, →Author 4.0, →Person 4.0, →Conference 3.9, →Publication 3.9, →Paper 3.8, →Ŷear 3.8, →Venue 3.8, →Proceedings 3.8, →InProceedings 3.8, →Editor 3.8, →L̂iteral 3.7, →D̂ate 3.7, →Article 3.7, →Journal 3.7, →D̂OI 3.7, →T̂ext 3.7, →N̂umber 3.6, →Âbstract 3.6, →Ŝubject 3.6, ←Thesis 3.4, →N̂ame 3.3, →Book 3.3, →Institution 3.2, →Publisher 3.0, →InBook 3.0, →ÎSBN 3.0, →Series 2.9, →Thesis 2.4, →Ê-mail 2.1, →Ĥomepage 1.8, →Country 1.5, ←Editor -0.2, ←Author -1.0, ←Person -1.0

- ←**InBook**: →Book 7.3, →InBook 5.1, →Editor 5.0, →Ŷear 5.0, →Publication 4.9, →D̂OI 4.9, →D̂ate 4.9, →Thing 4.7, →T̂ext 4.7, →L̂iteral 4.7, →Author 4.6, →Person 4.6, →Paper 4.6, →Article 4.6, →Ŝubject 4.5, →N̂ame 4.5, →InProceedings 4.5, →N̂umber 4.4, →ÎSBN 4.4, →Publisher 4.4, →Series 3.9, →Thesis 3.7, ←InBook 3.4, →Âbstract 3.1, →Ê-mail 2.8, →Institution 2.7, ←Proceedings 2.6, ←Publication 2.5, ←Paper 2.5, →Ĥomepage 2.5, ←InProceedings 2.5, ←Thing 2.4, ←Article 2.4, →Venue 1.9, →Conference 1.9, →Journal 1.8, ←Book 1.8, →Proceedings 1.8, ←Thesis 1.6, →Country 1.0, ←Editor -0.7, ←Institution -1.3, ←Author -1.6, ←Person -1.6

- →**InBook**: ←InBook 5.1, ←Publication 4.4, ←Paper 4.4, ←InProceedings 4.4, ←Thing 4.3, ←Article 4.2, ←Book 4.0, →Thing 3.2, →Author 3.2, →Person 3.2, ←Thesis 3.2, →Publication 3.1, →Venue 3.1, →Paper 3.1, →Ŷear 3.1, →Conference 3.1, →InProceedings 3.0, →Editor 3.0, →L̂iteral 3.0, →D̂ate 3.0, →Journal 3.0, →Article 3.0, →D̂OI 3.0, →Proceedings 3.0, →T̂ext 2.9, →N̂umber 2.9, →Âbstract 2.9, →Ŝubject 2.8, →N̂ame 2.6, →Book 2.6, →Institution 2.5, →Publisher 2.3, →InBook 2.3, →ÎSBN 2.3, →Series 2.1, →Thesis 1.7, →Ê-mail 1.4, →Ĥomepage 1.1, →Country 0.8, ←Proceedings 0.6, ←Editor -0.9, ←Author -1.8, ←Person -1.8

- ←**InProceedings**: →Conference 5.6, →Proceedings 5.5, →Publication 5.4, →Author 5.3, →Person 5.3, →Thing 5.3, →InProceedings 5.3, →Paper 5.2, →Editor 5.2, →Venue 5.1, →Ŷear 5.1, →Âbstract 5.0, →D̂ate 5.0, →Article 5.0, →T̂ext 5.0, →D̂OI 5.0, →L̂iteral 4.9, →Ŝubject 4.9, →Institution 4.8, →N̂ame 4.6, →N̂umber 4.6, →Book 4.5, →InBook 4.4, →Thesis 3.9, →Ê-mail 3.5, →Publisher 3.4, →Series 3.4, →ÎSBN 3.3, ←Publication 3.2, ←Paper 3.2, →Ĥomepage 3.2, ←InProceedings 3.2, →Country 3.1, ←Thing 3.1, ←Article 3.1, ←Proceedings 2.7, ←Book 2.5,

←InBook 2.5, →Journal 2.3, ←Thesis 2.3, ←Editor 1.4, ←Institution 0.9, ←Author 0.6, ←Person 0.6

- →**InProceedings**: ←InProceedings 5.3, ←Paper 5.1, ←Publication 5.1, ←Thing 5.0, ←Article 4.8, ←InBook 4.5, →Thing 4.0, →Author 4.0, →Person 4.0, →Conference 4.0, →Publication 3.9, →Venue 3.9, →Paper 3.9, →Proceedings 3.9, →Ŷear 3.8, →InProceedings 3.8, →L̂iteral 3.8, →Editor 3.8, →Journal 3.8, →D̂ate 3.8, →Article 3.7, →D̂OI 3.7, ←Book 3.7, →T̂ext 3.7, →N̂umber 3.7, →Âbstract 3.6, →Ŝubject 3.6, →N̂ame 3.4, ←Thesis 3.3, →Book 3.3, →Institution 3.3, →Publisher 3.0, →InBook 3.0, →ÎSBN 3.0, →Series 2.9, →Thesis 2.4, →Ê-mail 2.2, →Ĥomepage 1.9, →Country 1.5, ←Proceedings 1.3, ←Editor -0.2, ←Author -1.0, ←Person -1.0

- ←**Institution**: →Country 9.6, ←Institution 7.3, →N̂ame 5.4, →N̂umber 5.4, →L̂iteral 5.0, →T̂ext 4.7, →Thing 4.5, ←Editor 2.4, ←Thesis 2.1, ←Thing 1.9, ←Author 1.5, ←Person 1.5, ←InProceedings 0.9, ←Publication 0.6, ←Paper 0.6, ←Article -0.1, ←InBook -1.3, ←Book -3.9

- →**Institution**: ←Editor 6.4, ←Thesis 6.1, ←Author 5.5, ←Person 5.5, ←InProceedings 4.8, ←Thing 4.8, ←Publication 4.6, ←Paper 4.6, ←Article 3.9, →Conference 3.6, →Thing 3.5, →Proceedings 3.5, →Author 3.5, →Person 3.5, →Publication 3.4, →Venue 3.3, →Paper 3.3, →Ŷear 3.3, →InProceedings 3.3, →Editor 3.2, →L̂iteral 3.2, →D̂ate 3.2, →Journal 3.2, →Article 3.2, →D̂OI 3.2, →T̂ext 3.1, →N̂umber 3.1, →Âbstract 3.1, →Ŝubject 3.1, →N̂ame 2.8, →Book 2.8, →Institution 2.8, ←InBook 2.7, →Ê-mail 2.6, →Publisher 2.5, →InBook 2.5, →ÎSBN 2.5, →Ĥomepage 2.4, →Series 2.3, →Thesis 1.9, ←Book 1.6, ←Proceedings 1.6, →Country 1.1

- ←**Journal**: →Ŝubject 5.1, →N̂ame 4.8, →N̂umber 4.7, →T̂ext 4.7, →L̂iteral 4.6, →Thing 3.9

- →**Journal**: ←Article 5.9, ←Paper 5.1, ←Publication 5.1, ←Thing 5.0, →Journal 4.6, →Thing 4.0, →N̂umber 4.0, →Author 3.9, →Person 3.9, →L̂iteral 3.9, →Publication 3.9, →Venue 3.8, →Paper 3.8, →Ŷear 3.8, →Article 3.8, →D̂OI 3.8, →InProceedings 3.8, →Conference 3.8, →D̂ate 3.7, →Editor 3.7, →Proceedings 3.7, →T̂ext 3.6, →Âbstract 3.6, →Ŝubject 3.6, →N̂ame 3.3, →Book 3.3, →Institution 3.2, →InBook 3.0, →Publisher 3.0, →ÎSBN 3.0, →Series 2.8, →Thesis 2.4, ←InProceedings 2.3, →Ê-mail 2.1, →Ĥomepage 1.8, ←InBook 1.8, →Country 1.5, ←Proceedings 1.3, ←Book 1.1, ←Thesis 0.9, ←Editor -0.2, ←Author -1.0, ←Person -1.0

- ←**Paper**: →Thing 5.4, →Author 5.3, →Person 5.3, →Publication 5.2, →Venue 5.2, →Paper 5.2, →Ŷear 5.2, →InProceedings 5.1, →Conference 5.1, →L̂iteral 5.1, →Journal 5.1, →D̂ate 5.1,

→Editor 5.1, →Article 5.1, →D̂OI 5.1, →Proceedings 5.0, →T̂ext 5.0, →N̂umber 5.0, →Âbstract 5.0, →Ŝubject 5.0, →N̂ame 4.7, →Institution 4.6, →Book 4.5, →InBook 4.4, →Thesis 3.8, →Ê-mail 3.5, ←Publication 3.2, ←Paper 3.2, →Ĥomepage 3.2, ←InProceedings 3.2, ←Thing 3.1, ←Article 3.1, →Publisher 2.9, →Country 2.9, →Series 2.9, →ÎSBN 2.8, ←Proceedings 2.7, ←Book 2.5, ←InBook 2.5, ←Thesis 2.3, ←Editor 1.2, ←Institution 0.6, ←Author 0.3, ←Person 0.3

- →**Paper**: ←InProceedings 5.2, ←Paper 5.2, ←Publication 5.2, ←Thing 5.1, ←Article 5.0, ←InBook 4.6, →Thing 4.1, →Author 4.1, →Person 4.1, →Publication 4.0, →Conference 3.9, →Venue 3.9, →Paper 3.9, →Ŷear 3.9, →InProceedings 3.9, →Proceedings 3.8, →Editor 3.8, →L̂iteral 3.8, ←Book 3.8, →Journal 3.8, →D̂ate 3.8, →Article 3.8, →D̂OI 3.8, →T̂ext 3.7, →N̂umber 3.7, →Âbstract 3.7, →Ŝubject 3.7, →N̂ame 3.4, ←Thesis 3.4, →Book 3.4, →Institution 3.3, →Publisher 3.1, →InBook 3.1, →ÎSBN 3.1, →Series 2.9, →Thesis 2.5, →Ê-mail 2.2, →Ĥomepage 1.9, →Country 1.6, ←Proceedings 1.4, ←Editor -0.1, ←Author -0.9, ←Person -0.9

- ←**Person**: →Ê-mail 6.6, →Ĥomepage 6.4, →N̂ame 6.1, →N̂umber 6.1, →L̂iteral 5.8, →T̂ext 5.7, →Institution 5.5, →Ŝubject 5.2, →Thing 5.1, →Country 3.8, ←Editor 2.1, ←Thesis 1.9, ←Institution 1.5, ←Author 1.2, ←Person 1.2, ←InProceedings 0.6, ←Thing 0.5, ←Publication 0.3, ←Paper 0.3, ←Article -0.4, →Conference -0.7, →Proceedings -0.8, →Author -0.8, →Person -0.8, →Publication -0.9, →Venue -0.9, →Paper -0.9, →Ŷear -1.0, →InProceedings -1.0, →Editor -1.0, →D̂ate -1.0, →Journal -1.0, →Article -1.1, →D̂OI -1.1, →Âbstract -1.2, →Book -1.5, ←InBook -1.6, →Publisher -1.8, →InBook -1.8, →ÎSBN -1.8, →Series -1.9, →Thesis -2.4, ←Book -2.6, ←Proceedings -2.7

- →**Person**: ←Publication 5.3, ←Paper 5.3, ←InProceedings 5.3, ←Thing 5.2, ←Article 5.2, ←Book 4.6, ←InBook 4.6, ←Proceedings 4.6, ←Thesis 4.4, →Thing 4.2, →Author 4.2, →Person 4.2, →Publication 4.1, →Venue 4.1, →Paper 4.1, →Ŷear 4.0, →Conference 4.0, →InProceedings 4.0, →Editor 4.0, →L̂iteral 4.0, →Journal 3.9, →D̂ate 3.9, →Article 3.9, →D̂OI 3.9, →Proceedings 3.9, →T̂ext 3.9, →N̂umber 3.9, →Âbstract 3.8, →Ŝubject 3.8, →N̂ame 3.6, →Book 3.5, →Institution 3.5, →Publisher 3.2, →InBook 3.2, →ÎSBN 3.2, →Series 3.1, →Thesis 2.6, →Ê-mail 2.3, →Ĥomepage 2.1, →Country 1.7, ←Editor 0.0, ←Author -0.8, ←Person -0.8

- ←**Proceedings**: →Publisher 8.5, →Series 8.4, →ÎSBN 8.4, →Editor 6.1, ←Proceedings 5.4, ←Book 5.3, →Ŷear 5.0, →D̂ate 4.9, →N̂umber 4.8, →Conference 4.8, →Thing 4.7, →L̂iteral 4.7, →Author 4.6, →Person 4.6, →N̂ame 4.5, →Venue 4.3, →T̂ext 3.8, →Ĥomepage 3.0, →Ê-mail 2.7, ←InProceedings 2.7, ←Publication 2.7, ←Paper 2.7, ←Thing 2.6, ←InBook 2.6, ←Article 2.4,

→Institution 1.6, ←Thesis 1.5, →Publication 1.4, →Paper 1.4, →Proceedings 1.3, →InProceedings 1.3, →Article 1.3, →Journal 1.3, →D̂OI 1.3, →Ŝubject 1.3, →Âbstract 1.2, →Book 0.9, →InBook 0.6, →Thesis -0.0, →Country -0.1, ←Editor -1.8, ←Author -2.7, ←Person -2.7

- →**Proceedings**: ←InProceedings 5.5, ←Paper 5.0, ←Publication 5.0, ←Thing 4.9, →Conference 4.2, →Proceedings 4.1, →Publication 4.0, →Author 3.9, →Person 3.9, →Thing 3.9, →InProceedings 3.9, →Paper 3.8, →Editor 3.8, →Venue 3.8, →Ŷear 3.7, →L̂iteral 3.7, →Âbstract 3.7, →Journal 3.7, →D̂ate 3.6, →Article 3.6, →D̂OI 3.6, →T̂ext 3.6, →N̂umber 3.6, →Ŝubject 3.5, →Institution 3.5, →N̂ame 3.3, →Book 3.2, →InBook 3.0, →Publisher 2.9, →ÎSBN 2.9, →Series 2.7, →Thesis 2.5, →Ê-mail 2.1, →Ĥomepage 1.8, ←InBook 1.8, →Country 1.7, ←Article 1.7, ←Proceedings 1.3, ←Book 1.1, ←Thesis 0.9, ←Editor 0.0, ←Author -0.8, ←Person -0.8

- ←**Publication**: →Thing 5.4, →Author 5.3, →Person 5.3, →Publication 5.2, →Paper 5.2, →Ŷear 5.2, →Venue 5.2, →InProceedings 5.1, →Editor 5.1, →Conference 5.1, →L̂iteral 5.1, →D̂ate 5.1, →Article 5.1, →Journal 5.1, →D̂OI 5.1, →T̂ext 5.0, →Proceedings 5.0, →N̂umber 5.0, →Âbstract 5.0, →Ŝubject 4.9, →N̂ame 4.7, →Book 4.7, →Institution 4.6, →Publisher 4.4, →InBook 4.4, →ÎSBN 4.4, →Series 4.2, →Thesis 3.8, →Ê-mail 3.5, ←Publication 3.2, ←Paper 3.2, →Ĥomepage 3.2, ←InProceedings 3.2, ←Thing 3.1, ←Article 3.1, →Country 2.9, ←Proceedings 2.7, ←Book 2.5, ←InBook 2.5, ←Thesis 2.3, ←Editor 1.2, ←Institution 0.6, ←Author 0.3, ←Person 0.3

- →**Publication**: ←InProceedings 5.4, ←Paper 5.2, ←Publication 5.2, ←Thing 5.1, ←InBook 4.9, ←Article 4.8, →Thing 4.1, →Author 4.1, →Person 4.1, →Conference 4.1, →Publication 4.0, →Proceedings 4.0, →Venue 4.0, →Paper 4.0, →Ŷear 3.9, →InProceedings 3.9, →Editor 3.9, →L̂iteral 3.9, →Journal 3.9, →D̂ate 3.8, →Article 3.8, →D̂OI 3.8, →T̂ext 3.8, →N̂umber 3.8, →Âbstract 3.7, →Ŝubject 3.7, ←Book 3.7, →N̂ame 3.5, →Book 3.4, →Institution 3.4, ←Thesis 3.3, →Publisher 3.1, →InBook 3.1, →ÎSBN 3.1, →Series 3.0, →Thesis 2.5, →Ê-mail 2.3, →Ĥomepage 2.0, →Country 1.6, ←Proceedings 1.4, ←Editor -0.1, ←Author -0.9, ←Person -0.9

- ←**Publisher**: →N̂ame 5.6, →T̂ext 4.9, →L̂iteral 4.1, →Thing 3.3

- →**Publisher**: ←Proceedings 8.5, ←Book 8.4, →ÎSBN 5.4, →Publisher 5.4, →Series 4.9, ←Publication 4.4, ←InBook 4.4, ←Thing 4.3, ←InProceedings 3.4, →Thing 3.2, →Author 3.2, →Person 3.2, →Publication 3.1, →Paper 3.1, →Ŷear 3.1, →Venue 3.1, →InProceedings 3.0, →Editor 3.0, →Conference 3.0, →L̂iteral 3.0, →D̂ate 3.0, →Article 3.0, →Journal 3.0, →D̂OI 3.0, →T̂ext 2.9, →Proceedings 2.9, →N̂umber 2.9, ←Paper 2.9, →Âbstract 2.9, →Ŝubject 2.8, →N̂ame 2.6, →Book

2.6, →Institution 2.5, →InBook 2.3, →Thesis 1.7, ←Article 1.4, →Ê-mail 1.4, →Ĥomepage 1.1, →Country 0.8, ←Thesis 0.2, ←Editor -0.9, ←Author -1.8, ←Person -1.8

- →**Publisher**: ←Proceedings 8.5, ←Book 8.4, →ÎSBN 5.4, →Publisher 5.4, →Series 4.9, ←Publication 4.4, ←InBook 4.4, ←Thing 4.3, ←InProceedings 3.4, →Thing 3.2, →Author 3.2, →Person 3.2, →Publication 3.1, →Paper 3.1, →Ŷear 3.1, →Venue 3.1, →InProceedings 3.0, →Editor 3.0, →Conference 3.0, →L̂iteral 3.0, →D̂ate 3.0, →Article 3.0, →Journal 3.0, →D̂OI 3.0, →T̂ext 2.9, →Proceedings 2.9, →N̂umber 2.9, ←Paper 2.9, →Âbstract 2.9, →Ŝubject 2.8, →N̂ame 2.6, →Book 2.6, →Institution 2.5, →InBook 2.3, →Thesis 1.7, ←Article 1.4, →Ê-mail 1.4, →Ĥomepage 1.1, →Country 0.8, ←Thesis 0.2, ←Editor -0.9, ←Author -1.8, ←Person -1.8
- ←**Series**: →N̂ame 5.4, →T̂ext 4.7, →L̂iteral 3.9, →Thing 3.2
- →**Series**: ←Proceedings 8.4, ←Book 7.9, →Publisher 4.9, →ÎSBN 4.9, →Series 4.9, ←Publication 4.2, ←Thing 4.1, ←InBook 3.9, ←InProceedings 3.4, →Thing 3.1, →Author 3.1, →Person 3.1, →Publication 3.0, →Paper 2.9, →Ŷear 2.9, →Venue 2.9, ←Paper 2.9, →InProceedings 2.9, →Editor 2.9, →Conference 2.9, →L̂iteral 2.8, →D̂ate 2.8, →Article 2.8, →Journal 2.8, →D̂OI 2.8, →T̂ext 2.7, →Proceedings 2.7, →N̂umber 2.7, →Âbstract 2.7, →Ŝubject 2.7, →N̂ame 2.4, →Book 2.4, →Institution 2.3, →InBook 2.1, →Thesis 1.5, →Ê-mail 1.2, →Ĥomepage 0.9, ←Article 0.9, →Country 0.6, ←Thesis 0.0, ←Editor -1.1, ←Author -1.9, ←Person -1.9
- ←**Thesis**: →Institution 6.1, →Ŷear 5.0, →D̂ate 4.9, →Thesis 4.6, →N̂ame 4.5, →L̂iteral 4.5, →Thing 4.5, →Ŝubject 4.4, →Country 4.4, →Author 4.4, →Person 4.4, →N̂umber 4.3, →T̂ext 4.3, →Editor 3.4, →Paper 3.4, →InProceedings 3.3, →Article 3.3, →Publication 3.3, →InBook 3.2, →Book 3.1, ←Editor 2.7, →Ê-mail 2.6, →Âbstract 2.5, ←Thesis 2.5, ←Publication 2.3, ←Paper 2.3, →Ĥomepage 2.3, ←InProceedings 2.3, ←Thing 2.2, ←Article 2.2, ←Institution 2.1, ←Author 1.9, ←Person 1.9, ←Book 1.6, ←InBook 1.6, ←Proceedings 1.5, →Venue 1.0, →Conference 1.0, →Journal 0.9, →D̂OI 0.9, →Proceedings 0.9, →ÎSBN 0.3, →Publisher 0.2, →Series 0.0
- →**Thesis**: ←Thesis 4.6, ←InProceedings 3.9, ←Publication 3.8, ←Paper 3.8, ←InBook 3.7, ←Thing 3.7, ←Article 3.4, →Thing 2.6, →Author 2.6, →Person 2.6, →Conference 2.6, ←Book 2.6, →Publication 2.5, →Venue 2.5, →Paper 2.5, →Ŷear 2.5, →Proceedings 2.5, →InProceedings 2.4, →Editor 2.4, →L̂iteral 2.4, →D̂ate 2.4, →Journal 2.4, →Article 2.4, →D̂OI 2.4, →T̂ext 2.3, →N̂umber 2.3, →Âbstract 2.3, →Ŝubject 2.2, →N̂ame 2.0, →Book 2.0, →Institution 1.9,

→Publisher 1.7, →InBook 1.7, →ÎSBN 1.7, →Series 1.5, →Thesis 1.1, →Ê-mail 0.8, →Ĥomepage 0.5, →Country 0.2, ←Proceedings -0.0, ←Editor -1.5, ←Author -2.4, ←Person -2.4

- ←**Thing**: →L̂iteral 5.3, →N̂umber 5.3, →Author 5.2, →Person 5.2, →T̂ext 5.2, →Publication 5.1, →Paper 5.1, →Ŷear 5.1, →Venue 5.1, →N̂ame 5.1, →InProceedings 5.0, →Editor 5.0, →Conference 5.0, →D̂ate 5.0, →Ŝubject 5.0, →Article 5.0, →Journal 5.0, →D̂OI 5.0, →Proceedings 4.9, →Âbstract 4.9, →Institution 4.8, →Ê-mail 4.6, →Book 4.6, →Ĥomepage 4.3, →Publisher 4.3, →InBook 4.3, →ÎSBN 4.3, →Country 4.2, →Series 4.1, →Thesis 3.7, ←Publication 3.1, ←Paper 3.1, ←InProceedings 3.1, ←Article 3.0, ←Proceedings 2.6, ←Book 2.4, ←InBook 2.4, ←Thesis 2.2, ←Institution 1.9, ←Editor 1.4, ←Author 0.5, ←Person 0.5

- →**Thing**: ←Publication 5.4, ←Paper 5.4, ←InProceedings 5.3, ←Article 5.3, ←Author 5.1, ←Person 5.1, ←InBook 4.7, ←Proceedings 4.7, ←Editor 4.6, ←Book 4.6, ←Institution 4.5, ←Thesis 4.5, ←Venue 4.3, ←Conference 4.2, →Author 4.2, →Person 4.2, →Publication 4.1, →Venue 4.1, →Paper 4.1, →Ŷear 4.1, →InProceedings 4.0, →Conference 4.0, →Editor 4.0, →L̂iteral 4.0, →Journal 4.0, →D̂ate 4.0, →Article 4.0, →D̂OI 3.9, →Proceedings 3.9, →T̂ext 3.9, ←Journal 3.9, →N̂umber 3.9, →Âbstract 3.8, →Ŝubject 3.8, →N̂ame 3.6, →Book 3.5, →Institution 3.5, ←Publisher 3.3, →Publisher 3.2, →InBook 3.2, →ÎSBN 3.2, ←Series 3.2, ←Country 3.1, →Series 3.1, →Thesis 2.6, →Ê-mail 2.4, →Ĥomepage 2.1, →Country 1.8

- ←**Venue**: →Ŝubject 5.4, →N̂ame 5.2, →N̂umber 5.2, →T̂ext 5.0, →L̂iteral 5.0, →Thing 4.3

- →**Venue**: ←Paper 5.2, ←Publication 5.2, ←InProceedings 5.1, ←Article 5.1, ←Thing 5.1, ←Proceedings 4.3, →Thing 4.1, →Author 4.1, →Person 4.1, →Publication 4.0, →Venue 3.9, →Paper 3.9, →Ŷear 3.9, →InProceedings 3.9, →Conference 3.9, →L̂iteral 3.8, →Journal 3.8, →Editor 3.8, →D̂ate 3.8, →Article 3.8, →D̂OI 3.8, →Proceedings 3.8, →T̂ext 3.7, →N̂umber 3.7, →Âbstract 3.7, →Ŝubject 3.7, →N̂ame 3.4, →Book 3.4, →Institution 3.3, →InBook 3.1, →Publisher 3.1, →ÎSBN 3.1, →Series 2.9, →Thesis 2.5, →Ê-mail 2.2, →Ĥomepage 1.9, ←InBook 1.9, →Country 1.6, ←Book 1.2, ←Thesis 1.0, ←Editor -0.1, ←Author -0.9, ←Person -0.9

Appendix B

TOP-10 INTERPRETATIONS OF 220 DBLP+ TESTCASES

This appendix details top-10 interpretations of the 220 DBLP+ testcases, which are produced using the hybrid similarity and the optimal parameters as discussed in Section 9.3.6. The 64 natural language questions corresponding to the 220 testcases can be referenced in the Table 9.4.

Each query may contain one or more relations, which are separated by "||". Each relation in the query is mapped to a schema path. The top-10 interpretations are numbered from 0 to 9. An interpretations is serialized into a string, following a specific syntax. Inside the first brackets are the classes on the schema path and the "<" or ">" between two classes shows the direction of the property connecting the classes. Inside the second brackets are the properties on the schema path, which are in order with "<" or ">" in the first brackets. Sometimes you can find a "%" character appending to a class, which means this class is merged to another class with the same same in a different schema path. The ending number at each interpretation line is the fitness score of the interpretation.

(DS 1) ?x/Author, has, ?y/Paper

0. [Author < Paper]; [author] 17.80864
1. [Author < Article]; [author] 7.97759

2. [Author < Publication]; [author] 5.09145
3. [Person < Paper]; [author] 4.72045
4. [Editor < Paper]; [author] 4.17545
5. [Author < Paper < Paper]; [author, cites] 3.64557
6. [Author < Paper > Paper]; [author, cites] 3.64557
7. [Article > Person < Paper]; [author, author] 3.17234
8. [Person < Article]; [author] 2.33922
9. [Author < Book]; [author] 2.31013

(DS 1) ?x/Scholar, published ,?y/Paper

0. [Author < Paper]; [author] 2.19181
1. [Journal < Paper]; [journal] 1.40111
2. [Author < Article]; [author] 1.08612
3. [Editor < Paper]; [author] 1.08346
4. [Author < Paper]; [editor] 0.88985
5. [Author < Publication]; [author] 0.82260
6. [Author < Publication]; [editor] 0.81844
7. [Journal < Article]; [journal] 0.73178
8. [Author < Paper < Paper]; [author, cites] 0.56163
9. [Author < Paper > Paper]; [author, cites] 0.56163

(DS 1) ?x/Person, author of, ?y/Paper

0. [Person < Paper]; [author] 17.80864
1. [Person < Article]; [author] 8.82522
2. [Person < Publication]; [author] 6.68412
3. [Author < Paper]; [author] 4.72045
4. [Person < Paper < Paper]; [author, cites] 3.64557
5. [Person < Paper > Paper]; [author, cites] 3.64557
6. [Person < Book]; [author] 3.62538
7. [Editor < Paper]; [author] 3.25805
8. [Person < InBook]; [author] 2.92995
9. [Person < Publication]; [editor] 2.52365

(DS 1) ?x/Person, wrote, ?y/Paper

0. [Person < Paper]; [author] 13.40853
1. [Person < Article]; [author] 6.64470
2. [Person < Publication]; [author] 5.03261
3. [Author < Paper]; [author] 3.55412
4. [Person < Book]; [author] 2.72961

5. [Editor < Paper]; [author] 2.45304

6. [Person < InBook]; [author] 2.20600

7. [Author < Article]; [author] 1.76123

8. [Person < Thesis]; [author] 1.71712

9. [Person < Publication > Publication]; [author, book] 1.40254

(DS 1) ?x/Person, published, ?y/Paper

0. [Person < Paper]; [author] 7.35580

1. [Person < Article]; [author] 3.64520

2. [Person < Paper]; [editor] 2.98650

3. [Person < Publication]; [author] 2.76082

4. [Person < Publication]; [editor] 2.74687

5. [Author < Paper]; [author] 1.94972

6. [Person < Paper < Paper]; [author, cites] 1.88480

7. [Person < Paper > Paper]; [author, cites] 1.88480

8. [Person < Book]; [editor] 1.63817

9. [Person < Publication < Paper]; [editor, cites] 1.61766

(DS 1) ?x/Person, has, ?y/Paper

0. [Person < Paper]; [author] 6.81284

1. [Person < Article]; [author] 3.05186

2. [Person < Publication]; [author] 2.00170

3. [Author < Paper]; [author] 1.80580

4. [Person < Paper]; [editor] 1.60953

5. [Editor < Paper]; [author] 1.24634

6. [Person < Publication]; [editor] 1.12785

7. [Person < Book]; [author] 1.05299

8. [Author < Article]; [author] 0.89483

9. [Person < Publication > Publication]; [author, book] 0.89263

(DS 1) ?x/Person, has, ?y/Publication

0. [Person < Publication]; [author] 7.35988

1. [Person < Publication]; [editor] 4.75655

2. [Person < Paper > Journal]; [author, journal] 4.11027

3. [Person < Publication > Publication]; [author, book] 3.87704

4. [Person < Publication < Publication]; [author, book] 3.87704

5. [Person < Paper]; [author] 2.75928

6. [Person < Publication > Book < Publication]; [author, book, book] 2.51530

7. [Person < Book]; [author] 2.50495

8. [Person < Publication > Journal]; [editor, journal] 2.33975

9. [Person < Article]; [author] 2.26837

(DS 2) ?x/Paper, in, ?y/Book

0. [Paper > Journal]; [journal] 6.47502

1. [Paper > Journal < Publication]; [journal, journal] 4.82060

2. [Publication > Book]; [book] 3.49156

3. [Article > Journal]; [journal] 3.37297

4. [Paper > Journal < Article]; [journal, journal] 3.25550

5. [Paper > Author < Book]; [author, author] 3.09234

6. [Paper > Journal < Paper]; [journal, journal] 2.88172

7. [Publication < InBook]; [book] 2.78457

8. [Paper > Author < Publication]; [author, author] 2.57143

9. [Paper > Person < InBook]; [author, author] 2.53684

(DS 2) ?x/InBook, ,?y/Book

0. [InBook > Book]; [book] 12.21923

1. [InBook > Book < InBook]; [book, book] 6.69315

2. [InBook > Publication]; [book] 6.28692

3. [InBook > Book < InBook > Book]; [book, book, book] 5.37868

4. [InBook > Book < Publication]; [book, book] 4.58814

5. [InBook > Publication > Book]; [book, book] 3.69156

6. [InBook > Publication > Journal]; [book, journal] 3.17673

7. [InBook > Book < InBook > Publication]; [book, book, book] 2.77003

8. [InBook > Publication > Publication]; [book, book] 1.90264

9. [InBook > Book < Publication > Journal]; [book, book, journal] 1.82546

(DS 2) ?x/book chapter, in, ?y/book

0. [InBook > Book]; [book] 3.82014

1. [Book < InBook]; [book] 3.36104

2. [Publication > Book]; [book] 3.21454

3. [Book < InBook > Book]; [book, book] 3.11387

4. [Publication < InBook]; [book] 2.45213

5. [Book < Publication]; [book] 2.39341

6. [Publication < InBook > Book]; [book, book] 2.28423

7. [InBook > Book < InBook]; [book, book] 2.13252

8. [InBook > Publication]; [book] 2.07206

9. [Article > Journal]; [journal] 1.93734

(DS 2) ?x/InCollection, ,?y/Book

0. [InBook > Book]; [book] 8.51495
1. [InBook > Book < InBook]; [book, book] 4.95385
2. [InBook > Publication]; [book] 4.66492
3. [Publication > Book]; [book] 3.65454
4. [InBook > Book < Publication]; [book, book] 3.40442
5. [InBook > Book < InBook > Book]; [book, book, book] 2.96134
6. [Book < InBook]; [book] 2.92173
7. [Publication < InBook]; [book] 2.83269
8. [Book < InBook > Book]; [book, book] 2.67551
9. [Publication < InBook > Book]; [book, book] 2.59689

(DS 3) ?x/Conference, includes, ?y/Paper

0. [Conference < Paper]; [conference] 16.24247
1. [Conference < Publication]; [conference] 4.64567
2. [Conference < Paper]; [venue] 4.56947
3. [Conference < Publication < Paper]; [conference, proceedings] 4.19053
4. [Publication > Conference < Paper]; [conference, conference] 2.14142
5. [Conference < Publication > Publication]; [conference, book] 1.46403
6. [Conference < Publication < Publication]; [conference, book] 1.46403
7. [Conference < InProceedings > Publication]; [conference, proceedings] 1.32124
8. [Conference < Publication]; [venue] 1.30562
9. [Conference < Publication > Book]; [conference, book] 1.18580

(DS 3) ?x/Conference, published ,?y/Paper

0. [Conference < Paper]; [conference] 9.49187
1. [Conference < Publication]; [conference] 6.09890
2. [Conference < Paper > Journal < Paper]; [conference, venue, journal] 2.74378
3. [Conference < InProceedings > Venue < Paper]; [venue, conference, journal] 2.74000
4. [Conference < Publication < Paper]; [conference, proceedings] 2.72665
5. [Conference < Paper]; [venue] 2.67030
6. [Conference < Publication > Publication]; [conference, book] 2.04512
7. [Conference < Publication < Publication]; [conference, book] 2.04512
8. [Conference < Publication]; [venue] 1.71406
9. [Conference < Publication > Book]; [conference, book] 1.67358

(DS 3) ?x/Conference, has , ?y/Article

0. [Conference < Paper]; [conference] 7.66894
1. [Conference < Paper > Article]; [conference, cites] 4.04134

2. [Conference < Publication < Article]; [conference, cites] 4.00849

3. [Conference < Publication]; [conference] 3.62468

4. [Conference < Paper > Journal < Article]; [conference, venue, journal] 2.53982

5. [Conference < InProceedings > Venue < Article]; [venue, conference, journal] 2.53632

6. [Conference < Paper]; [venue] 2.15745

7. [Conference < InProceedings > Person < Article]; [conference, primaryAuthor, firstAuthor] 2.02263

8. [Conference < Publication < Paper]; [conference, proceedings] 2.01694

9. [Conference < InProceedings > Paper]; [conference, cites] 2.01400

(DS 3) ?x/Paper, in, ?y/Conference

0. [Paper > Conference]; [conference] 16.24257

1. [Publication > Conference]; [conference] 4.64577

2. [Paper > Conference]; [venue] 4.56957

3. [Paper > Publication > Conference]; [proceedings, conference] 4.19053

4. [Paper > Conference < Publication]; [conference, conference] 2.14142

5. [Publication < Publication > Conference]; [book, conference] 1.46403

6. [Publication > Publication > Conference]; [book, conference] 1.46403

7. [Publication < InProceedings > Conference]; [proceedings, conference] 1.32124

8. [Publication > Conference]; [venue] 1.30572

9. [Book < Publication > Conference]; [book, conference] 1.18580

(DS 3) ?y/Paper, published by ,?x/Conference

0. [Paper > Conference]; [conference] 9.49197

1. [Publication > Conference]; [conference] 6.09900

2. [Paper > Journal < Paper > Conference]; [journal, venue, conference] 2.74378

3. [Paper > Venue < InProceedings > Conference]; [journal, conference, venue] 2.74000

4. [Paper > Publication > Conference]; [proceedings, conference] 2.72665

5. [Paper > Conference]; [venue] 2.67040

6. [Publication < Publication > Conference]; [book, conference] 2.04512

7. [Publication > Publication > Conference]; [book, conference] 2.04512

8. [Publication > Conference]; [venue] 1.71416

9. [Book < Publication > Conference]; [book, conference] 1.67358

(DS 3) ?y/Paper, issued by ,?x/Conference

0. [Publication > Conference]; [conference] 5.00685

1. [Paper > Conference]; [conference] 1.89480

2. [Paper < InProceedings > Conference]; [cites, conference] 1.69340

3. [Paper > InProceedings > Conference]; [cites, conference] 1.68481

4. [Publication < InProceedings > Conference]; [proceedings, conference] 1.40916

5. [Publication > Conference]; [venue] 1.40721

6. [Publication > Publication > Conference]; [proceedings, conference] 1.34243

7. [Paper > Editor < Publication > Conference]; [author, editor, conference] 1.28584

8. [Paper > Publication < InProceedings > Conference]; [cites, proceedings, conference] 0.98336

9. [Article < Paper > Conference]; [cites, conference] 0.82030

(DS 3) ?y/Paper, presented by ,?x/Conference

0. [Paper > Conference]; [conference] 2.63979

1. [Paper < InProceedings > Conference]; [cites, conference] 2.60451

2. [Paper > InProceedings > Conference]; [cites, conference] 2.59129

3. [Paper > Venue < InProceedings > Conference]; [journal, conference, conference] 1.37143

4. [Paper > Conference < InProceedings > Conference]; [conference, conference, venue] 1.36802

5. [Publication > Conference]; [conference] 1.36657

6. [Article < Paper > Conference]; [cites, conference] 1.26165

7. [Article > Publication > Conference]; [cites, conference] 1.25139

8. [Publication < InProceedings > Conference]; [cites, conference] 0.97919

9. [Publication > InProceedings > Conference]; [cites, conference] 0.97275

(DS 3) ?y/InProceedings, presented by ,?x/Conference

0. [InProceedings > Conference]; [conference] 2.63979

1. [InProceedings < InProceedings > Conference]; [cites, conference] 2.51509

2. [InProceedings > InProceedings > Conference]; [cites, conference] 2.51509

3. [InProceedings > Publication > Conference]; [proceedings, conference] 1.89553

4. [InProceedings > Publication < InProceedings > Conference]; [proceedings, cites, conference] 1.69910

5. [InProceedings > Publication]; [cites] 0.49569

6. [InProceedings < Publication]; [cites] 0.49308

7. [InProceedings > Publication]; [proceedings] 0.39744

8. [InProceedings > Publication > Publication]; [proceedings, cites] 0.37637

9. [InProceedings > Publication < Publication]; [proceedings, cites] 0.37637

(DS 4) ?x/Journal, has, ?y/Paper

0. [Journal < Paper]; [journal] 15.94712

1. [Publication > Journal < Paper]; [journal, journal] 9.29575

2. [Journal < Article]; [journal] 7.52949

3. [Journal < Publication]; [journal] 4.55670

4. [Article > Journal < Paper]; [journal, journal] 4.05263

5. [Publication > Journal < Article]; [journal, journal] 3.91090

6. [Publication > Venue < Paper]; [conference, journal] 3.10103

7. [Publication > Venue < Paper]; [journal, conference] 3.09743

8. [Paper > Journal < Paper]; [journal, journal] 2.51826

9. [Publication > Journal < Publication]; [journal, journal] 2.25405

(DS 4) ?x/Journal, published ,?y/Paper

0. [Journal < Paper]; [journal] 9.31927

1. [Journal < Publication]; [journal] 5.98211

2. [Journal < Article]; [journal] 4.45319

3. [Publication > Journal < Paper]; [journal, journal] 3.27771

4. [Publication > Author < Paper]; [author, author] 2.48601

5. [Journal < Publication > Editor < Paper]; [journal, editor, author] 2.40603

6. [Journal < Paper < Paper]; [journal, cites] 2.38606

7. [Journal < Paper > Paper]; [journal, cites] 2.38606

8. [Publication > Publication]; [book] 2.28143

9. [Publication < Publication]; [book] 2.28133

(DS 4) ?y/Paper, published by ,?x/Journal

0. [Paper > Journal]; [journal] 9.31937

1. [Publication > Journal]; [journal] 5.98221

2. [Article > Journal]; [journal] 4.45329

3. [Paper > Journal < Publication]; [journal, journal] 3.27771

4. [Paper > Author < Publication]; [author, author] 2.48601

5. [Paper > Editor < Publication > Journal]; [author, editor, journal] 2.40603

6. [Paper > Paper > Journal]; [cites, journal] 2.38606

7. [Paper < Paper > Journal]; [cites, journal] 2.38606

8. [Publication > Publication]; [book] 2.28143

9. [Publication < Publication]; [book] 2.28133

(DS 4) ?x/Journal, ,?y/Article

0. [Journal < Article]; [journal] 15.94712

1. [Publication > Journal < Article]; [journal, journal] 9.29575

2. [Journal < Paper]; [journal] 7.52949

3. [Article > Journal < Article]; [journal, journal] 4.05263

4. [Journal < Article > Article]; [journal, cites] 3.96863

5. [Journal < Article < Article]; [journal, cites] 3.96863

6. [Publication > Journal < Paper]; [journal, journal] 3.91090

7. [Journal < Publication]; [journal] 3.55527

8. [Publication > Venue < Article]; [conference, journal] 3.10103

9. [Publication > Article]; [cites] 2.70312

(DS 4) ?y/Journal paper, issued by ,?x/Journal

0. [Publication > Journal]; [journal] 6.89765

1. [Journal < Publication]; [journal] 2.68944

2. [Publication > Publication]; [book] 2.63055

3. [Publication < Publication]; [book] 2.63045

4. [Publication > Book]; [book] 2.14593

5. [Book < Publication]; [book] 1.90685

6. [Publication > Publisher < Publication]; [publisher, publisher] 1.85123

7. [Paper > Venue < Publication]; [journal, conference] 1.75553

8. [Paper > Venue < Publication]; [conference, journal] 1.75350

9. [Publication < InBook]; [book] 1.61056

(DS 4) ?y/Article, issued by ,?x/Journal

0. [Publication > Journal]; [journal] 4.25531

1. [Article > Journal]; [journal] 2.39368

2. [Article < Publication]; [cites] 2.23300

3. [Article > Publication]; [cites] 2.22228

4. [Publication > Publication]; [book] 1.62285

5. [Publication < Publication]; [book] 1.62275

6. [Article > Article > Journal]; [cites, journal] 1.54223

7. [Article < Article > Journal]; [cites, journal] 1.54223

8. [Article > Editor < Publication > Journal]; [author, editor, journal] 1.45454

9. [Article > Editor < Publication]; [author, editor] 1.39454

(DS 5) ?x/InProceedings, in, ?y/Proceedings

0. [InProceedings > Proceedings]; [proceedings] 16.16038

1. [InProceedings > Proceedings < InProceedings]; [proceedings, proceedings] 11.29314

2. [InProceedings > InProceedings]; [proceedings] 5.71896

3. [InProceedings < InProceedings]; [proceedings] 5.71886

4. [InProceedings < InProceedings > Proceedings]; [proceedings, proceedings] 3.62537

5. [InProceedings > Thing]; [proceedings] 1.53675

6. [InProceedings > Proceedings < Thing]; [proceedings, proceedings] 1.43185

7. [InProceedings < Thing]; [proceedings] 0.72500

8. [InProceedings > Thing]; [conference] 0.42965

9. [InProceedings > Publication > Thing]; [proceedings, conference] 0.39849

(DS 5) ?x/Paper, in, ?y/Proceedings

0. [Paper > Proceedings]; [proceedings] 16.16038

1. [Paper > Proceedings < InProceedings]; [proceedings, proceedings] 11.29314

2. [Paper > InProceedings]; [proceedings] 5.71896
3. [Paper < InProceedings]; [proceedings] 5.71886
4. [Publication > Proceedings]; [proceedings] 4.61771
5. [Publication < InProceedings]; [proceedings] 3.68003
6. [Paper < InProceedings > Proceedings]; [proceedings, proceedings] 3.62537
7. [Publication > Proceedings < InProceedings]; [proceedings, proceedings] 2.11888
8. [Publication > InProceedings]; [proceedings] 1.73644
9. [Publication < InProceedings > Proceedings]; [proceedings, proceedings] 1.71125

(DS 5) ?x/Paper, published in, ?y/Proceedings

0. [Paper > Proceedings]; [proceedings] 9.44393
1. [Publication > Proceedings]; [proceedings] 6.06217
2. [Publication < InProceedings]; [proceedings] 4.54660
3. [Paper > Editor < Proceedings]; [author, editor] 3.46840
4. [Paper > InProceedings]; [proceedings] 3.34209
5. [Paper < InProceedings]; [proceedings] 3.34199
6. [Paper > Editor < Proceedings < InProceedings]; [author, editor, proceedings] 2.84576
7. [Paper > Person < InProceedings]; [author, author] 2.77988
8. [Paper > Venue < InProceedings > Proceedings]; [journal, conference, proceedings] 2.69677
9. [Paper > Editor < Publication > Proceedings]; [author, editor, proceedings] 2.48674

(DS 5) ?x/InProceedings, published in, ?y/Proceedings

0. [InProceedings > Editor < Proceedings]; [author, editor] 3.33980
1. [InProceedings > Editor < Proceedings < InProceedings]; [author, editor, proceedings] 2.72864
2. [InProceedings > Person < InProceedings]; [author, author] 2.68632
3. [InProceedings < InProceedings > Proceedings]; [cites, proceedings] 2.35622
4. [InProceedings > InProceedings > Proceedings]; [cites, proceedings] 2.35622
5. [InProceedings > Editor < Publication > Proceedings]; [author, editor, proceedings] 2.33058
6. [InProceedings > InProceedings]; [cites] 2.32805
7. [InProceedings < InProceedings]; [cites] 2.32795
8. [InProceedings > Proceedings > Editor < InProceedings]; [proceedings, editor, author] 2.25244
9. [InProceedings > Venue < Paper > Proceedings]; [conference, journal, proceedings] 2.10067

(DS 6) ?x/Paper, ,?y/Subject

0. [Paper > Ŝubject]; [@subject] 16.63370
1. [Article > Ŝubject]; [@subject] 8.22375
2. [Publication > Ŝubject]; [@subject] 6.24386
3. [Paper > Author > Ŝubject]; [author, @subject] 4.23903
4. [Paper > Venue > Ŝubject]; [journal, @subject] 3.35252

5. [InBook > Ŝubject]; [@subject] 2.74067

6. [Thesis > Ŝubject]; [@subject] 2.17790

7. [Article > Author > Ŝubject]; [author, @subject] 2.05712

8. [Author > Ŝubject]; [@subject] 1.92450

9. [Paper > T̂ext]; [@subject] 1.82512

(DS 6) ?x/Paper, ,?y/Field

0. [Paper > Ŝubject]; [@subject] 5.37642

1. [Article > Ŝubject]; [@subject] 2.65812

2. [Publication > Ŝubject]; [@subject] 2.01817

3. [Paper > Venue]; [venue] 1.94821

4. [Paper > Author > Ŝubject]; [author, @subject] 1.50099

5. [Paper > Venue > Ŝubject]; [venue, @subject] 1.47607

6. [Paper > Venue]; [journal] 1.29364

7. [Paper > Venue > Ŝubject]; [journal, @subject] 1.18709

8. [Paper > Journal]; [venue] 1.16253

9. [Paper > Venue]; [conference] 1.01246

(DS 6) ?x/Paper, ,?y/field of study

0. [Paper > Ŝubject]; [@subject] 5.38229

1. [Article > Ŝubject]; [@subject] 2.66102

2. [Publication > Ŝubject]; [@subject] 2.02037

3. [Paper > Author > Ŝubject]; [author, @subject] 1.51065

4. [Paper > Venue > Ŝubject]; [journal, @subject] 1.19472

5. [Paper > Venue > Ŝubject]; [venue, @subject] 1.18016

6. [Paper > Venue]; [venue] 1.09587

7. [InBook > Ŝubject]; [@subject] 0.88682

8. [Paper > T̂ext]; [@subject] 0.86763

9. [Paper > Venue]; [journal] 0.84428

(DS 6) ?x/Paper, ,?y/research area

0. [Paper > Ŝubject]; [@subject] 0.22640

1. [Article > Ŝubject]; [@subject] 0.11193

2. [Paper > Institution > Country]; [institution, country] 0.10359

3. [Publication > Ŝubject]; [@subject] 0.08499

4. [Paper > Author > Ŝubject]; [author, @subject] 0.05414

5. [Paper > Author > Institution > Country]; [author, institution, country] 0.05003

6. [Paper > Venue > Ŝubject]; [journal, @subject] 0.04895

7. [Article > Institution > Country]; [institution, country] 0.04629

8. [InBook > Ŝubject]; [@subject] 0.03730

9. [Publication > Institution > Country]; [institution, country] 0.03685

(DS 6) ?x/Paper, ,?y/topic

0. [Paper > Ŝubject]; [@subject] 12.34700

1. [Article > Ŝubject]; [@subject] 6.10440

2. [Publication > Ŝubject]; [@subject] 4.63474

3. [Paper > Author > Ŝubject]; [author, @subject] 3.14658

4. [Paper > Conference > Ŝubject]; [conference, @subject] 2.53036

5. [Paper > Venue > Ŝubject]; [journal, @subject] 2.48854

6. [InBook > Ŝubject]; [@subject] 2.03436

7. [Paper > T̂ext]; [@subject] 1.91736

8. [Thesis > Ŝubject]; [@subject] 1.61663

9. [Article > Author > Ŝubject]; [author, @subject] 1.52698

(DS 7) ?x/Conference, ,?y/Subject

0. [Conference < InProceedings > Ŝubject]; [conference, @subject] 12.18665

1. [Conference > Ŝubject]; [@subject] 10.57531

2. [Conference < InProceedings > Ŝubject]; [venue, @subject] 4.41836

3. [Conference < InProceedings > Publication > Ŝubject]; [conference, proceedings, @subject] 3.46512

4. [Conference < Publication < InProceedings > Ŝubject]; [conference, proceedings, @subject] 3.24167

5. [Publication > Ŝubject]; [@subject] 2.43590

6. [Journal > Ŝubject]; [@subject] 1.55055

7. [Article > Ŝubject]; [@subject] 1.46261

8. [Conference < InProceedings > T̂ext]; [conference, @subject] 1.33717

9. [Publication > Conference > Ŝubject]; [conference, @subject] 1.31204

(DS 7) ?x/Conference, ,?y/Field

0. [Conference < InProceedings > Ŝubject]; [conference, @subject] 3.93902

1. [Conference > Ŝubject]; [@subject] 3.41820

2. [Conference < InProceedings > Ŝubject]; [venue, @subject] 1.95453

3. [Conference < Paper > Venue]; [conference, venue] 1.52772

4. [Conference < InProceedings > Publication > Ŝubject]; [conference, proceedings, @subject] 1.30387

5. [Conference < Publication < InProceedings > Ŝubject]; [conference, proceedings, @subject] 1.21979

6. [Conference < InProceedings > Venue]; [conference, conference] 1.20052

7. [Conference < Paper > Venue]; [conference, journal] 0.96173

8. [Conference < Paper > Journal]; [conference, venue] 0.87274

9. [Conference < Paper > Venue]; [venue, venue] 0.78953

(DS 7) ?x/Conference, ,?y/Field of study

0. [Conference < InProceedings > Ŝubject]; [conference, @subject] 3.94333

1. [Conference > Ŝubject]; [@subject] 3.42193

2. [Conference < InProceedings > Ŝubject]; [venue, @subject] 1.57455

3. [Conference < InProceedings > Publication > Ŝubject]; [conference, proceedings, @subject] 1.31692

4. [Conference < Publication < InProceedings > Ŝubject]; [conference, proceedings, @subject] 1.23200

5. [Conference < Paper > Venue]; [conference, venue] 0.86103

6. [Publication > Ŝubject]; [@subject] 0.78820

7. [Conference < InProceedings > Venue]; [conference, conference] 0.67529

8. [Conference < InProceedings > T̂ext]; [conference, @subject] 0.63567

9. [Conference < Paper > Venue]; [conference, journal] 0.62491

(DS 7) ?x/Conference, ,?y/research area

0. [Conference > Ŝubject]; [@subject] 0.14394

1. [Conference < InProceedings > Institution > Country]; [conference, institution, country] 0.09532

2. [Conference < InProceedings > Ŝubject]; [conference, @subject] 0.05611

3. [Conference < InProceedings > Institution > Country]; [venue, institution, country] 0.04093

4. [Conference < Paper > Venue > Ŝubject]; [conference, journal, @subject] 0.03704

5. [Publication > Ŝubject]; [@subject] 0.03316

6. [Journal > Ŝubject]; [@subject] 0.02110

7. [Article > Ŝubject]; [@subject] 0.01991

8. [Publication > Venue < Paper > Ŝubject]; [conference, journal, @subject] 0.01355

9. [Publication > Institution > Country]; [institution, country] 0.01312

(DS 7) ?x/Conference, ,?y/topic

0. [Conference < InProceedings > Ŝubject]; [conference, @subject] 9.04600

1. [Conference > Ŝubject]; [@subject] 7.84993

2. [Conference < InProceedings > Ŝubject]; [venue, @subject] 3.27969

3. [Conference < InProceedings > Publication > Ŝubject]; [conference, proceedings, @subject] 2.57212

4. [Conference < Publication < InProceedings > Ŝubject]; [conference, proceedings, @subject] 2.40626

5. [Publication > Ŝubject]; [@subject] 1.80814

6. [Conference < InProceedings > T̂ext]; [conference, @subject] 1.40475

7. [Conference > T̂ext]; [@subject] 1.21901

8. [Journal > Ŝubject]; [@subject] 1.15095

9. [Article > Ŝubject]; [@subject] 1.08568

(DS 8) ?x/Venue, ,?y/Subject

0. [Venue < Paper > Ŝubject]; [venue, @subject] 12.72165

1. [Venue > Ŝubject]; [@subject] 10.82357

2. [Venue < InProceedings > Ŝubject]; [conference, @subject] 4.41836

3. [Venue < Paper > Person > Ŝubject]; [venue, author, @subject] 3.15859

4. [Venue < Paper > T̂ext]; [venue, @subject] 1.39587

5. [Publication > Ŝubject]; [@subject] 1.36221

6. [Venue < Paper > Person]; [venue, author] 1.29793

7. [Venue > T̂ext]; [@subject] 1.18761

8. [Journal < Paper > Ŝubject]; [venue, @subject] 1.07419

9. [Venue < Paper > Thing]; [venue, @subject] 0.99135

(DS 8) ?x/Venue, ,?y/Field

0. [Venue < Paper > Ŝubject]; [venue, @subject] 4.11195

1. [Venue > Ŝubject]; [@subject] 3.49845

2. [Venue < Paper > Venue]; [venue, venue] 1.82289

3. [Venue < InProceedings > Ŝubject]; [conference, @subject] 1.56449

4. [Venue < Paper > Ŝubject]; [journal, @subject] 1.30996

5. [Venue < Publication]; [venue] 0.91903

6. [Venue < Paper > Venue]; [conference, venue] 0.87019

7. [Venue < Article > Journal]; [venue, venue] 0.85312

8. [Venue < Paper > Venue]; [venue, conference] 0.76597

9. [Venue < Article > Venue]; [venue, journal] 0.73061

(DS 8) ?x/Venue, ,?y/Field of study

0. [Venue < Paper > Ŝubject]; [venue, @subject] 4.11644

1. [Venue > Ŝubject]; [@subject] 3.50227

2. [Venue < InProceedings > Ŝubject]; [conference, @subject] 1.57455

3. [Venue < Paper > Ŝubject]; [journal, @subject] 1.15628

4. [Venue < Paper > Venue]; [venue, venue] 1.00523

5. [Venue < Paper > Person > Ŝubject]; [venue, author, @subject] 0.88491

6. [Venue < Publication]; [venue] 0.83125

7. [Venue < Paper > T̂ext]; [venue, @subject] 0.66358

8. [Venue < Article > Journal]; [venue, venue] 0.56872

9. [Venue > T̂ext]; [@subject] 0.56457

(DS 8) ?x/Venue, ,?y/research area

0. [Venue < Paper > Ŝubject]; [venue, @subject] 0.17316

1. [Venue > Ŝubject]; [@subject] 0.14732

2. [Venue < Paper > Institution > Country]; [venue, institution, country] 0.12135

3. [Venue < Paper > Ŝubject]; [journal, @subject] 0.06957

4. [Venue < Paper > Person > Ŝubject]; [venue, author, @subject] 0.06480

5. [Venue < InProceedings > Ŝubject]; [conference, @subject] 0.05611
6. [Venue < InProceedings > Institution > Country]; [conference, institution, country] 0.05007
7. [Institution > Country]; [country] 0.03810
8. [Venue < Paper > Institution > Country]; [journal, institution, country] 0.02887
9. [Publication > Ŝubject]; [@subject] 0.01854

(DS 8) ?x/Venue, ,?y/topic

0. [Venue < Paper > Ŝubject]; [venue, @subject] 9.44313
1. [Venue > Ŝubject]; [@subject] 8.03421
2. [Venue < InProceedings > Ŝubject]; [conference, @subject] 3.44571
3. [Venue < Paper > Ŝubject]; [journal, @subject] 2.42182
4. [Venue < Paper > Person > Ŝubject]; [venue, author, @subject] 1.93966
5. [Venue < Paper > T̂ext]; [venue, @subject] 1.46642
6. [Venue > T̂ext]; [@subject] 1.24763
7. [Publication > Ŝubject]; [@subject] 1.01116
8. [Venue < Article]; [venue] 0.84058
9. [Journal < Paper > Ŝubject]; [venue, @subject] 0.79736

(DS 9) ?x/Journal, ,?y/Subject

0. [Journal < Paper > Ŝubject]; [journal, @subject] 11.86996
1. [Publication > Ŝubject]; [@subject] 10.79355
2. [Journal > Ŝubject]; [@subject] 9.30876
3. [Publication > Venue > Ŝubject]; [journal, @subject] 5.41581
4. [InBook > Ŝubject]; [@subject] 4.72086
5. [Article > Ŝubject]; [@subject] 4.45113
6. [Publication < InBook > Ŝubject]; [book, @subject] 4.38102
7. [Publication > Publication > Ŝubject]; [book, @subject] 4.25701
8. [Journal < Paper > Person > Ŝubject]; [journal, author, @subject] 3.72347
9. [Book < InBook > Ŝubject]; [book, @subject] 3.57267

(DS 9) ?x/Journal, ,?y/Field

0. [Journal < Paper > Ŝubject]; [journal, @subject] 3.83666
1. [Publication > Ŝubject]; [@subject] 3.48874
2. [Journal > Ŝubject]; [@subject] 3.00882
3. [Publication > Venue]; [journal] 2.17545
4. [Journal < Paper > Ŝubject]; [venue, @subject] 1.90374
5. [Publication > Venue > Ŝubject]; [journal, @subject] 1.75052
6. [Publication < InBook > Ŝubject]; [book, @subject] 1.55127
7. [InBook > Ŝubject]; [@subject] 1.52590

8. [Publication > Publication > Ŝubject]; [book, @subject] 1.50735
9. [Journal < Article > Venue]; [journal, venue] 1.49207

(DS 9) ?x/Journal, ,?y/Field of study

0. [Journal < Paper > Ŝubject]; [journal, @subject] 3.84085
1. [Publication > Ŝubject]; [@subject] 3.49255
2. [Journal > Ŝubject]; [@subject] 3.01211
3. [Publication > Venue > Ŝubject]; [journal, @subject] 1.75243
4. [Publication < InBook > Ŝubject]; [book, @subject] 1.56125
5. [InBook > Ŝubject]; [@subject] 1.52756
6. [Journal < Paper > Ŝubject]; [venue, @subject] 1.52210
7. [Publication > Publication > Ŝubject]; [book, @subject] 1.51705
8. [Article > Ŝubject]; [@subject] 1.44029
9. [Journal < Paper > Person > Ŝubject]; [journal, author, @subject] 1.41510

(DS 9) ?x/Journal, ,?y/research area

0. [Publication > Ŝubject]; [@subject] 0.14691
1. [Journal > Ŝubject]; [@subject] 0.12670
2. [Journal < Paper > Ŝubject]; [journal, @subject] 0.06957
3. [Publication > Institution > Country]; [institution, country] 0.06853
4. [InBook > Ŝubject]; [@subject] 0.06426
5. [Article > Ŝubject]; [@subject] 0.06058
6. [Book < InBook > Ŝubject]; [book, @subject] 0.06013
7. [Journal < Paper > Institution > Country]; [journal, institution, country] 0.05351
8. [Institution > Country]; [country] 0.04877
9. [InBook > Publication > Ŝubject]; [book, @subject] 0.04387

(DS 9) ?x/Journal, ,?y/topic

0. [Journal < Paper > Ŝubject]; [journal, @subject] 8.81093
1. [Publication > Ŝubject]; [@subject] 8.01193
2. [Journal > Ŝubject]; [@subject] 6.90978
3. [Publication > Venue > Ŝubject]; [journal, @subject] 4.02009
4. [InBook > Ŝubject]; [@subject] 3.50424
5. [Article > Ŝubject]; [@subject] 3.30402
6. [Publication < InBook > Ŝubject]; [book, @subject] 3.25198
7. [Publication > Publication > Ŝubject]; [book, @subject] 3.15993
8. [Journal < Paper > Person > Ŝubject]; [journal, author, @subject] 2.76389
9. [Book < InBook > Ŝubject]; [book, @subject] 2.65195

(DS 10) ?x/Author, ,?y/research area

0. [Author > Ŝubject]; [@subject] 0.20292

1. [Author < Paper > Institution > Country]; [author, institution, country] 0.10937

2. [Author > Institution > Country]; [institution, country] 0.07624

3. [Author < Paper > Author > Ŝubject]; [primaryAuthor, author, @subject] 0.05829

4. [Author < Paper > Ŝubject]; [author, @subject] 0.05778

5. [Person > Ŝubject]; [@subject] 0.05379

6. [Author < Paper > Venue > Ŝubject]; [author, journal, @subject] 0.05198

7. [Author < InProceedings > Conference > Ŝubject]; [author, conference, @subject] 0.04746

8. [Person < Paper > Ŝubject]; [firstAuthor, @subject] 0.04589

9. [Article > Ŝubject]; [@subject] 0.04577

(DS 10) ?x/Author, ,?y/field

0. [Author > Ŝubject]; [@subject] 4.81884

1. [Author < Paper > Ŝubject]; [author, @subject] 4.43835

2. [Author < Publication > Ŝubject]; [editor, @subject] 1.85643

3. [Author < Paper > Venue]; [author, venue] 1.72176

4. [Person > Ŝubject]; [@subject] 1.27733

5. [Author < Publication < InBook > Ŝubject]; [author, book, @subject] 1.26277

6. [Author < Paper < Paper > Ŝubject]; [author, cites, @subject] 1.24702

7. [Person < Paper > Ŝubject]; [author, @subject] 1.17647

8. [Author < Paper > Venue]; [author, journal] 1.15202

9. [Article > Ŝubject]; [@subject] 1.08684

(DS 10) ?x/Author, ,?y/subject

0. [Author > Ŝubject]; [@subject] 14.90865

1. [Author < Paper > Ŝubject]; [author, @subject] 13.73148

2. [Author < Publication > Ŝubject]; [editor, @subject] 5.24286

3. [Person > Ŝubject]; [@subject] 3.95182

4. [Person < Paper > Ŝubject]; [author, @subject] 3.63979

5. [Article > Ŝubject]; [@subject] 3.36250

6. [Author < Publication < InBook > Ŝubject]; [author, book, @subject] 3.35589

7. [Author < Paper < Paper > Ŝubject]; [author, cites, @subject] 3.31405

8. [Editor < Paper > Ŝubject]; [author, @subject] 3.11917

9. [Editor > Ŝubject]; [@subject] 3.11657

(DS 10) ?x/Scholar, ,?y/research area

0. [Author > Ŝubject]; [@subject] 0.06047

1. [Author > Institution > Country]; [institution, country] 0.05269

2. [Institution > Country]; [country] 0.04225

3. [Author < Paper > Institution > Country]; [author, institution, country] 0.03402

4. [Editor > Ŝubject]; [@subject] 0.02665

5. [Publication > Ŝubject]; [@subject] 0.02432

6. [Editor > Institution > Country]; [institution, country] 0.02199

7. [Journal > Ŝubject]; [@subject] 0.02136

8. [Publication > Institution > Country]; [institution, country] 0.02113

9. [Author < Paper > Ŝubject]; [author, @subject] 0.01722

(DS 10) ?x/Scholar, ,?y/field

0. [Author > Ŝubject]; [@subject] 1.43592

1. [Author < Paper > Ŝubject]; [author, @subject] 0.65757

2. [Editor > Ŝubject]; [@subject] 0.63288

3. [Publication > Ŝubject]; [@subject] 0.57750

4. [Journal > Ŝubject]; [@subject] 0.50718

5. [Journal < Paper > Ŝubject]; [venue, @subject] 0.32090

6. [Editor < Paper > Ŝubject]; [author, @subject] 0.31493

7. [Author < Publication > Ŝubject]; [editor, @subject] 0.30770

8. [Author < Paper > Venue]; [author, venue] 0.26568

9. [Journal < Paper > Ŝubject]; [journal, @subject] 0.24096

(DS 10) x/Scholar, ,?y/subject

0. [Author > Ŝubject]; [@subject] 4.44247

1. [Editor > Ŝubject]; [@subject] 1.95802

2. [Author < Paper > Ŝubject]; [author, @subject] 1.85708

3. [Publication > Ŝubject]; [@subject] 1.78669

4. [Journal > Ŝubject]; [@subject] 1.56911

5. [Editor < Paper > Ŝubject]; [author, @subject] 0.88942

6. [Journal < Paper > Ŝubject]; [journal, @subject] 0.68051

7. [Publication > Author > Ŝubject]; [author, @subject] 0.62733

8. [Author > T̂ext]; [@subject] 0.48745

9. [Editor < Publication > Ŝubject]; [editor, @subject] 0.47470

(DS 10) ?x/Person, ,?y/research area

0. [Person > Ŝubject]; [@subject] 0.20292

1. [Person < Paper > Ŝubject]; [firstAuthor, @subject] 0.09664

2. [Person < Paper > Ŝubject]; [secondAuthor, @subject] 0.09547

3. [Person > Institution > Country]; [institution, country] 0.07624

4. [Person < Publication > Ŝubject]; [editor, @subject] 0.07059

5. [Person < Paper > Ŝubject]; [author, @subject] 0.05778

6. [Person < Paper > Institution > Country]; [author, institution, country] 0.05764

7. [Author > Ŝubject]; [@subject] 0.05379

8. [Editor > Ŝubject]; [@subject] 0.03310

9. [Author > Institution > Country]; [institution, country] 0.02021

(DS 10) ?x/Person, ,?y/field

0. [Person > Ŝubject]; [@subject] 4.81884

1. [Person < Paper > Ŝubject]; [author, @subject] 2.25418

2. [Author > Ŝubject]; [@subject] 1.27733

3. [Person < Publication > Ŝubject]; [editor, @subject] 1.21161

4. [Person < Paper > Venue]; [author, venue] 0.91078

5. [Person > T̂ext]; [@subject] 0.80021

6. [Editor > Ŝubject]; [@subject] 0.78603

7. [Author < Paper > Ŝubject]; [author, @subject] 0.59751

8. [Person < Paper > Venue]; [author, journal] 0.58830

9. [Person < Paper > Journal]; [author, venue] 0.53742

(DS 10) ?x/Person, ,?y/field of study

0. [Person > Ŝubject]; [@subject] 4.82410

1. [Person < Paper > Ŝubject]; [author, @subject] 2.26869

2. [Author > Ŝubject]; [@subject] 1.27872

3. [Person < Publication > Ŝubject]; [editor, @subject] 1.21941

4. [Editor > Ŝubject]; [@subject] 0.78689

5. [Person > T̂ext]; [@subject] 0.77765

6. [Author < Paper > Ŝubject]; [author, @subject] 0.60136

7. [Person < Paper > Venue]; [author, venue] 0.54266

8. [Editor < Paper > Ŝubject]; [author, @subject] 0.40212

9. [Person < Publication]; [author] 0.39038

(DS 10) ?x/Person, ,?y/subject

0. [Person > Ŝubject]; [@subject] 14.90865

1. [Person < Paper > Ŝubject]; [author, @subject] 6.36617

2. [Author > Ŝubject]; [@subject] 3.95182

3. [Person < Publication > Ŝubject]; [editor, @subject] 3.42179

4. [Editor > Ŝubject]; [@subject] 2.43184

5. [Author < Paper > Ŝubject]; [author, @subject] 1.68748

6. [Person > T̂ext]; [@subject] 1.63584

7. [Person > Thing]; [@subject] 1.16177

8. [Editor < Paper > Ŝubject]; [author, @subject] 1.12839

9. [Author < Publication > Ŝubject]; [editor, @subject] 0.90701

(DS 11) ?x/Person, published, ?y/Book

0. [Person < Book]; [editor] 5.33040

1. [Person < Book]; [author] 4.87236

2. [Person < InBook]; [author] 3.93774

3. [Person < Publication]; [author] 3.78384

4. [Person < Publication]; [editor] 3.76473

5. [Person < Publication > Book]; [author, book] 3.46006

6. [Person < Publication < InBook]; [author, book] 2.59761

7. [Person < Publication < InBook > Book]; [author, book, book] 2.52465

8. [Person < Article]; [author] 2.42303

9. [Person < Paper]; [author] 2.26066

(DS 11) ?x/Author, published, ?y/Book

0. [Author < Publication]; [author] 9.16089

1. [Author < Book]; [author] 8.31866

2. [Author < Publication > Book]; [author, book] 6.54552

3. [Author < Book]; [editor] 5.33040

4. [Author < InBook]; [author] 5.04222

5. [Author < Publication < InBook]; [author, book] 4.91400

6. [Author < Publication < InBook > Book]; [author, book, book] 4.05170

7. [Author < Publication]; [editor] 3.76473

8. [Author < Paper > Journal]; [author, journal] 3.59983

9. [Author < Publication > Publication]; [author, book] 3.36854

(DS 11) ?x/Person , wrote, ?y/Book

0. [Person < Book]; [author] 8.88163

1. [Person < InBook]; [author] 7.17797

2. [Person < Publication > Book]; [author, book] 6.98848

3. [Person < Publication]; [author] 6.89743

4. [Person < Publication < InBook]; [author, book] 5.24655

5. [Person < Paper > Journal]; [author, journal] 4.45089

6. [Person < Article]; [author] 4.41689

7. [Person < Paper]; [author] 4.12090

8. [Author < Book]; [author] 2.35417

9. [Person < Book]; [editor] 2.14536

(DS 11) ?x/Author, book, ?y/Book

0. [Author < Book]; [author] 11.79621
1. [Author < Publication > Book]; [author, book] 9.28179
2. [Author < Publication]; [author] 7.43847
3. [Author < InBook]; [author] 7.15009
4. [Author < Publication < InBook]; [author, book] 6.96823
5. [Author < Publication < InBook > Book]; [author, book, book] 5.74545
6. [Author < Paper > Journal]; [author, journal] 5.29664
7. [Author < Book]; [editor] 4.89724
8. [Author < Publication > Publication]; [author, book] 4.77671
9. [Author < Publication < Publication]; [author, book] 4.77671

(DS 11) ?x/Scholar, published, ?y/Book

0. [Author < Book]; [editor] 1.58828
1. [Author < Book]; [author] 1.45179
2. [Publication > Book]; [book] 1.31165
3. [Author < InBook]; [author] 1.17330
4. [Author < Publication]; [author] 1.12744
5. [Author < Publication]; [editor] 1.12174
6. [Author < Publication > Book]; [author, book] 1.03103
7. [Publication < InBook]; [book] 0.98437
8. [Publisher < Book]; [publisher] 0.90235
9. [Editor < Book]; [editor] 0.86758

(DS 11) ?x/Researcher, published, ?y/Book

0. [Author < Book]; [editor] 1.36166
1. [Author < Book]; [author] 1.24465
2. [Publisher < Book]; [publisher] 1.04931
3. [Author < InBook]; [author] 1.00589
4. [Author < Publication]; [author] 0.96657
5. [Author < Publication]; [editor] 0.96169
6. [Publication > Book]; [book] 0.92443
7. [Author < Publication > Book]; [author, book] 0.88393
8. [Editor < Book]; [editor] 0.79421
9. [Publication < InBook]; [book] 0.69374

(DS 12) ?x/Book, published by, ?y/Publisher

0. [Book > Publisher]; [publisher] 8.66628
1. [Publication > Publisher]; [publisher] 6.42719

2. [InBook > Book > Publisher]; [book, publisher] 4.62402
3. [Book < Publication > Publisher]; [book, publisher] 3.00898
4. [Book < Publication]; [book] 2.95587
5. [Publication > Book > Publisher]; [book, publisher] 2.94509
6. [Journal < Publication > Publisher]; [journal, publisher] 2.64153
7. [InBook > Publication]; [book] 2.21864
8. [Journal < Publication]; [journal] 1.56628
9. [Publication > Publication]; [book] 1.52088

(DS 12) ?x/Book, has, ?y/Publisher

0. [Book > Publisher]; [publisher] 11.41806
1. [InBook > Book > Publisher]; [book, publisher] 6.09227
2. [Publication > Publisher]; [publisher] 5.84005
3. [Book < InBook > Book > Publisher]; [book, book, publisher] 4.57751
4. [Publication > Book > Publisher]; [book, publisher] 4.17624
5. [Book < Publication > Publisher]; [book, publisher] 3.96441
6. [Journal < Publication > Publisher]; [journal, publisher] 3.21344
7. [Book > Publisher < Publication]; [publisher, publisher] 2.53208
8. [Book < Publication]; [book] 2.41581
9. [Publication < InBook > Book > Publisher]; [book, book, publisher] 2.35813

(DS 12) ?x/Book, publisher, ?y/Firm

0. [Book > Publisher]; [publisher] 4.77175
1. [Publication > Publisher]; [publisher] 2.68600
2. [InBook > Book > Publisher]; [book, publisher] 2.54604
3. [Publication > Book > Publisher]; [book, publisher] 1.74530
4. [Book < Publication > Publisher]; [book, publisher] 1.65678
5. [Journal < Publication > Publisher]; [journal, publisher] 1.37641
6. [Publication < Publication > Publisher]; [book, publisher] 0.85376
7. [Thing > Publisher]; [publisher] 0.50947
8. [Author < Publication > Publisher]; [author, publisher] 0.46659
9. [Author < Proceedings > Publisher]; [editor, publisher] 0.38101

(DS 12) ?x/Book, publisher, ?y/Company

0. [Book > Publisher]; [publisher] 3.23907
1. [Publication > Publisher]; [publisher] 1.82327
2. [InBook > Book > Publisher]; [book, publisher] 1.72826
3. [Publication > Institution]; [institution] 1.57119
4. [Publication > Book > Publisher]; [book, publisher] 1.18472

5. [Book < Publication > Institution]; [book, institution] 1.17393

6. [Book < Publication > Publisher]; [book, publisher] 1.12462

7. [Publication > Person > Institution]; [author, institution] 0.94465

8. [Journal < Publication > Publisher]; [journal, publisher] 0.93431

9. [Book > Author > Institution]; [author, institution] 0.91274

(DS 12) ?x/Company, published, ?y/Book

0. [Publisher < Book]; [publisher] 2.45835

1. [Institution < Publication]; [institution] 2.06999

2. [Institution < Publication > Book]; [institution, book] 1.46369

3. [Publisher < Publication]; [publisher] 1.38375

4. [Publisher < Book < InBook]; [publisher, book] 1.31174

5. [Institution < Publication < InBook]; [institution, book] 1.09931

6. [Institution < Book]; [institution] 1.03815

7. [Institution < Paper > Journal]; [institution, journal] 0.95828

8. [Institution < InBook]; [institution] 0.93708

9. [Institution < Person < Publication]; [institution, author] 0.89940

(DS 13) ?x/Person, second author of, ?y/Article

0. [Person < Article]; [secondAuthor] 15.66614

1. [Person < Article]; [firstAuthor] 13.69085

2. [Person < Article]; [author] 12.83388

3. [Person < Paper]; [secondAuthor] 8.66053

4. [Person < Paper]; [firstAuthor] 7.53387

5. [Person < Paper]; [author] 7.06542

6. [Person < Publication]; [secondAuthor] 5.39196

7. [Person < Publication]; [firstAuthor] 4.69275

8. [Person < Publication]; [author] 4.39922

9. [Author < Article]; [secondAuthor] 4.15254

(DS 13) ?y/Article, second, ?x/Author

0. [Article > Author]; [secondAuthor] 11.89969

1. [Article > Author]; [firstAuthor] 8.73228

2. [Paper > Author]; [secondAuthor] 6.57840

3. [Article > Person < Paper > Author]; [firstAuthor, secondAuthor, author] 5.44506

4. [Paper > Author]; [firstAuthor] 4.80527

5. [Publication > Author]; [secondAuthor] 4.09567

6. [Article > Editor < Publication > Author]; [secondAuthor, secondEditor, author] 3.29661

7. [Article > Person]; [secondAuthor] 3.15424

8. [Paper > Person < Paper > Author]; [firstAuthor, secondAuthor, author] 3.09254

9. [Publication > Author]; [firstAuthor] 2.99316

(DS 14) ?x/Conference proceedings, first editor, ?y/Person

0. [Proceedings > Person]; [firstEditor] 9.75422

1. [Proceedings > Person]; [secondEditor] 8.25185

2. [Proceedings > Person]; [editor] 8.00969

3. [InProceedings > Person]; [firstAuthor] 7.02187

4. [Proceedings < Paper > Person]; [proceedings, firstAuthor] 6.86518

5. [Proceedings < InProceedings > Person]; [proceedings, secondAuthor] 5.73910

6. [InProceedings > Person]; [secondAuthor] 5.65508

7. [InProceedings > Proceedings > Person]; [proceedings, firstEditor] 5.24535

8. [InProceedings > Publication > Person]; [proceedings, firstAuthor] 4.75971

9. [InProceedings > Proceedings > Person]; [proceedings, editor] 4.59345

(DS 14) ?x/proceedings, first, ?y/Editor

0. [Proceedings > Editor]; [firstEditor] 8.98795

1. [InProceedings > Editor]; [firstAuthor] 8.32332

2. [Proceedings < InProceedings > Editor]; [proceedings, firstAuthor] 8.01352

3. [Proceedings > Editor]; [secondEditor] 6.37030

4. [InProceedings > Editor]; [secondAuthor] 6.12487

5. [InProceedings > Proceedings > Editor]; [proceedings, firstEditor] 6.11634

6. [InProceedings > Publication > Editor]; [proceedings, firstAuthor] 5.96880

7. [Proceedings < InProceedings > Editor]; [proceedings, secondAuthor] 5.92957

8. [Proceedings < Publication > Editor]; [proceedings, firstEditor] 5.54911

9. [InProceedings > Publication > Editor]; [proceedings, secondAuthor] 4.37172

(DS 15) ?x/Person, has ,?y/email

0. [Person > Ê-mail]; [@e-mail] 6.23443

1. [Author > Ê-mail]; [@e-mail] 1.65256

2. [Editor > Ê-mail]; [@e-mail] 1.05613

3. [Publisher < Publication > Person > Ê-mail]; [publisher, author, @e-mail] 0.09151

(DS 15) ?x/Person, has ,?y/e-mail

0. [Person > Ê-mail]; [@e-mail] 14.02746

1. [Author > Ê-mail]; [@e-mail] 3.71825

2. [Editor > Ê-mail]; [@e-mail] 2.37629

3. [Person > Ĥomepage]; [@homepage] 0.83321

4. [Author > Ĥomepage]; [@homepage] 0.22086

5. [Publisher < Publication > Person > Ê-mail]; [publisher, author, @e-mail] 0.17987

6. [Person < Article]; [author] 0.15591

7. [Editor > Ĥomepage]; [@homepage] 0.15055

8. [Author < Article]; [author] 0.08327

9. [Person < Paper > Article]; [author, cites] 0.07042

(DS 15) ?x/Person, email ,?y/Thing

0. [Person > Thing]; [@e-mail] 9.35164

1. [Author > Thing]; [@e-mail] 2.47883

2. [Editor > Thing]; [@e-mail] 1.58420

3. [Publisher < Publication > Author > Thing]; [publisher, author, @e-mail] 0.13727

4. [Publisher < Proceedings > Author > Thing]; [publisher, editor, @e-mail] 0.06119

5. [Ŝubject < InBook > Editor < Thing]; [@subject, author, author] 0.00000

6. [N̂ame < Institution < Person < Thing]; [@name, institution, author] 0.00000

(DS 15) ?x/Person, email ,?y/L̂iteral

0. [Person > L̂iteral]; [@e-mail] 9.35164

1. [Author > L̂iteral]; [@e-mail] 2.47883

2. [Editor > L̂iteral]; [@e-mail] 1.58420

3. [Person > N̂umber]; [@e-mail] 0.28043

4. [Publisher < Publication > Author > L̂iteral]; [publisher, author, @e-mail] 0.13727

5. [Author > N̂umber]; [@e-mail] 0.07433

6. [Publisher < Proceedings > Author > L̂iteral]; [publisher, editor, @e-mail] 0.06119

7. [Editor < Book < Article > N̂umber]; [author, cites, @DOI] 0.00000

8. [Publisher < Publication > Editor > N̂umber]; [publisher, author, @numberOfCitations] 0.00000

(DS 15) ?x/Person, email ,?y/String

0. [Person > L̂iteral]; [@e-mail] 2.13840

1. [Person > T̂ext]; [@e-mail] 1.19319

2. [Author > L̂iteral]; [@e-mail] 0.56683

3. [Person > N̂umber]; [@e-mail] 0.38141

4. [Editor > L̂iteral]; [@e-mail] 0.36225

5. [Author > T̂ext]; [@e-mail] 0.31628

6. [Editor > T̂ext]; [@e-mail] 0.20213

7. [Author > N̂umber]; [@e-mail] 0.10110

8. [Publisher < Publication > Author > L̂iteral]; [publisher, author, @e-mail] 0.03139

9. [Publisher < Publication > Person > T̂ext]; [publisher, author, @e-mail] 0.01751

(DS 15) ?x/Person, email ,?y/Attribute

0. [Person > L̂iteral]; [@e-mail] 1.60519

1. [Author > L̂iteral]; [@e-mail] 0.42549

2. [Editor > L̂iteral]; [@e-mail] 0.27192

3. [Publisher < Publication > Author > L̂iteral]; [publisher, author, @e-mail] 0.02356

4. [Publisher < Proceedings > Author > L̂iteral]; [publisher, editor, @e-mail] 0.01050

5. [Person < Article > Editor > N̂ame]; [author, author, @name] 0.00000

6. [Author < Thesis < Paper > N̂ame]; [author, cites, @name] 0.00000

7. [Publisher < Book > Person > N̂ame]; [publisher, author, @name] 0.00000

8. [Editor < Publication > Paper > N̂ame]; [author, cites, @name] 0.00000

(DS 16) ?x/Person, ,?y/homepage

0. [Person > Ĥomepage]; [@homepage] 12.53285

1. [Author > Ĥomepage]; [@homepage] 3.32207

2. [Editor > Ĥomepage]; [@homepage] 2.26448

3. [Person > Ê-mail]; [@e-mail] 0.93258

4. [Person < Publication]; [author] 0.32481

5. [Person < Paper > Âbstract]; [author, @abstract] 0.31117

6. [Author > Ê-mail]; [@e-mail] 0.24720

7. [Person < Publication > Âbstract]; [editor, @abstract] 0.16086

8. [Editor > Ê-mail]; [@e-mail] 0.15798

9. [Person < Publication]; [editor] 0.12432

(DS 16) ?x/Person, ,?y/website

0. [Person > Ĥomepage]; [@homepage] 2.30448

1. [Person > Ê-mail]; [@e-mail] 1.54092

2. [Person < Publication]; [author] 0.75836

3. [Author > Ĥomepage]; [@homepage] 0.61085

4. [Person < Publication]; [editor] 0.43902

5. [Editor > Ĥomepage]; [@homepage] 0.41638

6. [Author > Ê-mail]; [@e-mail] 0.40845

7. [Person < Article]; [author] 0.39699

8. [Author < Publication]; [author] 0.36353

9. [Person < Paper > Journal]; [author, journal] 0.29834

(DS 16) ?x/Person, ,?y/home page

0. [Person > Ĥomepage]; [@homepage] 11.70788

1. [Author > Ĥomepage]; [@homepage] 3.10340

2. [Editor > Ĥomepage]; [@homepage] 2.11542

3. [Person > T̂ext]; [@homepage] 1.42871

4. [Person > Ê-mail]; [@e-mail] 1.05688

5. [Person < Paper]; [author] 0.83291

6. [Person > T̂ext]; [@e-mail] 0.62049

7. [Person < Article]; [author] 0.47458

8. [Person < Publication]; [author] 0.47361

9. [Author > T̂ext]; [@homepage] 0.37871

(DS 16) ?x/Person, homepage, ?y/Thing

0. [Person > Thing]; [@homepage] 12.53285

1. [Person > Thing]; [@e-mail] 5.96293

2. [Person < Thing]; [author] 3.57188

3. [Author > Thing]; [@homepage] 3.32207

4. [Person < Paper > Thing]; [author, @pageNumbers] 2.99899

5. [Editor > Thing]; [@homepage] 2.26448

6. [Author > Thing]; [@e-mail] 1.58059

7. [Editor > Thing]; [@e-mail] 1.01014

8. [Author < Thing]; [author] 0.94672

9. [Author < Paper > Thing]; [author, @pageNumbers] 0.79494

(DS 16) ?x/Person, homepage, ?y/Attribute

0. [Person > L̂iteral]; [@homepage] 2.15123

1. [Person > L̂iteral]; [@e-mail] 1.02352

2. [Author > L̂iteral]; [@homepage] 0.57023

3. [Person < Paper > L̂iteral]; [author, @pageNumbers] 0.51477

4. [Editor > L̂iteral]; [@homepage] 0.38869

5. [Author > L̂iteral]; [@e-mail] 0.27130

6. [Person < Paper > N̂ame]; [author, @name] 0.23379

7. [Editor > L̂iteral]; [@e-mail] 0.17339

8. [Person < Paper > Author > N̂ame]; [author, author, @name] 0.14693

9. [Author < Paper > L̂iteral]; [author, @pageNumbers] 0.13645

(DS 16) ?x/Person, homepage, ?y/String

0. [Person > L̂iteral]; [@homepage] 2.86584

1. [Person > T̂ext]; [@homepage] 1.59910

2. [Person > L̂iteral]; [@e-mail] 1.36352

3. [Person > T̂ext]; [@e-mail] 0.76082

4. [Author > L̂iteral]; [@homepage] 0.75965

5. [Person < Paper > L̂iteral]; [author, @pageNumbers] 0.68577

6. [Editor > L̂iteral]; [@homepage] 0.51781

7. [Author > T̂ext]; [@homepage] 0.42387

8. [Author > L̂iteral]; [@e-mail] 0.36143

9. [Person < Paper > N̂umber]; [author, @pageNumbers] 0.34347

(DS 17) ?x/Paper, has, ?y/Abstract

0. [Paper > Âbstract]; [@abstract] 15.76089

1. [Article > Âbstract]; [@abstract] 7.67913

2. [Publication > Âbstract]; [@abstract] 5.91323

3. [Paper > Publication > Âbstract]; [proceedings, @abstract] 3.42110

4. [InBook > Âbstract]; [@abstract] 2.21296

5. [Paper > Journal]; [journal] 1.76882

6. [Thesis > Âbstract]; [@abstract] 1.59868

7. [Publication > Publication > Âbstract]; [book, @abstract] 1.49943

8. [Publication < Publication > Âbstract]; [book, @abstract] 1.49943

9. [Publication < InProceedings > Âbstract]; [proceedings, @abstract] 1.30259

(DS 17) ?x/Paper, abstract, ?y/Thing

0. [Paper > Thing]; [@abstract] 15.76089

1. [Article > Thing]; [@abstract] 7.67913

2. [Paper > Thing]; [journal] 7.31341

3. [Paper > Thing]; [@numberOfCitations] 6.08695

4. [Publication > Thing]; [@abstract] 5.91323

5. [Paper > Thing]; [author] 4.97858

6. [Paper > Journal < Thing]; [journal, journal] 4.11799

7. [Article > Thing]; [journal] 3.81997

8. [Article > Thing]; [@numberOfCitations] 3.01819

9. [Publication > Thing]; [journal] 2.74345

(DS 17) ?x/Paper, has, ?y/Description

0. [Paper > N̂ame]; [@name] 0.81713

1. [Paper > T̂ext]; [@name] 0.55682

2. [Article > N̂ame]; [@name] 0.40517

3. [Paper > Âbstract]; [@abstract] 0.32979

4. [Paper > Publication > N̂ame]; [cites, @name] 0.30956

5. [Paper < Paper > N̂ame]; [cites, @name] 0.30927

6. [Publication > N̂ame]; [@name] 0.30704

7. [Paper > Person > N̂ame]; [author, @name] 0.30399

8. [Article > T̂ext]; [@name] 0.27610

9. [Paper > T̂ext]; [@abstract] 0.24539

(DS 18) ?x/book, in, ?y/series

0. [Book > Series]; [series] 10.32869

1. [Publication > Series]; [series] 5.48369

2. [InBook > Book > Series]; [book, series] 5.27524

3. [Publication > Book > Series]; [book, series] 3.61617

4. [Book < InBook > Book > Series]; [book, book, series] 3.48814

5. [Book < Publication > Series]; [book, series] 3.20158

6. [Journal < Publication > Series]; [journal, series] 2.89478

7. [Book > Series < Publication]; [series, series] 2.76113

8. [Book < Publication]; [book] 2.27214

9. [InBook > Publication]; [book] 1.84481

(DS 18) ?y/Series, has books, ?x/Book

0. [Series < Book]; [series] 10.32859

1. [Series < Book < InBook]; [series, book] 5.27524

2. [Series < Publication]; [series] 4.90022

3. [Series < Book < Publication]; [series, book] 3.61617

4. [Publication > Book]; [book] 3.60379

5. [Series < Book < InBook > Book]; [series, book, book] 3.48814

6. [Series < Publication > Book]; [series, book] 3.20158

7. [Series < Publication > Journal]; [series, journal] 2.81360

8. [Publication < InBook]; [book] 2.70478

9. [Publication < InBook > Book]; [book, book] 2.56083

(DS 19) ?x/paper, references, ?y/paper

0. [Paper > Paper]; [cites] 14.65611

1. [Paper > Article]; [cites] 7.27292

2. [Article > Paper]; [cites] 7.22860

3. [Paper > Publication]; [cites] 5.50731

4. [Publication > Paper]; [cites] 5.50099

5. [Article > Article]; [cites] 3.62014

6. [Paper > Book]; [cites] 3.53542

7. [Book > Paper]; [cites] 2.85140

8. [Publication > Article]; [cites] 2.73003

9. [Article > Publication]; [cites] 2.71680

(DS 19) ?x/paper, cites, ?y/paper

0. [Paper > Paper]; [cites] 17.52569

1. [Paper > Article]; [cites] 8.69692

2. [Article > Paper]; [cites] 8.64392

3. [Paper > Publication]; [cites] 6.58561

4. [Publication > Paper]; [cites] 6.57805

5. [Article > Article]; [cites] 4.32894

6. [Paper > Book]; [cites] 4.22763

7. [Book > Paper]; [cites] 3.40969

8. [Publication > Article]; [cites] 3.26455

9. [Article > Publication]; [cites] 3.24874

(DS 19) ?x/paper, refers to, ?y/paper

0. [Paper > Paper]; [cites] 14.25909

1. [Paper > Article]; [cites] 7.07591

2. [Article > Paper]; [cites] 7.03279

3. [Paper > Publication]; [cites] 5.35812

4. [Publication > Paper]; [cites] 5.35197

5. [Article > Article]; [cites] 3.52207

6. [Paper > Book]; [cites] 3.43965

7. [Book > Paper]; [cites] 2.77416

8. [Publication > Article]; [cites] 2.65607

9. [Article > Publication]; [cites] 2.64321

(DS 19) ?x/paper, cited by, ?y/paper

0. [Paper < Paper]; [cites] 17.52559

1. [Article < Paper]; [cites] 8.69682

2. [Paper < Article]; [cites] 8.64382

3. [Publication < Paper]; [cites] 6.58551

4. [Paper < Publication]; [cites] 6.57795

5. [Article < Article]; [cites] 4.32884

6. [Book < Paper]; [cites] 4.22753

7. [Paper < Book]; [cites] 3.40959

8. [Article < Publication]; [cites] 3.26445

9. [Publication < Article]; [cites] 3.24864

(DS 19) ?x/paper, referenced by, ?y/paper

0. [Paper < Paper]; [cites] 14.65601

1. [Article < Paper]; [cites] 7.27282

2. [Paper < Article]; [cites] 7.22850

3. [Publication < Paper]; [cites] 5.50721

4. [Paper < Publication]; [cites] 5.50089
5. [Article < Article]; [cites] 3.62004
6. [Book < Paper]; [cites] 3.53532
7. [Paper < Book]; [cites] 2.85130
8. [Article < Publication]; [cites] 2.72993
9. [Publication < Article]; [cites] 2.71670

(DS 19) ?x/paper, has, ?y/reference

0. [Paper > Publication]; [cites] 4.42963
1. [Paper < Publication]; [cites] 4.42445
2. [Paper < Paper > Publication]; [cites, cites] 3.25706
3. [Paper > Publication < Publication]; [cites, cites] 3.25596
4. [Article < Publication]; [cites] 2.19571
5. [Article > Publication]; [cites] 2.18518
6. [Paper > Book < Publication]; [cites, book] 1.84898
7. [Publication > Publication]; [cites] 1.66263
8. [Publication < Publication]; [cites] 1.66253
9. [Article < Paper > Publication]; [cites, cites] 1.59762

(DS 19) ?y/paper, has, ?x/citations

0. [Paper > Âbstract]; [@abstract] 2.28703
1. [Paper > Publication]; [cites] 1.86588
2. [Paper < Publication]; [cites] 1.86364
3. [Paper < Paper > Âbstract]; [cites, @abstract] 1.82311
4. [Paper > Paper > Âbstract]; [cites, @abstract] 1.82311
5. [Paper > Article]; [cites] 1.74779
6. [Paper < Article]; [cites] 1.73704
7. [Paper < Paper > Publication]; [cites, cites] 1.35436
8. [Paper > Publication < Publication]; [cites, cites] 1.35390
9. [Paper < Paper > Article]; [cites, cites] 1.25335

(DS 20) ?x/Person, citations, ?y/Number

0. [Person > N̂umber]; [@numberOfCitations] 13.36638
1. [Person < Paper > N̂umber]; [author, @numberOfCitations] 5.59759
2. [Person < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 4.49626
3. [Person < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 4.49127
4. [Person < Paper > Publication > N̂umber]; [author, cites, @pageNumbers] 4.41905
5. [Author > N̂umber]; [@numberOfCitations] 3.54302
6. [Editor > N̂umber]; [@numberOfCitations] 2.05204

7. [Author < Paper > N̂umber]; [author, @numberOfCitations] 1.48375

8. [Author < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 1.19182

9. [Author < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 1.19050

(DS 20) ?x/Person, , ?y/number of citations

0. [Person > N̂umber]; [@numberOfCitations] 13.36638

1. [Person > N̂umber]; [@numberOfPublications] 8.18487

2. [Person < Paper > N̂umber]; [author, @numberOfCitations] 5.33886

3. [Author > N̂umber]; [@numberOfCitations] 3.54302

4. [Person < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 3.50961

5. [Person < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 3.50572

6. [Author > N̂umber]; [@numberOfPublications] 2.16956

7. [Editor > N̂umber]; [@numberOfCitations] 2.05204

8. [Author < Paper > N̂umber]; [author, @numberOfCitations] 1.41517

9. [Editor > N̂umber]; [@numberOfPublications] 1.25656

(DS 20) ?x/Person, number of citations, ?y/Thing

0. [Person > Thing]; [@numberOfCitations] 16.24161

1. [Person > Thing]; [@numberOfPublications] 9.94551

2. [Person < Paper > Thing]; [author, @numberOfCitations] 6.44280

3. [Person < Paper > Thing]; [author, cites] 4.72602

4. [Person < Paper < Thing]; [author, cites] 4.71952

5. [Author > Thing]; [@numberOfCitations] 4.30515

6. [Author > Thing]; [@numberOfPublications] 2.63625

7. [Editor > Thing]; [@numberOfCitations] 2.45973

8. [Author < Paper > Thing]; [author, @numberOfCitations] 1.70779

9. [Editor > Thing]; [@numberOfPublications] 1.50621

(DS 20) ?x/Author, citations, ?y/Number

0. [Author > N̂umber]; [@numberOfCitations] 13.36638

1. [Author < Paper > N̂umber]; [author, @numberOfCitations] 11.51562

2. [Author < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 8.53199

3. [Author < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 8.52252

4. [Author < Paper > Publication > N̂umber]; [author, cites, @pageNumbers] 8.38549

5. [Person > N̂umber]; [@numberOfCitations] 3.54302

6. [Person < Paper > N̂umber]; [author, @numberOfCitations] 3.05244

7. [Article > N̂umber]; [@numberOfCitations] 2.80501

8. [Editor < Paper > N̂umber]; [author, @numberOfCitations] 2.63746

9. [Editor > N̂umber]; [@numberOfCitations] 2.62983

(DS 20) ?x/Author, , ?y/number of citations

0. [Author > N̂umber]; [@numberOfCitations] 13.36638

1. [Author < Paper > N̂umber]; [author, @numberOfCitations] 11.51562

2. [Author > N̂umber]; [@numberOfPublications] 8.18487

3. [Author < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 6.65976

4. [Author < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 6.65237

5. [Person > N̂umber]; [@numberOfCitations] 3.54302

6. [Person < Paper > N̂umber]; [author, @numberOfCitations] 3.05244

7. [Article > N̂umber]; [@numberOfCitations] 2.80501

8. [Editor < Paper > N̂umber]; [author, @numberOfCitations] 2.63746

9. [Editor > N̂umber]; [@numberOfCitations] 2.62983

(DS 20) ?x/Author, number of citations, ?y/Thing

0. [Author > Thing]; [@numberOfCitations] 16.24161

1. [Author < Paper > Thing]; [author, @numberOfCitations] 13.89676

2. [Author > Thing]; [@numberOfPublications] 9.94551

3. [Author < Paper > Thing]; [author, cites] 9.30962

4. [Author < Paper < Thing]; [author, cites] 9.29682

5. [Person > Thing]; [@numberOfCitations] 4.30515

6. [Person < Paper > Thing]; [author, @numberOfCitations] 3.68360

7. [Article > Thing]; [@numberOfCitations] 3.40582

8. [Editor < Paper > Thing]; [author, @numberOfCitations] 3.16199

9. [Editor > Thing]; [@numberOfCitations] 3.15232

(DS 20) ?x/Scholar, citations, ?y/Number

0. [Author > N̂umber]; [@numberOfCitations] 3.98290

1. [Editor > N̂umber]; [@numberOfCitations] 1.65221

2. [Author < Paper > N̂umber]; [author, @numberOfCitations] 1.63288

3. [Publication > N̂umber]; [@numberOfCitations] 1.48436

4. [Institution > N̂umber]; [@numberOfCitations] 1.37221

5. [Journal > N̂umber]; [@numberOfCitations] 1.34376

6. [Author < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 1.31626

7. [Author < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 1.31480

8. [Author < Paper > Publication > N̂umber]; [author, cites, @pageNumbers] 1.29366

9. [Publication < Paper > N̂umber]; [cites, @numberOfCitations] 1.08496

(DS 20) ?x/Scholar, , ?y/number of citations

0. [Author > N̂umber]; [@numberOfCitations] 3.98290

1. [Author > N̂umber]; [@numberOfPublications] 2.43892
2. [Editor > N̂umber]; [@numberOfCitations] 1.65221
3. [Author < Paper > N̂umber]; [author, @numberOfCitations] 1.55740
4. [Publication > N̂umber]; [@numberOfCitations] 1.48436
5. [Institution > N̂umber]; [@numberOfCitations] 1.37221
6. [Journal > N̂umber]; [@numberOfCitations] 1.34376
7. [Author < Paper > Publication > N̂umber]; [author, cites, @numberOfCitations] 1.02743
8. [Author < Paper < Paper > N̂umber]; [author, cites, @numberOfCitations] 1.02629
9. [Editor > N̂umber]; [@numberOfPublications] 1.01173

(DS 20) ?x/Scholar, number of citations, ?y/Thing

0. [Author > Thing]; [@numberOfCitations] 4.83967
1. [Author > Thing]; [@numberOfPublications] 2.96356
2. [Editor > Thing]; [@numberOfCitations] 1.98047
3. [Author < Paper > Thing]; [author, @numberOfCitations] 1.87943
4. [Publication > Thing]; [@numberOfCitations] 1.80623
5. [Institution > Thing]; [@numberOfCitations] 1.64491
6. [Journal > Thing]; [@numberOfCitations] 1.58567
7. [Author < Paper > Thing]; [author, cites] 1.37863
8. [Author < Paper < Thing]; [author, cites] 1.37673
9. [Editor > Thing]; [@numberOfPublications] 1.21274

(DS 21) ?x/Person, publications, ?y/Number

0. [Person > N̂umber]; [@numberOfPublications] 12.73856
1. [Person < Paper > N̂umber]; [author, @numberOfCitations] 5.95171
2. [Person < Paper > N̂umber]; [author, @pageNumbers] 5.85204
3. [Person < Paper > N̂umber]; [author, @volumeNumber] 4.05755
4. [Person < Publication > N̂umber]; [author, @numberOfPublications] 3.87573
5. [Author > N̂umber]; [@numberOfPublications] 3.37660
6. [Editor > N̂umber]; [@numberOfPublications] 1.95565
7. [Author < Paper > N̂umber]; [author, @numberOfCitations] 1.57762
8. [Author < Paper > N̂umber]; [author, @pageNumbers] 1.55120
9. [Author > N̂umber]; [@numberOfCitations] 1.46344

(DS 21) ?x/Person, , ?y/number of publications

0. [Person > N̂umber]; [@numberOfPublications] 12.73856
1. [Person > N̂umber]; [@numberOfCitations] 7.80042
2. [Person < Paper > N̂umber]; [author, @volumeNumber] 4.20660
3. [Person < Paper > N̂umber]; [author, @issueNumber] 3.88664

4. [Person < Paper > N̂umber]; [author, @pageNumbers] 3.68102

5. [Author > N̂umber]; [@numberOfPublications] 3.37660

6. [Author > N̂umber]; [@numberOfCitations] 2.06765

7. [Editor > N̂umber]; [@numberOfPublications] 1.95565

8. [Person > L̂iteral]; [@numberOfPublications] 1.27046

9. [Editor > N̂umber]; [@numberOfCitations] 1.19754

(DS 21) ?x/Person, number of publications, ?y/Thing

0. [Person > Thing]; [@numberOfPublications] 16.24161

1. [Person > Thing]; [@numberOfCitations] 9.94551

2. [Person < Thing]; [author] 5.83655

3. [Person < Thing]; [firstAuthor] 5.76807

4. [Person < Thing]; [secondAuthor] 5.45721

5. [Author > Thing]; [@numberOfPublications] 4.30515

6. [Author > Thing]; [@numberOfCitations] 2.63625

7. [Editor > Thing]; [@numberOfPublications] 2.45973

8. [Author < Thing]; [author] 1.54702

9. [Author < Thing]; [firstAuthor] 1.52886

(DS 21) ?x/Author, publications, ?y/Number

0. [Author > N̂umber]; [@numberOfPublications] 12.73856

1. [Author < Publication > N̂umber]; [author, @numberOfPublications] 7.42711

2. [Author < Paper > N̂umber]; [author, @volumeNumber] 7.24186

3. [Author < Paper > N̂umber]; [author, @numberOfCitations] 5.95171

4. [Author < Paper > N̂umber]; [author, @pageNumbers] 5.85204

5. [Person > N̂umber]; [@numberOfPublications] 3.37660

6. [Editor > N̂umber]; [@numberOfPublications] 2.50630

7. [Article > Person > N̂umber]; [author, @numberOfPublications] 2.24587

8. [Publication > N̂umber]; [@numberOfCitations] 2.05868

9. [Publication > N̂umber]; [@pageNumbers] 2.01839

(DS 21) ?x/Author, , ?y/number of publications

0. [Author > N̂umber]; [@numberOfPublications] 12.73856

1. [Author < Paper > N̂umber]; [author, @volumeNumber] 8.78315

2. [Author < Paper > N̂umber]; [author, @issueNumber] 7.97233

3. [Author > N̂umber]; [@numberOfCitations] 7.80042

4. [Author < Publication > N̂umber]; [author, @numberOfPublications] 7.42711

5. [Person > N̂umber]; [@numberOfPublications] 3.37660

6. [Editor > N̂umber]; [@numberOfPublications] 2.50630

7. [Person < Paper > N̂umber]; [author, @volumeNumber] 2.32814
8. [Article > N̂umber]; [@volumeNumber] 2.27206
9. [Article > Person > N̂umber]; [author, @numberOfPublications] 2.24587

(DS 21) ?x/Author, number of publications, ?y/Thing

0. [Author > Thing]; [@numberOfPublications] 16.24161
1. [Author < Paper > Thing]; [author, @volumeNumber] 11.08725
2. [Author < Paper > Thing]; [author, @issueNumber] 10.05937
3. [Author > Thing]; [@numberOfCitations] 9.94551
4. [Author < Paper > Thing]; [author, journal] 9.92603
5. [Person > Thing]; [@numberOfPublications] 4.30515
6. [Editor > Thing]; [@numberOfPublications] 3.15232
7. [Person < Paper > Thing]; [author, @volumeNumber] 2.93889
8. [Article > Thing]; [@volumeNumber] 2.89468
9. [Article > Author > Thing]; [author, @numberOfPublications] 2.83765

(DS 21) ?x/Scholar, publications, ?y/Number

0. [Author > N̂umber]; [@numberOfPublications] 3.79583
1. [Author < Paper > N̂umber]; [author, @numberOfCitations] 1.77349
2. [Journal < Article > N̂umber]; [journal, @numberOfCitations] 1.75116
3. [Author < Paper > N̂umber]; [author, @pageNumbers] 1.74379
4. [Journal < Article > N̂umber]; [journal, @volumeNumber] 1.74290
5. [Journal < Paper > N̂umber]; [journal, @pageNumbers] 1.71955
6. [Journal < Article > N̂umber]; [journal, @issueNumber] 1.67771
7. [Author > N̂umber]; [@numberOfCitations] 1.64514
8. [Editor > N̂umber]; [@numberOfPublications] 1.57461
9. [Publication > N̂umber]; [@numberOfCitations] 1.48436

(DS 21) ?x/Scholar, , ?y/number of publications

0. [Author > N̂umber]; [@numberOfPublications] 3.79583
1. [Author > N̂umber]; [@numberOfCitations] 2.32436
2. [Editor > N̂umber]; [@numberOfPublications] 1.57461
3. [Institution > N̂umber]; [@numberOfPublications] 1.30776
4. [Journal > N̂umber]; [@numberOfPublications] 1.28065
5. [Author < Paper > N̂umber]; [author, @volumeNumber] 1.22711
6. [Publication > N̂umber]; [@volumeNumber] 1.14285
7. [Journal < Article > N̂umber]; [journal, @volumeNumber] 1.13945
8. [Author < Paper > N̂umber]; [author, @issueNumber] 1.13377
9. [Author < Paper > N̂umber]; [author, @pageNumbers] 1.07379

(DS 21) ?x/Scholar, number of publications, ?y/Thing

0. [Author > Thing]; [@numberOfPublications] 4.83967

1. [Author > Thing]; [@numberOfCitations] 2.96356

2. [Journal < Thing]; [journal] 2.04528

3. [Editor > Thing]; [@numberOfPublications] 1.98047

4. [Author < Thing]; [author] 1.73910

5. [Author < Thing]; [firstAuthor] 1.71870

6. [Institution > Thing]; [@numberOfPublications] 1.64491

7. [Author < Thing]; [secondAuthor] 1.62607

8. [Journal > Thing]; [@numberOfPublications] 1.58567

9. [Publication > Thing]; [@volumeNumber] 1.45608

(DS 22) ?x/Author, in, ?y/Institution

0. [Author > Institution]; [institution] 13.99929

1. [Author < Paper > Institution]; [author, institution] 12.14094

2. [Author < Publication > Institution]; [editor, institution] 4.30837

3. [Person < Paper > Institution]; [author, institution] 3.21819

4. [Author < Paper > Publication > Institution]; [author, cites, institution] 2.81596

5. [Author < Paper < Paper > Institution]; [author, cites, institution] 2.81423

6. [Editor < Paper > Institution]; [author, institution] 2.70707

7. [Article > Author > Institution]; [author, institution] 2.35877

8. [Person > Institution]; [institution] 2.02692

9. [Editor > Institution]; [institution] 1.96260

(DS 22) ?x/Person, in, ?y/Institution

0. [Person > Institution]; [institution] 13.99929

1. [Person < Paper > Institution]; [author, institution] 5.62877

2. [Person < Publication > Institution]; [editor, institution] 2.81189

3. [Author > Institution]; [institution] 2.02692

4. [Person > Institution < Publication]; [institution, institution] 1.42739

5. [Author < Paper > Institution]; [author, institution] 1.30030

6. [Editor > Institution]; [institution] 1.05906

7. [Editor < Paper > Institution]; [author, institution] 0.84270

8. [Author < Publication > Institution]; [editor, institution] 0.62927

9. [Editor < Publication > Institution]; [editor, institution] 0.47236

(DS 22) ?x/Scholar, in, ?y/Institution

0. [Author > Institution]; [institution] 2.33396

1. [Author < Paper > Institution]; [author, institution] 1.43540
2. [Editor > Institution]; [institution] 0.71175
3. [Editor < Paper > Institution]; [author, institution] 0.65609
4. [Author < Publication > Institution]; [editor, institution] 0.59240
5. [Publication > Institution]; [institution] 0.50316
6. [Publication > Person > Institution]; [author, institution] 0.48976
7. [Journal < Paper > Institution]; [journal, institution] 0.47352
8. [Author > Institution < Publication]; [institution, institution] 0.42533
9. [Institution < InProceedings > Institution]; [institution, institution] 0.42357

(DS 22) ?x/Scholar, works at, ?y/Institution

0. [Author > Institution]; [institution] 0.90843
1. [Author < Paper > Institution]; [author, institution] 0.78784
2. [Author < Paper > Person > Institution]; [author, author, institution] 0.50564
3. [Publication > Publication > Institution]; [book, institution] 0.43851
4. [Publication < Publication > Institution]; [book, institution] 0.43851
5. [Author < Publication > Publication > Institution]; [author, book, institution] 0.40584
6. [Author < Publication < Publication > Institution]; [author, book, institution] 0.40584
7. [Editor < Paper > Institution]; [author, institution] 0.37037
8. [Publication > Person > Institution]; [author, institution] 0.27974
9. [Publication > Institution]; [institution] 0.25545

(DS 22) ?x/Person, works at, ?y/Institution

0. [Person < Paper > Institution]; [author, institution] 2.64393
1. [Person < Paper > Author > Institution]; [author, author, institution] 1.72724
2. [Person < Publication > Publication > Institution]; [author, book, institution] 1.38631
3. [Person < Publication < Publication > Institution]; [author, book, institution] 1.38631
4. [Person > Institution]; [institution] 1.05041
5. [Author > Institution]; [institution] 0.80810
6. [Author < Paper > Institution]; [author, institution] 0.70083
7. [Editor < Paper > Institution]; [author, institution] 0.46000
8. [Author < Paper > Person > Institution]; [author, author, institution] 0.45784
9. [Person < Publication]; [author] 0.43796

(DS 22) ?x/Person, works with, ?y/Institution

0. [Person < Paper > Institution]; [author, institution] 2.64393
1. [Person < Paper > Author > Institution]; [author, author, institution] 1.72724
2. [Person < Publication > Publication > Institution]; [author, book, institution] 1.38631
3. [Person < Publication < Publication > Institution]; [author, book, institution] 1.38631

4. [Person > Institution]; [institution] 1.05041

5. [Author > Institution]; [institution] 0.80810

6. [Author < Paper > Institution]; [author, institution] 0.70083

7. [Editor < Paper > Institution]; [author, institution] 0.46000

8. [Author < Paper > Person > Institution]; [author, author, institution] 0.45784

9. [Person < Publication]; [author] 0.43796

(DS 22) ?y/Institution, has author, ?x/Author

0. [Institution < Author]; [institution] 13.99919

1. [Institution < Paper > Author]; [institution, author] 12.14094

2. [Institution < Publication > Author]; [institution, editor] 3.77765

3. [Institution < Paper > Person]; [institution, author] 3.21819

4. [Institution < Publication < Paper > Author]; [institution, cites, author] 2.81596

5. [Institution < Paper > Paper > Author]; [institution, cites, author] 2.81423

6. [Institution < Paper > Editor]; [institution, author] 2.70707

7. [Institution < Author < Article]; [institution, author] 2.35877

8. [Publication > Author]; [author] 2.01157

9. [Institution < Person < Publication]; [institution, author] 1.78160

(DS 22) ?y/Institution, has author, ?x/Person

0. [Institution < Paper > Person]; [institution, author] 12.14094

1. [Institution < Person]; [institution] 5.35549

2. [Institution < Publication > Person]; [institution, editor] 3.77765

3. [Institution < Author]; [institution] 3.71068

4. [Institution < Paper > Author]; [institution, author] 3.21819

5. [Institution < Publication < Paper > Person]; [institution, cites, author] 2.81596

6. [Institution < Paper > Paper > Person]; [institution, cites, author] 2.81423

7. [Institution < Person < Paper > Ŝubject]; [institution, author, @subject] 2.47786

8. [Institution < Paper > Author > Ŝubject]; [institution, author, @subject] 2.26366

9. [Institution < Paper > Editor]; [institution, author] 2.11231

(DS 22) ?y/Institution, has author, ?x/Scholar

0. [Institution < Author]; [institution] 4.17140

1. [Institution < Paper > Author]; [institution, author] 3.61775

2. [Institution < Paper > Editor]; [institution, author] 1.70074

3. [Institution < Person < Publication]; [institution, author] 1.28458

4. [Institution < Publication > Author]; [institution, editor] 1.12566

5. [Institution < Editor]; [institution] 0.99360

6. [Institution < Publication < Paper > Author]; [institution, cites, author] 0.83910

7. [Institution < Paper > Paper > Author]; [institution, cites, author] 0.83858

8. [Institution < Publication]; [institution] 0.66779

9. [Institution < Publication > Editor]; [institution, editor] 0.61491

(DS 23) ?x/Conference, , ?y/Proceedings

0. [Conference < InProceedings > Proceedings]; [conference, proceedings] 12.24266

1. [Conference < Proceedings]; [conference] 12.11501

2. [Conference < InProceedings]; [conference] 11.42046

3. [Conference < Publication < InProceedings]; [conference, proceedings] 8.74734

4. [Conference < InProceedings > Proceedings]; [venue, proceedings] 4.43866

5. [Conference < InProceedings]; [venue] 3.21288

6. [Conference < Publication < InProceedings]; [venue, proceedings] 3.16706

7. [Conference < InProceedings > InProceedings]; [conference, proceedings] 2.78067

8. [Publication > Conference < InProceedings]; [conference, conference] 1.60607

9. [Publication > Conference < Proceedings]; [conference, conference] 1.53738

(DS 23) ?x/Workshop, has, ?y/Proceedings

0. [Conference < InProceedings > Proceedings]; [conference, proceedings] 2.49500

1. [Conference < InProceedings]; [conference] 2.34024

2. [Conference < Proceedings]; [conference] 2.32741

3. [Conference < Publication < InProceedings]; [conference, proceedings] 1.81672

4. [Conference < InProceedings > Proceedings]; [venue, proceedings] 1.32229

5. [Conference < InProceedings]; [venue] 1.16549

6. [Conference < Publication < InProceedings]; [venue, proceedings] 0.97936

7. [Conference < InProceedings > InProceedings]; [conference, proceedings] 0.57751

8. [Publication > Proceedings]; [proceedings] 0.31590

9. [Publication > Conference < InProceedings]; [conference, conference] 0.31110

(DS 24) ?x/Book, ISBN, ?y/Number

0. [Book > N̂umber]; [@ISBN] 11.33614

1. [Publication > N̂umber]; [@ISBN] 6.34610

2. [InBook > Book > N̂umber]; [book, @ISBN] 6.03083

3. [Publication > Book > N̂umber]; [book, @ISBN] 4.13413

4. [Book < Publication > N̂umber]; [book, @ISBN] 3.80668

5. [Journal < Publication > N̂umber]; [journal, @ISBN] 3.22601

6. [Publication < Publication > N̂umber]; [book, @ISBN] 1.96183

7. [Thing > N̂umber]; [@ISBN] 1.20369

8. [Author < Publication > N̂umber]; [author, @ISBN] 1.09806

9. [Book > L̂iteral]; [@ISBN] 0.91000

(DS 24) ?x/Book, ISBN, ?y/Thing

0. [Book > Thing]; [@ISBN] 11.34829

1. [Publication > Thing]; [@ISBN] 6.36450

2. [InBook > Book > Thing]; [book, @ISBN] 6.03995

3. [Publication > Book > Thing]; [book, @ISBN] 4.14037

4. [Book < Publication > Thing]; [book, @ISBN] 3.84246

5. [Journal < Publication > Thing]; [journal, @ISBN] 3.24133

6. [Publication < Publication > Thing]; [book, @ISBN] 1.98022

7. [Author < Publication > Thing]; [author, @ISBN] 1.10224

8. [Author < Publication > Book > Thing]; [author, book, @ISBN] 0.65722

9. [Paper > Person < Publication > Thing]; [primaryAuthor, firstAuthor, @ISBN] 0.50623

(DS 24) ?x/Book, ,?y/ISBN

0. [Book > ÎSBN]; [@ISBN] 11.34829

1. [Publication > ÎSBN]; [@ISBN] 6.36450

2. [InBook > Book > ÎSBN]; [book, @ISBN] 6.03995

3. [Publication > Book > ÎSBN]; [book, @ISBN] 4.14037

4. [Book < Publication > ÎSBN]; [book, @ISBN] 3.84246

5. [Journal < Publication > ÎSBN]; [journal, @ISBN] 3.24133

6. [Publication < Publication > ÎSBN]; [book, @ISBN] 1.98022

7. [Thing > ÎSBN]; [@ISBN] 1.20718

8. [Author < Publication > ÎSBN]; [author, @ISBN] 1.10224

9. [Thing > Book > ÎSBN]; [book, @ISBN] 0.78532

(DS 25) ?x/Journal Article, volume no., ?y/Number

0. [Article > N̂umber]; [@volumeNumber] 11.30828

1. [Publication > N̂umber]; [@volumeNumber] 6.39393

2. [Paper > N̂umber]; [@volumeNumber] 6.15807

3. [Article > Journal < Article > N̂umber]; [journal, journal, @numberOfCitations] 5.89140

4. [Article > Journal < Paper > N̂umber]; [journal, journal, @pageNumbers] 5.79140

5. [Article > Journal < Article > N̂umber]; [journal, journal, @issueNumber] 5.66911

6. [Article > Venue > N̂umber]; [journal, @numberOfCitations] 5.00660

7. [Publication > N̂umber]; [@numberOfCitations] 4.73594

8. [Publication > N̂umber]; [@pageNumbers] 4.64326

9. [Journal < Article > N̂umber]; [journal, @volumeNumber] 4.32000

(DS 25) ?x/Article, volume, ?y/Number

0. [Article > N̂umber]; [@volumeNumber] 13.05719

1. [Paper > N̂umber]; [@volumeNumber] 6.82009

2. [Article > Venue > N̂umber]; [journal, @numberOfCitations] 5.78091

3. [Article > Venue > N̂umber]; [journal, @numberOfPublications] 5.56264

4. [Article > Person > N̂umber]; [author, @numberOfPublications] 4.45029

5. [Publication > N̂umber]; [@volumeNumber] 4.24736

6. [Article > N̂umber]; [@pageNumbers] 4.09526

7. [Publication > N̂umber]; [@numberOfCitations] 3.14599

8. [Publication > N̂umber]; [@pageNumbers] 3.08442

9. [Book > N̂umber]; [@numberOfCitations] 3.05237

(DS 25) ?x/Article, volume number, ?y/Thing

0. [Article > Thing]; [@volumeNumber] 15.94660

1. [Article > Thing]; [@pageNumbers] 10.17352

2. [Article > Thing]; [@numberOfCitations] 9.15356

3. [Article > Thing]; [@issueNumber] 8.84488

4. [Article > Thing]; [journal] 8.75488

5. [Paper > Thing]; [@volumeNumber] 8.32930

6. [Paper > Thing]; [@pageNumbers] 5.60874

7. [Publication > Thing]; [@volumeNumber] 5.18743

8. [Paper > Thing]; [@numberOfCitations] 5.03644

9. [Paper > Thing]; [@issueNumber] 4.61989

(DS 26) ?x/Journal Article, issue no., ?y/Number

0. [Article > N̂umber]; [@issueNumber] 10.82798

1. [Publication > N̂umber]; [@issueNumber] 6.11772

2. [Paper > N̂umber]; [@issueNumber] 5.89651

3. [Journal < Article > N̂umber]; [journal, @issueNumber] 4.15840

4. [Publication > N̂umber]; [@numberOfCitations] 3.75920

5. [Publication > N̂umber]; [@pageNumbers] 3.68563

6. [Publication > N̂umber]; [@volumeNumber] 3.55230

7. [Article > Venue > N̂umber]; [journal, @numberOfPublications] 3.40662

8. [Article > Person > N̂umber]; [author, @numberOfPublications] 2.91336

9. [Article > N̂umber]; [@numberOfCitations] 2.80470

(DS 26) ?x/Article, issue, ?y/Number

0. [Article > N̂umber]; [@issueNumber] 12.50260

1. [Paper > N̂umber]; [@issueNumber] 6.53041

2. [Publication > N̂umber]; [@issueNumber] 4.06388

3. [Article > Person > N̂umber]; [author, @numberOfPublications] 3.55805

4. [Article < Article > N̂umber]; [cites, @issueNumber] 3.32860

5. [Article > Article > N̂umber]; [cites, @issueNumber] 3.32860

6. [Article > N̂umber]; [@numberOfCitations] 3.23847

7. [Publication > N̂umber]; [@numberOfCitations] 2.49716

8. [Publication > N̂umber]; [@pageNumbers] 2.44829

9. [Publication > N̂umber]; [@volumeNumber] 2.35972

(DS 26) ?x/Article, issue number, ?y/Thing

0. [Article > Thing]; [@issueNumber] 15.83554

1. [Article > Thing]; [@numberOfCitations] 9.09315

2. [Article > Thing]; [@volumeNumber] 8.90691

3. [Article > Thing]; [@pageNumbers] 8.76137

4. [Paper > Thing]; [@issueNumber] 8.27128

5. [Article > Thing]; [@publicationYear] 6.23176

6. [Publication > Thing]; [@issueNumber] 5.14723

7. [Paper > Thing]; [@numberOfCitations] 5.00320

8. [Paper > Thing]; [@pageNumbers] 4.83021

9. [Paper > Thing]; [@volumeNumber] 4.65230

(DS 26) ?x/Article, issue number, ?y/Attribute

0. [Article > L̂iteral]; [@issueNumber] 2.71813

1. [Article > L̂iteral]; [@numberOfCitations] 1.56082

2. [Article > L̂iteral]; [@volumeNumber] 1.52885

3. [Article > L̂iteral]; [@pageNumbers] 1.50387

4. [Paper > L̂iteral]; [@issueNumber] 1.41975

5. [Article > L̂iteral]; [@publicationYear] 1.06967

6. [Publication > L̂iteral]; [@issueNumber] 0.88351

7. [Paper > L̂iteral]; [@numberOfCitations] 0.85879

8. [Paper > L̂iteral]; [@pageNumbers] 0.82909

9. [Paper > L̂iteral]; [@volumeNumber] 0.79855

(DS 27) ?x/Journal Article, page no., ?y/Number

0. [Article > N̂umber]; [@pageNumbers] 11.13880

1. [Publication > N̂umber]; [@pageNumbers] 6.63392

2. [Paper > N̂umber]; [@pageNumbers] 6.38832

3. [Article > N̂umber]; [@numberOfCitations] 4.74688

4. [Article > N̂umber]; [@volumeNumber] 4.71907

5. [Paper > N̂umber]; [@numberOfCitations] 4.66706

6. [Article > N̂umber]; [@issueNumber] 4.51863

7. [Paper > N̂umber]; [@volumeNumber] 4.41383

8. [Journal < Paper > N̂umber]; [journal, @pageNumbers] 4.26211

9. [Paper > N̂umber]; [@issueNumber] 4.22636

(DS 27) ?x/Article, pages, ?y/Number

0. [Article > N̂umber]; [@pageNumbers] 12.86149

1. [Paper > N̂umber]; [@pageNumbers] 7.07510

2. [Article > N̂umber]; [@numberOfCitations] 5.48102

3. [Article > N̂umber]; [@volumeNumber] 5.44891

4. [Article > N̂umber]; [@issueNumber] 5.21747

5. [Paper > N̂umber]; [@numberOfCitations] 5.16879

6. [Paper > N̂umber]; [@volumeNumber] 4.88834

7. [Paper > N̂umber]; [@issueNumber] 4.68071

8. [Publication > N̂umber]; [@pageNumbers] 4.40678

9. [Article > Person > N̂umber]; [author, @numberOfCitations] 3.73355

(DS 27) ?x/Article, page number, ?y/Thing

0. [Article > Thing]; [@pageNumbers] 15.84861

1. [Article > Thing]; [@volumeNumber] 10.23642

2. [Article > Thing]; [@numberOfCitations] 9.10766

3. [Article > Thing]; [@issueNumber] 8.75414

4. [Paper > Thing]; [@pageNumbers] 8.73747

5. [Publication > Thing]; [@pageNumbers] 5.44295

6. [Paper > Thing]; [@volumeNumber] 5.34673

7. [Paper > Thing]; [@numberOfCitations] 5.01118

8. [Article > Author > Thing]; [author, @numberOfPublications] 4.89227

9. [Paper > Thing]; [@issueNumber] 4.57250

(DS 28) ?x/Paper, published by, ?y/Institution

0. [Paper > Institution]; [institution] 8.79104

1. [Publication > Institution]; [institution] 5.65078

2. [Paper > Person > Institution]; [author, institution] 4.93890

3. [Article > Institution]; [institution] 3.77933

4. [Article > Author > Institution]; [author, institution] 2.38284

5. [Paper > Publication > Institution]; [cites, institution] 2.21806

6. [Paper < Paper > Institution]; [cites, institution] 2.21703

7. [Paper > Editor < Publication > Institution]; [author, editor, institution] 2.07148

8. [Publication > Person > Institution]; [author, institution] 1.85424

9. [Publication > Publication > Institution]; [book, institution] 1.72903

(DS 28) ?x/Paper, author, ?y/Institution

0. [Paper > Person > Institution]; [author, institution] 11.95715

1. [Article > Author > Institution]; [author, institution] 5.76890

2. [Publication > Person > Institution]; [author, institution] 4.48914

3. [Paper > Institution]; [institution] 3.75489

4. [Paper > Editor < Publication > Institution]; [author, editor, institution] 3.52973

5. [Article > Institution]; [institution] 3.02025

6. [Paper > Paper > Author > Institution]; [cites, author, institution] 2.75648

7. [Paper < Paper > Author > Institution]; [cites, author, institution] 2.75648

8. [Publication > Institution]; [institution] 2.33405

9. [Book > Author > Institution]; [author, institution] 1.82705

(DS 28) ?x/Paper, published by, ?y/University

0. [Paper > Institution]; [institution] 3.76991

1. [Publication > Institution]; [institution] 2.42325

2. [Paper > Person > Institution]; [author, institution] 2.37374

3. [Article > Institution]; [institution] 1.62071

4. [Article > Author > Institution]; [author, institution] 1.14525

5. [Paper > Publication > Institution]; [cites, institution] 1.06605

6. [Paper < Paper > Institution]; [cites, institution] 1.06555

7. [Paper > Editor < Publication > Institution]; [author, editor, institution] 0.92273

8. [Publication > Person > Institution]; [author, institution] 0.89119

9. [Paper > Journal]; [journal] 0.88476

(DS 28) ?x/Paper, institution, ?y/Institution

0. [Paper > Institution]; [institution] 15.04315

1. [Article > Institution]; [institution] 7.06901

2. [Publication > Institution]; [institution] 5.65078

3. [Paper > Person > Institution]; [author, institution] 3.93943

4. [Thesis > Institution]; [institution] 2.27271

5. [InBook > Institution]; [institution] 2.03741

6. [Article > Author > Institution]; [author, institution] 1.90064

7. [Author > Institution]; [institution] 1.80712

8. [Book > Institution]; [institution] 1.69286

9. [Paper > Institution < Publication]; [institution, institution] 1.54524

(DS 28) ?x/Paper, published by, ?y/Organization

0. [Paper > Institution]; [institution] 4.97475

1. [Paper > Person > Institution]; [author, institution] 3.23133

2. [Publication > Institution]; [institution] 3.19771

3. [Article > Institution]; [institution] 2.13868

4. [Article > Author > Institution]; [author, institution] 1.55900

5. [Paper > Publication > Institution]; [cites, institution] 1.45119

6. [Paper < Paper > Institution]; [cites, institution] 1.45052

7. [Paper > Editor < Publication > Institution]; [author, editor, institution] 1.23032

8. [Publication > Person > Institution]; [author, institution] 1.21316

9. [Paper > Journal]; [journal] 1.09015

(DS 28) ?y/Institution, has paper, ?x/Paper

0. [Institution < Paper]; [institution] 15.04305

1. [Institution < Article]; [institution] 5.64755

2. [Institution < Publication]; [institution] 3.49595

3. [Institution < Author < Article]; [institution, author] 1.43996

4. [Institution < Publication < Publication]; [institution, book] 1.16770

5. [Institution < Publication > Publication]; [institution, book] 1.16770

6. [Institution < Person < Publication]; [institution, author] 1.12052

7. [Institution < Publication > Book]; [institution, book] 0.95535

8. [Institution < Book]; [institution] 0.80613

9. [Journal < Paper]; [journal] 0.77643

(DS 29) ?x/Paper, published in, ?y/Year

0. [Paper > Ŷear]; [@publicationYear] 13.30359

1. [Article > Ŷear]; [@publicationYear] 6.61049

2. [Publication > Ŷear]; [@publicationYear] 5.03755

3. [Paper > D̂ate]; [@publicationYear] 4.23134

4. [Paper > Publication > Ŷear]; [cites, @publicationYear] 3.00775

5. [Paper < Paper > Ŷear]; [cites, @publicationYear] 3.00509

6. [Book > Ŷear]; [@publicationYear] 2.83122

7. [InBook > Ŷear]; [@publicationYear] 2.27447

8. [Publication < InBook > Ŷear]; [book, @publicationYear] 2.16422

9. [Article > D̂ate]; [@publicationYear] 2.09809

(DS 29) ?x/Paper, published in, ?y/When

0. [Paper > Ŷear]; [@publicationYear] 7.08111

1. [Paper > D̂ate]; [@publicationYear] 4.86619

2. [Article > Ŷear]; [@publicationYear] 3.51857

3. [Publication > Ŷear]; [@publicationYear] 2.66051

4. [Article > D̂ate]; [@publicationYear] 2.41288

5. [Publication > D̂ate]; [@publicationYear] 1.82844

6. [Paper > Publication > Ŷear]; [cites, @publicationYear] 1.60094

7. [Paper < Paper > Ŷear]; [cites, @publicationYear] 1.59952

8. [Book > Ŷear]; [@publicationYear] 1.50697

9. [InBook > Ŷear]; [@publicationYear] 1.21063

(DS 29) ?x/Paper, ,?y/publication year

0. [Paper > Ŷear]; [@publicationYear] 13.40771

1. [Article > Ŷear]; [@publicationYear] 6.66223

2. [Publication > Ŷear]; [@publicationYear] 5.03755

3. [Paper > D̂ate]; [@publicationYear] 4.62001

4. [Book > Ŷear]; [@publicationYear] 2.85338

5. [InBook > Ŷear]; [@publicationYear] 2.29227

6. [Article > D̂ate]; [@publicationYear] 2.29081

7. [Publication < InBook > Ŷear]; [book, @publicationYear] 1.91173

8. [Thesis > Ŷear]; [@publicationYear] 1.85737

9. [Publication > Publication > Ŷear]; [book, @publicationYear] 1.82149

(DS 30) ?x/Book, edited by, ?y/Person

0. [Book > Person]; [editor] 8.23730

1. [Publication > Person]; [editor] 5.81785

2. [Book > Person]; [author] 4.78207

3. [InBook > Book > Person]; [book, editor] 4.02377

4. [Book < Publication > Person]; [book, editor] 3.87447

5. [InBook > Person]; [author] 3.86479

6. [Book < Publication > Person]; [book, author] 3.76272

7. [Publication > Person]; [author] 3.71375

8. [Journal < Publication > Person]; [journal, editor] 3.13518

9. [InBook > Publication > Person]; [book, author] 2.82483

(DS 30) ?x/Book, editor, ?y/Person

0. [Book > Person]; [editor] 9.44968

1. [Publication > Person]; [editor] 6.67413

2. [Book > Person]; [author] 6.11349

3. [InBook > Person]; [author] 4.94082

4. [Book < Publication > Person]; [book, author] 4.81033

5. [Publication > Person]; [author] 4.74772

6. [InBook > Book > Person]; [book, editor] 4.61599

7. [Book < Publication > Person]; [book, editor] 4.44471
8. [InBook > Publication > Person]; [book, author] 3.61131
9. [Journal < Publication > Person]; [journal, editor] 3.49919

(DS 31) ?x/organization, publisher of, ?y/Proceedings

0. [Publisher < Proceedings]; [publisher] 3.35928
1. [Institution < Author < InProceedings]; [institution, author] 2.46621
2. [Publisher < Proceedings < InProceedings]; [publisher, proceedings] 2.28967
3. [Institution < Person < Proceedings]; [institution, editor] 2.27826
4. [Publisher < Publication > Proceedings]; [publisher, proceedings] 2.19704
5. [Institution < Person < InProceedings > Proceedings]; [institution, author, proceedings] 2.13894
6. [Publication > Proceedings]; [proceedings] 1.79456
7. [Publication > Publisher < Proceedings]; [publisher, publisher] 1.62941
8. [Institution < Person < Publication < InProceedings]; [institution, author, proceedings] 1.58863
9. [Institution < Author < Proceedings < InProceedings]; [institution, editor, proceedings] 1.44292

(DS 31) ?x/Company, published, ?y/Proceedings

0. [Publisher < Proceedings]; [publisher] 2.60259
1. [Publisher < Proceedings < InProceedings]; [publisher, proceedings] 1.77393
2. [Publisher < Publication > Proceedings]; [publisher, proceedings] 1.70216
3. [Institution < Person < Proceedings]; [institution, editor] 1.33000
4. [Institution < Person < InProceedings > Proceedings]; [institution, author, proceedings] 1.31210
5. [Institution < Author < InProceedings]; [institution, author] 1.25467
6. [Institution < Person < Publication < InProceedings]; [institution, author, proceedings] 0.97452
7. [Institution < Author < Proceedings < InProceedings]; [institution, editor, proceedings] 0.97435
8. [Institution < Publication < InProceedings]; [institution, cites] 0.66236
9. [Publisher < Book < Publication > Proceedings]; [publisher, book, proceedings] 0.65965

(IS 1) ?x/Conference, , ?y/Year

0. [Conference < Paper > Ŷear]; [conference, @publicationYear] 9.98270
1. [Conference < Paper > Ŷear]; [venue, @publicationYear] 3.81700
2. [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 3.21453
3. [Conference < Paper > D̂ate]; [conference, @publicationYear] 3.13015
4. [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 3.01381
5. [Publication > Ŷear]; [@publicationYear] 1.96529
6. [Conference < Paper > D̂ate]; [venue, @publicationYear] 1.19685
7. [Article > Ŷear]; [@publicationYear] 1.18489
8. [Conference < InProceedings > Publication > D̂ate]; [conference, proceedings, @publicationYear] 0.98267
9. [Conference < Publication < Paper > D̂ate]; [conference, proceedings, @publicationYear] 0.91619

(IS 1) ?x/Conference, held in, ?y/Year

0. [Conference < Paper > Ŷear]; [conference, @publicationYear] 9.98270
1. [Conference < Paper > Ŷear]; [venue, @publicationYear] 3.81700
2. [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 3.21453
3. [Conference < Paper > D̂ate]; [conference, @publicationYear] 3.13015
4. [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 3.01381
5. [Publication > Ŷear]; [@publicationYear] 1.96529
6. [Conference < Paper > D̂ate]; [venue, @publicationYear] 1.19685
7. [Article > Ŷear]; [@publicationYear] 1.18489
8. [Conference < InProceedings > Publication > D̂ate]; [conference, proceedings, @publicationYear] 0.98267
9. [Conference < Publication < Paper > D̂ate]; [conference, proceedings, @publicationYear] 0.91619

(IS 1) ?x/Conference, held in, ?y/Time

0. [Conference < Paper > Ŷear]; [conference, @publicationYear] 2.63022
1. [Conference < Paper > D̂ate]; [conference, @publicationYear] 1.78192
2. [Conference < Paper > Ŷear]; [venue, @publicationYear] 1.15756
3. [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 1.07068
4. [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 1.00382
5. [Conference < Paper > D̂ate]; [venue, @publicationYear] 0.78422
6. [Conference < InProceedings > Publication > D̂ate]; [conference, proceedings, @publicationYear] 0.70717
7. [Conference < Publication < Paper > D̂ate]; [conference, proceedings, @publicationYear] 0.65933
8. [Publication > Ŷear]; [@publicationYear] 0.51781
9. [Publication > D̂ate]; [@publicationYear] 0.35587

(IS 1) ?x/Conference, held in, ?y/When

0. [Conference < Paper > Ŷear]; [conference, @publicationYear] 2.63022
1. [Conference < Paper > D̂ate]; [conference, @publicationYear] 1.78192
2. [Conference < Paper > Ŷear]; [venue, @publicationYear] 1.15756
3. [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 1.07068
4. [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 1.00382
5. [Conference < Paper > D̂ate]; [venue, @publicationYear] 0.78422
6. [Conference < InProceedings > Publication > D̂ate]; [conference, proceedings, @publicationYear] 0.70717
7. [Conference < Publication < Paper > D̂ate]; [conference, proceedings, @publicationYear] 0.65933
8. [Publication > Ŷear]; [@publicationYear] 0.51781
9. [Publication > D̂ate]; [@publicationYear] 0.35587

(IS 2) ?x/paper, in, ?y/series

0. [Publication > Series]; [series] 3.35417

1. [Book > Series]; [series] 2.02281

2. [Publication > Book > Series]; [book, series] 1.33199

3. [Publication > Series < Publication]; [series, series] 1.19174

4. [Paper > Proceedings > Series]; [proceedings, series] 0.95308

5. [Paper > Person < Publication > Series]; [author, firstAuthor, series] 0.93313

6. [Book > Series < Publication]; [series, series] 0.84861

7. [InBook > Book > Series]; [book, series] 0.80061

8. [Paper > Person < Publication > Series]; [secondAuthor, firstAuthor, series] 0.75389

9. [Paper > Person < Publication > Series]; [firstAuthor, secondAuthor, series] 0.75254

(IS 2) ?x/InProceedings, in, ?y/series

0. [InProceedings > Proceedings > Series]; [proceedings, series] 10.35559

1. [InProceedings > Publication]; [proceedings] 3.00517

2. [InProceedings < Publication]; [proceedings] 1.41787

3. [InProceedings > Proceedings < Publication]; [proceedings, proceedings] 1.30974

4. [InProceedings > Proceedings > N̂umber]; [proceedings, @numberOfPublications] 0.65944

5. [InProceedings > N̂umber]; [@pageNumbers] 0.45561

6. [InProceedings > N̂umber]; [@numberOfCitations] 0.44320

7. [InProceedings > Publication > N̂umber]; [proceedings, @volumeNumber] 0.38715

8. [InProceedings > Publication > N̂umber]; [proceedings, @issueNumber] 0.36170

9. [InProceedings > Author]; [firstAuthor] 0.34966

(IS 3) ?x/Scholar, cites, ?y/Paper

0. [Author < Paper > Paper]; [author, cites] 1.97772

1. [Publication > Paper]; [cites] 1.88232

2. [Author < Paper > Article]; [author, cites] 0.97118

3. [Author < Publication > Paper]; [editor, cites] 0.95627

4. [Editor < Paper > Paper]; [author, cites] 0.95532

5. [Author < Paper]; [author] 0.93469

6. [Publication > Article]; [cites] 0.93416

7. [Author < Paper > Publication]; [author, cites] 0.74342

8. [Journal < Paper > Paper]; [journal, cites] 0.73164

9. [Publication > Publication]; [cites] 0.70733

(IS 3) ?x/Scholar, references, ?y/Paper

0. [Author < Paper > Paper]; [author, cites] 1.71411

1. [Publication > Paper]; [cites] 1.57412

2. [Author < Paper]; [author] 1.34231

3. [Author < Paper > Article]; [author, cites] 0.84173

4. [Author < Publication > Paper]; [editor, cites] 0.82881

5. [Editor < Paper > Paper]; [author, cites] 0.82798

6. [Publication > Article]; [cites] 0.78120

7. [Author < Article]; [author] 0.66515

8. [Editor < Paper]; [author] 0.66352

9. [Author < Paper > Publication]; [author, cites] 0.64433

(IS 3) ?x/Researcher, cites, ?y/Paper

0. [Author < Paper > Paper]; [author, cites] 1.59587

1. [Publication > Paper]; [cites] 1.32663

2. [Article > Paper]; [cites] 1.27233

3. [Editor < Paper > Paper]; [author, cites] 0.82312

4. [Author < Paper]; [author] 0.80132

5. [Author < Paper > Article]; [author, cites] 0.78367

6. [Author < Publication > Paper]; [editor, cites] 0.70812

7. [Publication > Article]; [cites] 0.65838

8. [Article > Article]; [cites] 0.63719

9. [Author < Paper > Publication]; [author, cites] 0.59988

(IS 3) ?x/Researcher, references, ?y/Paper

0. [Author < Paper > Paper]; [author, cites] 1.38316

1. [Author < Paper]; [author] 1.15079

2. [Publication > Paper]; [cites] 1.10941

3. [Article > Paper]; [cites] 1.06400

4. [Editor < Paper > Paper]; [author, cites] 0.71341

5. [Author < Paper > Article]; [author, cites] 0.67922

6. [Author < Publication > Paper]; [editor, cites] 0.61373

7. [Editor < Paper]; [author] 0.60741

8. [Author < Article]; [author] 0.57024

9. [Publication > Article]; [cites] 0.55058

(IS 3) ?x/Person, cites, ?y/Paper

0. [Person < Paper > Paper]; [author, cites] 6.77972

1. [Person < Publication > Paper]; [editor, cites] 3.76546

2. [Person < Paper > Article]; [author, cites] 3.32926

3. [Person < Paper]; [author] 3.13698

4. [Person < Paper > Publication]; [author, cites] 2.54847

5. [Author < Paper > Paper]; [author, cites] 1.79710

6. [Person < Publication > Article]; [editor, cites] 1.79061

7. [Person < Article]; [author] 1.55452

8. [Person < Paper > Book]; [author, cites] 1.54648

9. [Person < Publication > Publication]; [editor, cites] 1.41691

(IS 3) ?x/Person, references, ?y/Paper

0. [Person < Paper > Paper]; [author, cites] 5.87606

1. [Person < Paper]; [author] 4.50496

2. [Person < Publication > Paper]; [editor, cites] 3.26357

3. [Person < Paper > Article]; [author, cites] 2.88551

4. [Person < Article]; [author] 2.23243

5. [Person < Paper > Publication]; [author, cites] 2.20879

6. [Person < Publication]; [author] 1.69080

7. [Author < Paper > Paper]; [author, cites] 1.55756

8. [Person < Publication > Article]; [editor, cites] 1.55194

9. [Person < Paper > Book]; [author, cites] 1.34035

(IS 3) ?x/Scientist, cites, ?y/Paper

0. [Author < Paper > Paper]; [author, cites] 1.32360

1. [Author < Paper]; [author] 0.80352

2. [Author < Paper > Article]; [author, cites] 0.64997

3. [Author < Publication > Paper]; [editor, cites] 0.62587

4. [Editor < Paper > Paper]; [author, cites] 0.60166

5. [Author < Paper > Publication]; [author, cites] 0.49754

6. [Author < Article]; [author] 0.39814

7. [Editor < Paper]; [author] 0.37375

8. [Editor < Publication > Paper]; [editor, cites] 0.32174

9. [Author < Paper > Book]; [author, cites] 0.30192

(IS 4) ?x/Person, has, ?y/citations

0. [Person < Paper > Âbstract]; [author, @abstract] 1.01448

1. [Person < Paper > Paper > Âbstract]; [author, cites, @abstract] 0.84410

2. [Person < Paper < Paper > Âbstract]; [author, cites, @abstract] 0.84410

3. [Person < Paper > Publication]; [author, cites] 0.75157

4. [Person < Paper < Publication]; [author, cites] 0.75054

5. [Person < Paper > Article]; [author, cites] 0.69642

6. [Person < Paper < Article]; [author, cites] 0.69128

7. [Person < Publication]; [author] 0.53495

8. [Person < Publication > Âbstract]; [editor, @abstract] 0.52443

9. [Person < Article]; [author] 0.50038

(IS 4) ?x/Scholar, has, ?y/citations

0. [Author < Paper > Âbstract]; [author, @abstract] 0.29594

1. [Author < Paper < Paper > Âbstract]; [author, cites, @abstract] 0.24711

2. [Author < Paper > Paper > Âbstract]; [author, cites, @abstract] 0.24711

3. [Publication > Âbstract]; [@abstract] 0.24553

4. [Author < Paper > Publication]; [author, cites] 0.21924

5. [Author < Paper < Publication]; [author, cites] 0.21894

6. [Author < Paper > Article]; [author, cites] 0.20315

7. [Author < Paper < Article]; [author, cites] 0.20165

8. [Publication < Paper > Âbstract]; [cites, @abstract] 0.19617

9. [Publication > Paper > Âbstract]; [cites, @abstract] 0.19585

(IS 5) ?x/Person, cites, ?y/Person

0. [Person < Paper > Paper > Person]; [author, cites, author] 3.25519

1. [Person < Paper > Paper > Ŝubject]; [author, cites, @subject] 1.74765

2. [Person < Paper > Person]; [author, author] 1.72287

3. [Person < Paper > Publication > Person]; [author, cites, editor] 1.63906

4. [Person < Publication > Paper > Person]; [editor, cites, author] 1.63426

5. [Person < Publication > Person]; [author, editor] 0.96593

6. [Person < Paper > Ŝubject]; [author, @subject] 0.94336

7. [Author < Paper > Paper > Person]; [author, cites, author] 0.86285

8. [Person < Paper > Paper > Author]; [author, cites, author] 0.86285

9. [Person < Publication > Paper > Ŝubject]; [editor, cites, @subject] 0.80810

(IS 5) ?x/Scholar, cites, ?y/Scholar

0. [Author < Paper > Paper > Author]; [author, cites, author] 0.27897

1. [Author < Paper > Publication]; [author, cites] 0.21273

2. [Publication > Paper > Author]; [cites, author] 0.21244

3. [Publication > Publication]; [cites] 0.20240

4. [Author < Paper > Author]; [author, author] 0.14976

5. [Author < Paper > Publication > Editor]; [author, cites, author] 0.13039

6. [Editor < Paper > Paper > Author]; [author, cites, author] 0.13039

7. [Author < Paper > Publication > Author]; [author, cites, editor] 0.12515

8. [Author < Publication > Paper > Author]; [editor, cites, author] 0.12478

9. [Journal < Paper > Paper > Author]; [journal, cites, author] 0.10438

(IS 5) ?x/Researcher, cites, ?y/Researcher

0. [Author < Paper > Paper > Author]; [author, cites, author] 0.18535

1. [Author < Paper > Publication]; [author, cites] 0.12098

2. [Publication > Paper > Author]; [cites, author] 0.12082

3. [Author < Paper > Article]; [author, cites] 0.11535

4. [Article > Paper > Author]; [cites, author] 0.11450

5. [Author < Paper > Author]; [author, author] 0.10360

6. [Publication > Publication]; [cites] 0.10054

7. [Publication > Article]; [cites] 0.09691

8. [Article > Publication]; [cites] 0.09644

9. [Article > Article]; [cites] 0.09379

(IS 6) ?x/Paper, ISBN, ?y/Number

0. [Publication > N̂umber]; [@ISBN] 4.63037

1. [Book > N̂umber]; [@ISBN] 3.48404

2. [Publication > Book > N̂umber]; [book, @ISBN] 1.66634

3. [InBook > Book > N̂umber]; [book, @ISBN] 1.02392

4. [Paper > Proceedings > N̂umber]; [proceedings, @ISBN] 0.99272

5. [Paper > Venue < Publication > N̂umber]; [journal, conference, @ISBN] 0.77188

6. [Paper > Venue < Publication > N̂umber]; [conference, journal, @ISBN] 0.76959

7. [Paper > Editor < Proceedings > N̂umber]; [author, editor, @ISBN] 0.74846

8. [Journal < Publication > N̂umber]; [journal, @ISBN] 0.53133

9. [Paper > Book > N̂umber]; [cites, @ISBN] 0.45114

(IS 7) ?x/Author, co-author, ?y/Author

0. [Author < Paper > Author]; [author, author] 11.46656

1. [Author < Publication > Author]; [author, editor] 4.81273

2. [Author < Publication > Author]; [editor, author] 4.81273

3. [Author < Publication > Publication > Author]; [author, book, author] 3.30504

4. [Author < Publication < Publication > Author]; [author, book, author] 3.30504

5. [Author < Paper > Person]; [author, author] 3.03943

6. [Person < Paper > Author]; [author, author] 3.03943

7. [Article > Author]; [author] 2.77575

8. [Author < Article]; [author] 2.77565

9. [Author < Paper > Editor]; [author, author] 2.63355

(IS 7) ?x/Author, co-author, ?y/Person

0. [Author < Paper > Person]; [author, author] 11.46656

1. [Author < Publication > Person]; [author, editor] 4.81273

2. [Author < Publication > Person]; [editor, author] 4.81273

3. [Author < Paper > Ŝubject]; [author, @subject] 3.35288

4. [Author > Ŝubject]; [@subject] 3.23815
5. [Person < Paper > Person]; [author, author] 3.03943
6. [Author < Paper > Author]; [author, author] 3.03943
7. [Author < Proceedings > Person]; [editor, editor] 3.01176
8. [Article > Person]; [author] 2.77575
9. [Author < Publication < Publication > Person]; [author, book, author] 2.77469

(IS 7) ?x/Person, co-author, ?y/Person

0. [Person < Paper > Person]; [author, author] 9.79641
1. [Person < Publication > Person]; [author, editor] 4.81273
2. [Person < Publication > Person]; [editor, author] 4.81273
3. [Person < Paper > Ŝubject]; [author, @subject] 3.35288
4. [Person < Proceedings > Person]; [editor, editor] 3.01176
5. [Person < Publication < Publication > Person]; [author, book, author] 2.77469
6. [Author < Paper > Person]; [author, author] 2.59673
7. [Person < Paper > Author]; [author, author] 2.59673
8. [Editor < Paper > Person]; [author, author] 1.75563
9. [Person < Paper > Editor]; [author, author] 1.75563

(IS 7) ?x/Person, co-author, ?y/Author

0. [Person < Paper > Author]; [author, author] 11.46656
1. [Person < Publication > Author]; [author, editor] 4.81273
2. [Person < Publication > Author]; [editor, author] 4.81273
3. [Person < Publication > Publication > Author]; [author, book, author] 3.30504
4. [Person < Publication < Publication > Author]; [author, book, author] 3.30504
5. [Author < Paper > Author]; [author, author] 3.03943
6. [Person < Paper > Person]; [author, author] 3.03943
7. [Person < Article]; [author] 2.77565
8. [Person < Paper > Editor]; [author, author] 2.63355
9. [Editor < Paper > Author]; [author, author] 2.05494

(IS 7) ?x/Scholar, co-author, ?y/Person

0. [Author < Paper > Person]; [author, author] 2.91913
1. [Publication > Person]; [author] 1.47131
2. [Author < Publication > Person]; [author, editor] 1.43409
3. [Author < Publication > Person]; [editor, author] 1.43409
4. [Editor < Paper > Person]; [author, author] 1.41356
5. [Author < Paper > Ŝubject]; [author, @subject] 0.99909
6. [Author > Ŝubject]; [@subject] 0.96490

7. [Author < Proceedings > Person]; [editor, editor] 0.89744

8. [Author < Publication < Publication > Person]; [author, book, author] 0.82680

9. [Editor < Publication > Person]; [editor, author] 0.78340

(IS 7) ?x/Author, has ,?y/co-author

0. [Author < Paper > Author]; [author, author] 10.55944

1. [Author < Publication > Author]; [author, editor] 4.36718

2. [Author < Publication > Author]; [editor, author] 4.36718

3. [Person < Paper > Author]; [author, author] 3.03943

4. [Author < Paper > Person]; [author, author] 2.95577

5. [Author < Article]; [author] 2.95349

6. [Author < Publication > Publication > Author]; [author, book, author] 2.82199

7. [Author < Publication < Publication > Author]; [author, book, author] 2.82199

8. [Article > Author]; [author] 2.77575

9. [Editor < Paper > Author]; [author, author] 2.63355

(IS 7) ?x/Author, worked with, ?y/Author

0. [Author < Publication > Publication > Author]; [author, book, author] 1.98554

1. [Author < Publication < Publication > Author]; [author, book, author] 1.98554

2. [Author < Paper > Author]; [author, author] 1.62310

3. [Author < Publication > Author]; [author, editor] 1.00721

4. [Author < Publication > Author]; [editor, author] 1.00721

5. [Book > Author]; [author] 0.76432

6. [Author < Book]; [author] 0.76422

7. [Author < Publication > Book]; [author, book] 0.60140

8. [Book < Publication > Author]; [book, author] 0.60140

9. [Author < Publication < Publication > Person]; [author, book, author] 0.52630

(IS 7) ?x/Person, worked with, ?y/Person

0. [Person < Paper > Person]; [author, author] 1.17250

1. [Person > Ŝubject]; [@subject] 0.86800

2. [Person < Publication < Publication > Person]; [author, book, author] 0.81257

3. [Person < Publication > Publication > Person]; [author, book, author] 0.81257

4. [Person < Publication > Person]; [author, editor] 0.65736

5. [Person < Paper > Ŝubject]; [author, @subject] 0.58311

6. [Person < Publication > Person]; [editor, author] 0.56968

7. [Person < Publication < InBook > Ŝubject]; [author, book, @subject] 0.41332

8. [Person < Publication > Publication > Ŝubject]; [author, book, @subject] 0.39527

9. [Person < Paper > Author > Ŝubject]; [author, author, @subject] 0.37652

(IS 7) ?x/Scholar, worked with, ?y/Scholar

0. [Author < Paper > Author]; [author, author] 0.10192

1. [Author < Publication > Publication > Author]; [author, book, author] 0.06964

2. [Author < Publication < Publication > Author]; [author, book, author] 0.06964

3. [Publication > Author]; [author] 0.06206

4. [Author < Publication]; [author] 0.06196

5. [Publication < Publication > Author]; [book, author] 0.05319

6. [Publication > Publication > Author]; [book, author] 0.05319

7. [Author < Publication > Publication]; [author, book] 0.05319

8. [Author < Publication < Publication]; [author, book] 0.05319

9. [Author < Publication > Author]; [author, editor] 0.04975

(IS 8) ?x/Country, published paper in , ?y/Journal

0. [Country < Institution < Publication]; [country, institution] 1.90651

1. [Country < Institution < Paper > Journal]; [country, institution, journal] 1.27659

2. [Country < Institution < Person < Publication]; [country, institution, author] 0.91780

3. [Country < Institution < Article]; [country, institution] 0.59967

4. [Country < Institution < Paper]; [country, institution] 0.51156

5. [Country < Institution < Paper > Publication]; [country, institution, cites] 0.48590

6. [Country < Institution < Publication < Publication]; [country, institution, cites] 0.48495

7. [Country < Institution < Person < Article]; [country, institution, author] 0.35008

8. [Institution < Publication]; [institution] 0.31357

9. [Country < Institution < Publication > Book]; [country, institution, book] 0.28434

(IS 8) ?x/Country, has article in , ?y/Journal

0. [Country < Institution < Paper > Journal]; [country, institution, journal] 1.26464

1. [Country < Institution < Publication]; [country, institution] 1.25167

2. [Country < Institution < Person < Publication]; [country, institution, author] 1.00622

3. [Country < Institution < Article]; [country, institution] 0.76895

4. [Country < Institution < Paper > Publication]; [country, institution, cites] 0.69967

5. [Country < Institution < Publication < Publication]; [country, institution, cites] 0.69829

6. [Country < Institution < Paper]; [country, institution] 0.46668

7. [Country < Institution < Person < Article]; [country, institution, author] 0.38380

8. [Country < Institution < Person < Publication]; [country, institution, editor] 0.29420

9. [Country < Institution < Paper > Book]; [country, institution, cites] 0.27363

(IS 9) ?x/Country, published paper in , ?y/Conference

0. [Country < Institution < InProceedings > Conference]; [country, institution, conference] 0.68736

1. [Country < Institution < Publication]; [country, institution] 0.43026

2. [Country < Institution < InProceedings > Conference]; [country, institution, venue] 0.25522

3. [Institution < Author < InProceedings > Conference]; [institution, author, conference] 0.22694

4. [Country < Institution < Paper > Journal]; [country, institution, journal] 0.21264

5. [Country < Institution < Person < Publication]; [country, institution, author] 0.20713

6. [Country < Institution < Article]; [country, institution] 0.19705

7. [Institution < InProceedings > Conference]; [institution, conference] 0.12009

8. [Country < Institution < Person < Article]; [country, institution, author] 0.11503

9. [Country < Institution < Paper > Publication]; [country, institution, cites] 0.10966

(IS 9) ?x/Country, has inProceedings in , ?y/Conference

0. [Country < Institution < InProceedings > Conference]; [country, institution, conference] 0.99310

1. [Country < Institution < InProceedings > Conference]; [country, institution, venue] 0.49431

2. [Institution < Publication < InProceedings > Conference]; [institution, proceedings, conference] 0.38174

3. [Institution < InProceedings > Publication > Conference]; [institution, proceedings, conference] 0.36692

4. [Country < Institution < InProceedings > Publication]; [country, institution, proceedings] 0.33661

5. [Country < Institution < Publication < Publication]; [country, institution, proceedings] 0.31894

6. [Country < Institution < InProceedings > Proceedings]; [country, institution, proceedings] 0.19003

7. [Institution < InProceedings > Conference]; [institution, conference] 0.18675

8. [Institution < Publication < InProceedings > Conference]; [institution, proceedings, venue] 0.16390

9. [Institution < InProceedings > Publication > Conference]; [institution, proceedings, venue] 0.15714

(IS 10) ?x/Author, ,?y/Country

0. [Author > Institution > Country]; [institution, country] 2.86154

1. [Author < Paper > Institution > Country]; [author, institution, country] 2.79988

2. [Person < Paper > Institution > Country]; [author, institution, country] 0.74216

3. [Author < Publication > Institution > Country]; [editor, institution, country] 0.64524

4. [Editor < Paper > Institution > Country]; [author, institution, country] 0.56274

5. [Article > Author > Institution > Country]; [author, institution, country] 0.51111

6. [Author > Institution]; [institution] 0.49690

7. [Person > Institution > Country]; [institution, country] 0.45950

8. [Author < Paper > Institution]; [author, institution] 0.45439

9. [Publication > Author > Institution > Country]; [author, institution, country] 0.40723

(IS 10) ?x/Scholar, ,?y/Country

0. [Author > Institution > Country]; [institution, country] 0.70553

1. [Institution > Country]; [country] 0.53138

2. [Author < Paper > Institution > Country]; [author, institution, country] 0.43501

3. [Editor > Institution > Country]; [institution, country] 0.28204

4. [Publication > Institution > Country]; [institution, country] 0.26611

5. [Institution > Country < Institution]; [country, country] 0.19844

6. [Editor < Paper > Institution > Country]; [author, institution, country] 0.18434

7. [Publication > Author > Institution > Country]; [author, institution, country] 0.15309

8. [Author > Institution]; [institution] 0.14890

9. [Journal < Paper > Institution > Country]; [journal, institution, country] 0.14576

(IS 10) ?x/Researcher, ,?y/Country

0. [Author > Institution > Country]; [institution, country] 0.41926

1. [Author < Paper > Institution > Country]; [author, institution, country] 0.35209

2. [Editor < Paper > Institution > Country]; [author, institution, country] 0.15932

3. [Author > Institution]; [institution] 0.12694

4. [Editor > Institution > Country]; [institution, country] 0.10806

5. [Publication > Author > Institution > Country]; [author, institution, country] 0.10186

6. [Journal < Paper > Institution > Country]; [journal, institution, country] 0.09564

7. [Article > Author > Institution > Country]; [author, institution, country] 0.09057

8. [Author < Publication > Institution > Country]; [editor, institution, country] 0.08949

9. [Publication > Institution > Country]; [institution, country] 0.06605

(IS 10) ?x/Person, ,?y/Country

0. [Person > Institution > Country]; [institution, country] 2.86154

1. [Person < Paper > Institution > Country]; [author, institution, country] 1.47550

2. [Person > Institution]; [institution] 0.49690

3. [Author > Institution > Country]; [institution, country] 0.45950

4. [Author < Paper > Institution > Country]; [author, institution, country] 0.39111

5. [Person < Paper > Institution]; [author, institution] 0.26901

6. [Editor < Paper > Institution > Country]; [author, institution, country] 0.23140

7. [Editor > Institution > Country]; [institution, country] 0.17351

8. [Person < Publication > Institution]; [editor, institution] 0.13439

9. [Author > Institution]; [institution] 0.13171

(IS 11) ?x/InProceedings, has, ?y/Publisher

0. [InProceedings > Proceedings > Publisher]; [proceedings, publisher] 10.98514

1. [InProceedings > Publication]; [proceedings] 3.19518

2. [InProceedings > Publication > Book > Publisher]; [proceedings, book, publisher] 2.32548

3. [InProceedings > Author]; [author] 1.92139

4. [InProceedings > Editor]; [author] 1.81939

5. [InProceedings > Proceedings > Editor]; [proceedings, editor] 1.59566

6. [InProceedings > Person]; [author] 1.53563

7. [InProceedings > Publication > Author]; [proceedings, author] 1.53547

8. [InProceedings > Proceedings > Author]; [proceedings, editor] 1.52761

9. [InProceedings < Publication]; [proceedings] 1.50752

(IS 12) ?x/Book, cited by, ?y/Person

0. [Book < Paper > Person]; [cites, author] 5.03182

1. [Publication < Paper > Person]; [cites, author] 3.49278

2. [InBook < Paper > Person]; [cites, author] 3.31750

3. [Book < Publication < Paper > Person]; [book, cites, author] 3.22343

4. [InBook > Book < Paper > Person]; [book, cites, author] 2.98736

5. [Book < Paper > Ŝubject]; [cites, @subject] 2.91511

6. [Book < Publication > Person]; [cites, editor] 2.31313

7. [Journal < Paper < Paper > Person]; [journal, cites, author] 2.30479

8. [Article < Paper > Person]; [cites, author] 2.21305

9. [Publication < Paper > Ŝubject]; [cites, @subject] 2.08720

(IS 12) ?x/Book, referenced by, ?y/Person

0. [Book < Paper > Person]; [cites, author] 4.36114

1. [Publication < Paper > Person]; [cites, author] 3.02724

2. [Book > Person]; [author] 2.98410

3. [InBook < Paper > Person]; [cites, author] 2.87531

4. [Book < Publication < Paper > Person]; [book, cites, author] 2.86119

5. [InBook > Book < Paper > Person]; [book, cites, author] 2.65165

6. [Book < Paper > Ŝubject]; [cites, @subject] 2.52656

7. [InBook > Person]; [author] 2.41170

8. [Publication > Person]; [author] 2.31744

9. [Journal < Paper < Paper > Person]; [journal, cites, author] 2.04579

(IS 13) ?x/Proceedings, author, ?y/Person

0. [Proceedings < InProceedings > Person]; [proceedings, author] 13.29546

1. [InProceedings > Person]; [author] 12.97152

2. [InProceedings > Publication > Person]; [proceedings, author] 9.91739

3. [Proceedings > Person]; [editor] 6.70728

4. [InProceedings > Proceedings > Person]; [proceedings, editor] 4.60179

5. [Proceedings < Publication > Person]; [proceedings, editor] 4.34601

6. [InProceedings < InProceedings > Person]; [proceedings, author] 3.57027

7. [Proceedings < InProceedings > Author]; [proceedings, author] 3.52422

8. [InProceedings > Author]; [author] 3.43835

9. [Proceedings < InProceedings > Paper > Person]; [proceedings, cites, author] 3.18988

(IS 13) ?x/Proceedings, ,?y/Author

0. [Proceedings < InProceedings > Author]; [proceedings, author] 13.29546

1. [InProceedings > Author]; [author] 12.16080

2. [InProceedings > Publication > Author]; [proceedings, author] 9.91739

3. [Proceedings > Author]; [editor] 6.70728

4. [InProceedings > Proceedings > Author]; [proceedings, editor] 5.24829

5. [Proceedings < Publication > Author]; [proceedings, editor] 4.95658

6. [InProceedings < InProceedings > Author]; [proceedings, author] 3.57027

7. [Proceedings < InProceedings > Person]; [proceedings, author] 3.52422

8. [InProceedings > Person]; [author] 3.43835

9. [Proceedings < InProceedings > Paper > Author]; [proceedings, cites, author] 3.18988

(IS 14) ?x/Person, cites, ?y/Proceedings

0. [Person < Paper > InProceedings > Proceedings]; [author, cites, proceedings] 5.31437

1. [Person < Paper > InProceedings]; [author, cites] 4.89732

2. [Person < Paper > Publication < InProceedings]; [author, cites, proceedings] 3.97537

3. [Person < Paper > Proceedings]; [author, cites] 3.09155

4. [Person < Publication > InProceedings]; [editor, cites] 2.66793

5. [Person < Publication > InProceedings > Proceedings]; [editor, cites, proceedings] 2.36077

6. [Person < InProceedings > Proceedings]; [author, proceedings] 2.34205

7. [Person < InProceedings]; [author] 2.28489

8. [Person < Publication > Publication < InProceedings]; [editor, cites, proceedings] 1.76196

9. [Author < Paper > InProceedings > Proceedings]; [author, cites, proceedings] 1.40868

(IS 15) ?x/Author, contributed to, ?y/Conference

0. [Author < InProceedings > Conference]; [author, conference] 4.50872

1. [Author < InProceedings > Conference]; [author, venue] 2.03187

2. [Author < Publication < InProceedings > Conference]; [author, proceedings, conference] 1.92043

3. [Author < InProceedings > Publication > Conference]; [author, proceedings, conference] 1.81574

4. [Author < Paper < InProceedings > Conference]; [author, cites, conference] 1.47758

5. [Person < InProceedings > Conference]; [author, conference] 1.19512

6. [Editor < InProceedings > Conference]; [author, conference] 1.02918

7. [Author < Publication]; [author] 0.87878

8. [Article > Person < InProceedings > Conference]; [author, author, conference] 0.75024

9. [Publication > Author < InProceedings > Conference]; [author, author, conference] 0.56927

(DM 1) ?x/Author, wrote, ?y/Book || ?x/Author, , ?z/Institution

0. [Author < Book]; [author] || [Author > Institution]; [institution] 11.15063

1. [Author < Book]; [author] || [Author < Paper > Institution]; [author, institution] 10.38419

2. [Author < Publication]; [author] || [Author > Institution]; [institution] 10.10083

3. [Author < InBook]; [author] || [Author > Institution]; [institution] 10.02429

4. [Author < Publication > Book]; [author, book] || [Author > Institution]; [institution] 9.89110

5. [Author < Publication]; [author] || [Author < Paper > Institution]; [author, institution] 9.40655

6. [Author < InBook]; [author] || [Author < Paper > Institution]; [author, institution] 9.33527

7. [Author < Publication > Book]; [author, book] || [Author < Paper > Institution]; [author, institution] 9.21123

8. [Author < Publication < InBook]; [author, book] || [Author > Institution]; [institution] 8.57018

9. [Author < Publication < InBook]; [author, book] || [Author < Paper > Institution]; [author, institution] 7.98110

(DM 1) ?x/Scholar, published, ?y/Book || ?x/Scholar, , ?z/Institution

0. [Author < Book]; [editor] || [Author > Institution]; [institution] 1.92535

1. [Author < Book]; [author] || [Author > Institution]; [institution] 1.84077

2. [Author < InBook]; [author] || [Author > Institution]; [institution] 1.65482

3. [Author < Publication]; [author] || [Author > Institution]; [institution] 1.62216

4. [Author < Publication]; [editor] || [Author > Institution]; [institution] 1.61805

5. [Author < Publication > Book]; [author, book] || [Author > Institution]; [institution] 1.55125

6. [Author < Book]; [editor] || [Author < Paper > Institution]; [author, institution] 1.50990

7. [Author < Book]; [author] || [Author < Paper > Institution]; [author, institution] 1.44357

8. [Author < Publication < InBook]; [author, book] || [Author > Institution]; [institution] 1.34409

9. [Author < Publication < InBook > Book]; [author, book, book] || [Author > Institution]; [institution] 1.32507

(DM 2) ?x/Paper, in, ?y/Book || ?y/Book, has, ?z/ISBN

0. [Publication > Book]; [book] || [Book > ÎSBN]; [@ISBN] 6.29470

1. [Paper > Author < Book]; [author, author] || [Book > ÎSBN]; [@ISBN] 5.92391

2. [Paper > Journal < Publication]; [journal, journal] || [Publication > ÎSBN]; [@ISBN] 5.53901

3. [Paper > Journal]; [journal] || [Journal < Publication > ÎSBN]; [journal, @ISBN] 4.58123

4. [Paper > Journal < Publication]; [journal, journal] || [Publication > Book > ÎSBN]; [book, @ISBN] 4.46756

5. [Book < InBook > Book]; [book, book] || [Book > ÎSBN]; [@ISBN] 4.39059

6. [InBook > Book]; [book] || [Book > ÎSBN]; [@ISBN] 4.36825

7. [Publication < InBook]; [book] || [InBook > Publication% > ÎSBN]; [book, @ISBN] 4.10106

8. [Paper > Author < Publication]; [author, author] || [Publication > ÎSBN]; [@ISBN] 4.04547

9. [Article > Journal < Publication]; [journal, journal] || [Publication > ÎSBN]; [@ISBN] 4.00316

(DM 3) ?x/Author, published, ?y/Book || ?y/Book, ,?z/ISBN

0. [Author < Book]; [author] || [Book > ÎSBN]; [@ISBN] 9.71610

1. [Author < Publication > Book]; [author, book] || [Book > ÎSBN]; [@ISBN] 8.61861

2. [Author < Book]; [editor] || [Book > ÎSBN]; [@ISBN] 7.77759

3. [Author < Publication]; [author] || [Publication > ÎSBN]; [@ISBN] 7.63573

4. [Author < Publication < InBook > Book]; [author, book, book] || [Book > ÎSBN]; [@ISBN] 6.78084

5. [Author < Publication]; [author] || [Publication > Book > ÎSBN]; [book, @ISBN] 6.15869

6. [Author < Book]; [author] || [Book < Publication > ÎSBN]; [book, @ISBN] 5.65369

7. [Author < InBook]; [author] || [InBook > Book > ÎSBN]; [book, @ISBN] 5.51858

8. [Author < Publication < InBook]; [author, book] || [InBook > Publication% > ÎSBN]; [book, @ISBN] 5.44796

9. [Author < Publication > Book]; [author, book] || [Book < Publication > ÎSBN]; [book, @ISBN] 5.01507

(DM 4) ?x/Paper, in, ?y/Book || ?y/Book, has, ?z/Publisher

0. [Publication > Book]; [book] || [Book > Publisher]; [publisher] 6.31402

1. [Paper > Author < Book]; [author, author] || [Book > Publisher]; [publisher] 5.94209

2. [Paper > Journal < Publication]; [journal, journal] || [Publication > Publisher]; [publisher] 5.30590

3. [Paper > Journal < Publication]; [journal, journal] || [Publication > Book > Publisher]; [book, publisher] 4.48687

4. [Book < InBook > Book]; [book, book] || [Book > Publisher]; [publisher] 4.40406

5. [InBook > Book]; [book] || [Book > Publisher]; [publisher] 4.38165

6. [Publication < InBook]; [book] || [InBook > Book > Publisher]; [book, publisher] 4.11878

7. [Publication > Book]; [book] || [Book < InBook > Book > Publisher]; [book, book, publisher] 3.99783

8. [Paper > Venue < Publication > Book]; [conference, journal, book] || [Book > Publisher]; [publisher] 3.92272

9. [Publication > Publisher < Book]; [publisher, publisher] || [Book > Publisher]; [publisher] 3.87990

(DM 5) ?x/Author, published, ?y/Book || ?y/Book, ,?z/Publisher

0. [Author < Book]; [author] || [Book > Publisher]; [publisher] 9.74592

1. [Author < Publication > Book]; [author, book] || [Book > Publisher]; [publisher] 8.64506

2. [Author < Book]; [editor] || [Book > Publisher]; [publisher] 7.80146

3. [Author < Publication]; [author] || [Publication > Publisher]; [publisher] 7.31437

4. [Author < Publication < InBook > Book]; [author, book, book] || [Book > Publisher]; [publisher] 6.80165

5. [Author < Publication]; [author] || [Publication > Book > Publisher]; [book, publisher] 6.18531

6. [Author < Book]; [author] || [Book < InBook > Book > Publisher]; [book, book, publisher] 6.17080

7. [Author < Book]; [author] || [Book < Publication > Publisher]; [book, publisher] 5.74270

8. [Author < InBook]; [author] || [InBook > Book > Publisher]; [book, publisher] 5.54243

9. [Author < Publication > Book]; [author, book] || [Book < InBook > Book > Publisher]; [book, book, publisher] 5.47378

(DM 6) ?x/Author, , ?y/Institution || ?x/Author, ,?z/research area

0. [Author > Institution]; [institution] || [Author > Ŝubject]; [@subject] 1.68546

1. [Author < Paper > Institution]; [author, institution] || [Author > Ŝubject]; [@subject] 1.56961

2. [Author > Institution]; [institution] || [Author < Paper > Institution > Country]; [author, institution, country] 1.23739

3. [Author < Paper > Institution]; [author, institution] || [Author < Paper > Institution > Country]; [author, institution, country] 1.15234

4. [Author > Institution]; [institution] || [Author > Institution > Country]; [institution, country] 1.03308

5. [Author < Paper > Institution]; [author, institution] || [Author > Institution > Country]; [institution, country] 0.96207

6. [Author < Publication > Institution]; [editor, institution] || [Author > Ŝubject]; [@subject] 0.93502

7. [Author > Institution]; [institution] || [Author < Paper > Author > Ŝubject]; [primaryAuthor, author, @subject] 0.90335

8. [Author > Institution]; [institution] || [Author < Paper > Ŝubject]; [author, @subject] 0.89939

9. [Author > Institution]; [institution] || [Author < Paper > Venue > Ŝubject]; [author, journal, @subject] 0.85304

(DM 7) ?x/Conference, has, ?y/Publication || ?y, citations, ?z/number

0. [Conference < Publication]; [conference] || [Publication > N̂umber]; [@numberOfCitations] 14.99416

1. [Conference < Publication]; [conference] || [Publication < Paper > N̂umber]; [cites, @numberOfCitations] 12.81916

2. [Conference < Publication]; [conference] || [Publication > Publication > N̂umber]; [cites, @numberOfCitations] 12.81594

3. [Conference < Publication]; [conference] || [Publication < Paper > N̂umber]; [cites, @pageNumbers] 12.71081

4. [Conference < Publication]; [conference] || [Publication > Publication > N̂umber]; [cites, @pageNumbers] 12.70221

5. [Conference < Publication > Publication]; [conference, book] || [Publication > N̂umber]; [@numberOfCitations] 9.51298

6. [Conference < Publication < Publication]; [conference, book] || [Publication > N̂umber]; [@numberOfCitations] 9.51298

7. [Conference < Publication > Publication]; [conference, book] || [Publication < Paper > N̂umber]; [cites, @numberOfCitations] 8.13305

8. [Conference < Publication < Publication]; [conference, book] || [Publication < Paper > N̂umber]; [cites, @numberOfCitations] 8.13305

9. [Conference < Publication > Publication]; [conference, book] || [Publication > Publication > N̂umber]; [cites, @numberOfCitations] 8.13101

(DM 8) ?x/paper, in, ?y/Journal || ?x, issue, ?z/number || ?x, volume, ?u/number

0. [Paper > Journal]; [journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > N̂umber]; [@volumeNumber] 13.75662

1. [Paper > Journal < Publication]; [journal, journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > N̂umber]; [@volumeNumber] 11.49154

2. [Paper > Journal]; [journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > Venue > N̂umber]; [journal, @numberOfCitations] 10.48481

3. [Paper > Journal]; [journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > Journal% > N̂umber]; [journal, @numberOfPublications] 10.35115

4. [Paper > Journal]; [journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > N̂umber]; [@pageNumbers] 9.50946

5. [Paper > Journal]; [journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > Person > N̂umber]; [author, @numberOfCitations] 9.05146

6. [Paper > Journal < Publication]; [journal, journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > Venue > N̂umber]; [journal, @numberOfCitations] 8.75844

7. [Paper > Journal < Article]; [journal, journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > N̂umber]; [@volumeNumber] 8.71356

8. [Paper > Journal < Publication]; [journal, journal] || [Paper > N̂umber]; [@issueNumber] || [Paper > Journal% > N̂umber]; [journal, @numberOfPublications] 8.64680

9. [Paper > Venue < Publication]; [journal, conference] || [Paper > N̂umber]; [@issueNumber] || [Paper > N̂umber]; [@volumeNumber] 7.96988

(DM 8) ?x/paper, in, ?y/Journal || ?y, issue, ?z/number || ?y, volume, ?u/number

0. [Paper > Journal]; [journal] || [Journal < Paper% > N̂umber]; [journal, @issueNumber] || [Journal < Paper% > N̂umber]; [journal, @volumeNumber] 11.50243

1. [Paper > Journal]; [journal] || [Journal < Paper% > N̂umber]; [journal, @issueNumber] || [Journal < Article > N̂umber]; [journal, @numberOfCitations] 10.64431

2. [Paper > Journal]; [journal] || [Journal < Paper% > N̂umber]; [journal, @issueNumber] || [Journal < Paper% > N̂umber]; [journal, @pageNumbers] 10.57988

3. [Paper > Journal]; [journal] || [Journal < Paper% > N̂umber]; [journal, @issueNumber] || [Journal < Article > N̂umber]; [journal, @issueNumber] 10.49336

4. [Paper > Journal]; [journal] || [Journal < Paper% > N̂umber]; [journal, @issueNumber] || [Journal > N̂umber]; [@numberOfPublications] 9.51256

5. [Article > Journal]; [journal] || [Journal < Article% > N̂umber]; [journal, @issueNumber] || [Journal < Article% > N̂umber]; [journal, @volumeNumber] 8.95678

6. [Paper > Journal]; [journal] || [Journal > N̂umber]; [@numberOfPublications] || [Journal < Paper% > N̂umber]; [journal, @volumeNumber] 8.79810

7. [Paper > Journal < Publication]; [journal, journal] || [Publication > N̂umber]; [@issueNumber] || [Publication > N̂umber]; [@volumeNumber] 8.61152

8. [Article > Journal]; [journal] || [Journal < Article% > N̂umber]; [journal, @issueNumber] || [Journal < Article > N̂umber]; [journal, @numberOfCitations] 8.28857

9. [Article > Journal]; [journal] || [Journal < Article% > N̂umber]; [journal, @issueNumber] || [Journal < Article% > N̂umber]; [journal, @pageNumbers] 8.23840

(IM 1) ?x/Editor, ,?y/Conference || ?y/Conference, ,?z/Year

0. [Editor < Publication > Conference]; [editor, conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 9.20292

1. [Editor < InProceedings > Conference]; [author, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 8.33921

2. [Editor < Proceedings < InProceedings > Conference]; [editor, proceedings, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 5.80497

3. [Editor < Publication > Conference]; [editor, conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 5.69066

4. [Editor < Publication > Conference]; [editor, venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 5.53610

5. [Editor < Publication > Conference]; [editor, conference] || [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 5.22229

6. [Editor < InProceedings > Conference]; [author, conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 5.15658

7. [Editor < Publication > Conference]; [editor, conference] || [Conference < Publication% > D̂ate]; [conference, @publicationYear] 5.15328

8. [Editor < Publication > Conference]; [editor, conference] || [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 5.05661

9. [Editor < InProceedings > Conference]; [author, venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 5.02127

(IM 2) ?x/Paper, ,?y/Conference || ?y/Conference, ,?z/Year

0. [Paper > Conference]; [conference] || [Conference < Paper% > Ŷear]; [conference, @publicationYear] 12.73360

1. [Paper > Conference]; [conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 7.87387

2. [Paper > Conference]; [conference] || [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 7.22581

3. [Paper > Conference]; [conference] || [Conference < Paper% > D̂ate]; [conference, @publicationYear] 7.13033

4. [Paper > Conference]; [conference] || [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 6.99657

5. [Publication > Conference]; [conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.81009

6. [Paper > Conference]; [venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 6.75401

7. [Paper > Publication > Conference]; [proceedings, conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.46782

8. [Paper > Conference]; [conference] || [Conference < Paper > D̂ate]; [venue, @publicationYear] 4.40907

9. [Publication > Conference]; [conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 4.21104

(IM 2) ?y/Conference, published, ?x/Paper || ?y/Conference, , ?z/Year

0. [Conference < Paper]; [conference] || [Conference < Paper% > Ŷear]; [conference, @publicationYear] 9.73419

1. [Conference < Publication]; [conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 7.80279

2. [Conference < Paper]; [conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 6.01917

3. [Conference < Paper]; [conference] || [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 5.52376

4. [Conference < Paper]; [conference] || [Conference < Paper% > D̂ate]; [conference, @publicationYear] 5.45078

5. [Conference < Paper]; [conference] || [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 5.34852

6. [Conference < Paper > Journal < Paper]; [conference, venue, journal] || [Conference < Paper% > Ŷear]; [conference, @publicationYear] 5.23358

7. [Conference < InProceedings > Venue < Paper]; [venue, conference, journal] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 5.22997

8. [Conference < Publication < Paper]; [conference, proceedings] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 5.21722

9. [Conference < Paper]; [venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 5.16303

(IM 3) ?x/Paper, cited by, ?y/Conference || ?y/Conference, ,?z/Year

0. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 11.45110

1. [Article < Paper > Conference]; [cites, conference] || [Conference < Paper% > Ŷear]; [conference, @publicationYear] 7.96992

2. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 7.08083

3. [Publication < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 7.02129

4. [Paper < InProceedings > Conference]; [cites, venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 6.89502

5. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 6.49804

6. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings% > D̂ate]; [conference, @publicationYear] 6.41218

7. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 6.29189

8. [Paper < Publication < InProceedings > Conference]; [cites, proceedings, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 6.19524

9. [Paper < InProceedings > Publication > Conference]; [cites, proceedings, conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.03300

(IM 3) ?x/Paper, referenced by, ?y/Conference || ?y/Conference, ,?z/Year

0. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 10.47174

1. [Article < Paper > Conference]; [cites, conference] || [Conference < Paper% > Ŷear]; [conference, @publicationYear] 7.28829

2. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 6.47524

3. [Publication < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 6.42079

4. [Paper < InProceedings > Conference]; [cites, venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 6.41908

5. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 5.94229

6. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < InProceedings% > D̂ate]; [conference, @publicationYear] 5.86378

7. [Paper < Publication < InProceedings > Conference]; [cites, proceedings, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 5.83677

8. [Paper < InProceedings > Conference]; [cites, conference] || [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 5.75378

9. [Paper < InProceedings > Publication > Conference]; [cites, proceedings, conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 5.68392

(IM 3) ?x/Paper, cited by, ?y/Venue || ?y, ,?z/Year

0. [Paper < Paper > Venue]; [cites, venue] || [Venue < Paper% > Ŷear]; [venue, @publicationYear] 11.90404

1. [Article < Paper > Venue]; [cites, venue] || [Venue < Paper% > Ŷear]; [venue, @publicationYear] 8.32212

2. [Publication < Paper > Venue]; [cites, venue] || [Venue < Paper% > Ŷear]; [venue, @publicationYear] 7.29897

3. [Paper < Paper > Venue]; [cites, venue] || [Venue < Paper > Ŷear]; [conference, @publicationYear] 7.20855

4. [Paper < InProceedings > Venue]; [cites, conference] || [Venue < Paper > Ŷear]; [venue, @publicationYear] 7.04075

5. [Paper < Paper > Venue]; [cites, venue] || [Venue < Paper% > D̂ate]; [venue, @publicationYear] 6.67348

6. [Book < Paper > Venue]; [cites, venue] || [Venue < Paper% > Ŷear]; [venue, @publicationYear] 5.61166

7. [Paper < Paper > Venue]; [cites, journal] || [Venue < Paper > Ŷear]; [venue, @publicationYear] 5.33787

8. [Article < Paper > Venue]; [cites, venue] || [Venue < Paper > Ŷear]; [conference, @publicationYear] 5.03951

9. [Article < Paper > Venue]; [cites, conference] || [Venue < Paper > Ŷear]; [venue, @publicationYear] 4.90033

(IM 3) ?x/Paper, cited by, ?y/Journal || ?y, ,?z/Year

0. [Paper < Paper > Journal]; [cites, journal] || [Journal < Paper% > Ŷear]; [journal, @publicationYear] 11.16179

1. [Paper < Publication]; [cites] || [Publication > Ŷear]; [@publicationYear] 9.95102

2. [Article < Article > Journal]; [cites, journal] || [Journal < Article% > Ŷear]; [journal, @publicationYear] 7.81021

3. [Article < Publication]; [cites] || [Publication > Ŷear]; [@publicationYear] 7.01018

4. [Paper < Publication]; [cites] || [Publication < InBook > Ŷear]; [book, @publicationYear] 6.74517

5. [Paper < Publication]; [cites] || [Publication > Publication > Ŷear]; [book, @publicationYear] 6.58405

6. [Paper < InBook > Publication]; [cites, book] || [Publication > Ŷear]; [@publicationYear] 6.44464

7. [Paper < Publication < Publication]; [cites, book] || [Publication > Ŷear]; [@publicationYear] 6.41403

8. [Paper < Paper > Journal]; [cites, journal] || [Journal < Paper% > D̂ate]; [journal, @publicationYear] 6.24621

9. [Publication < Publication]; [cites] || [Publication > Ŷear]; [@publicationYear] 6.09998

(IM 4) ?x/Author, ,?y/Conference || ?y/Conference, ,?z/Year

0. [Author < InProceedings > Conference]; [author, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 11.55617

1. [Author < InProceedings > Conference]; [author, conference] || [Conference < Paper > Ŷear]; [venue, @publicationYear] 7.14580

2. [Author < Publication > Conference]; [editor, conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 7.07527

3. [Author < InProceedings > Conference]; [author, venue] || [Conference < Paper > Ŷear]; [conference, @publicationYear] 6.95828

4. [Author < InProceedings > Conference]; [author, conference] || [Conference < InProceedings > Publication > Ŷear]; [conference, proceedings, @publicationYear] 6.55766

5. [Author < InProceedings > Conference]; [author, conference] || [Conference < InProceedings% > D̂ate]; [conference, @publicationYear] 6.47101

6. [Author < InProceedings > Conference]; [author, conference] || [Conference < Publication < Paper > Ŷear]; [conference, proceedings, @publicationYear] 6.34962

7. [Author < Publication < InProceedings > Conference]; [author, proceedings, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 6.29166

8. [Author < InProceedings > Publication > Conference]; [author, proceedings, conference] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.11777

9. [Person < InProceedings > Conference]; [author, conference] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 5.94968

(IM 5) ?x/Journal, ,?y/volume no. || ?x, ,?z/author

0. [Journal < Paper% > N̂umber]; [journal, @volumeNumber] || [Journal < Paper > Author]; [journal, author] 6.86567

1. [Publication > N̂umber]; [@volumeNumber] || [Publication > Author]; [author] 5.93130

2. [Journal > N̂umber]; [@numberOfPublications] || [Journal < Paper > Author]; [journal, author] 5.16350

3. [Publication > N̂umber]; [@numberOfPublications] || [Publication > Author]; [author] 4.39349

4. [Journal < Publication > N̂umber]; [journal, @numberOfPublications] || [Journal < Paper > Author]; [journal, author] 4.22214

5. [Journal < Publication% > N̂umber]; [journal, @volumeNumber] || [Journal < Publication > Author]; [journal, editor] 4.17793

6. [Publication > Publication > N̂umber]; [book, @volumeNumber] || [Publication > Author]; [author] 4.10470

7. [Publication < Publication > N̂umber]; [book, @volumeNumber] || [Publication > Author]; [author] 4.10470

8. [Journal < Paper% > N̂umber]; [journal, @pageNumbers] || [Journal < Paper > Author]; [journal, author] 3.83191

9. [Publication > N̂umber]; [@volumeNumber] || [Publication < Publication > Author]; [book, author] 3.82859

(IM 6) ?x/Editor, in, ?y/Conference || ?x/Editor, published, ?z/Paper || ?y, ,?u/Year

0. [Editor < Publication > Conference]; [editor, conference] || [Editor < Paper]; [author] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 8.26107

1. [Editor < InProceedings > Conference]; [author, conference] || [Editor < Paper]; [author] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 7.73575

2. [Editor < Publication > Conference]; [editor, conference] || [Editor < Publication]; [editor] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 7.44358

3. [Editor < InProceedings > Conference]; [author, conference] || [Editor < Publication]; [editor] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 6.97025

4. [Editor < Publication > Conference]; [editor, conference] || [Editor < Article]; [author] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.50018

5. [Editor < Publication > Conference]; [editor, conference] || [Editor < Publication]; [author] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.42801

6. [Editor < Publication > Conference]; [editor, conference] || [Editor < Paper]; [editor] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.39914

7. [Editor < Publication > Conference]; [editor, conference] || [Editor < Publication > Author < Paper]; [editor, author, author] || [Conference < Publication% > Ŷear]; [conference, @publicationYear] 6.23613

8. [Editor < InProceedings > Conference]; [author, conference] || [Editor < Article]; [author] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 6.08684

9. [Editor < Proceedings < InProceedings > Conference]; [editor, proceedings, conference] || [Editor < Paper]; [author] || [Conference < InProceedings% > Ŷear]; [conference, @publicationYear] 6.07602

(IM 7) ?x/Person, from, ?z/Organization || ?x/Person, has, ?y/co-author || ?y, work in, ?u/Institution

0. [Person > Institution]; [institution] || [Person < Paper > Author]; [author, author] || [Author > Institution]; [institution] 5.19247

1. [Person > Institution]; [institution] || [Person < Paper > Author]; [author, author] || [Author < Paper > Institution]; [author, institution] 4.95172

2. [Person > Institution]; [institution] || [Person < Paper > Author]; [author, author] || [Author < Publication > Publication > Institution]; [author, book, institution] 4.66343

3. [Person > Institution]; [institution] || [Person < Paper > Author]; [author, author] || [Author < Publication < Publication > Institution]; [author, book, institution] 4.66343

4. [Person > Institution]; [institution] || [Person < Paper > Author]; [author, author] || [Author < Paper > Person > Institution]; [author, author, institution] 4.49056

5. [Person > Institution]; [institution] || [Person < Publication > Author]; [author, editor] || [Author > Institution]; [institution] 4.09715

6. [Person > Institution]; [institution] || [Person < Publication > Author]; [editor, author] || [Author > Institution]; [institution] 4.09715

7. [Person > Institution]; [institution] || [Person < Publication > Author]; [author, editor] || [Author < Paper > Institution]; [author, institution] 3.90718

8. [Person > Institution]; [institution] || [Person < Publication > Author]; [editor, author] || [Author < Paper > Institution]; [author, institution] 3.90718

9. [Person < Paper > Institution]; [author, institution] || [Person < Paper > Author]; [author, author] || [Author > Institution]; [institution] 3.86970

(IM 8) ?x/Proceedings, author, ?y/Person || ?x/Proceedings, editor, ?z/Person

0. [Proceedings < InProceedings > Person]; [proceedings, author] || [Proceedings > Person]; [editor] 13.11757

1. [InProceedings > Person]; [author] || [InProceedings > Proceedings > Person]; [proceedings, editor] 10.73217

2. [Proceedings < InProceedings > Person]; [proceedings, author] || [Proceedings < Publication > Person]; [proceedings, editor] 10.55906

3. [Proceedings < InProceedings > Person]; [proceedings, author] || [Proceedings < InProceedings > Person]; [proceedings, author] 9.57138

4. [InProceedings > Publication > Person]; [proceedings, author] || [InProceedings > Proceedings > Person]; [proceedings, editor] 9.38406

5. [InProceedings > Person]; [author] || [InProceedings > Person]; [author] 9.33819

6. [Proceedings > Person]; [editor] || [Proceedings > Person]; [editor] 9.31697

7. [InProceedings > Person]; [author] || [InProceedings > Publication > Person]; [proceedings, author] 8.16518

8. [InProceedings > Publication > Person]; [proceedings, author] || [InProceedings > Person]; [author] 8.16518

9. [Proceedings > Person]; [editor] || [Proceedings < Publication > Person]; [proceedings, editor] 7.49975

(IM 9) ?y/paper, has, ?x/research areas || ?y, published by, ?z/company || ?y, published in, ?u/Year

0. [Paper > Ŝubject]; [@subject] || [Paper > Institution]; [institution] || [Paper > Ŷear]; [@publicationYear] 1.91997

1. [Paper > Ŝubject]; [@subject] || [Paper > Person > Institution]; [author, institution] || [Paper > Ŷear]; [@publicationYear] 1.73964

2. [Paper > Institution > Country]; [institution, country] || [Paper > Institution]; [institution] || [Paper > Ŷear]; [@publicationYear] 1.47947

3. [Paper > Ŝubject]; [@subject] || [Paper > Publication > Institution]; [cites, institution] || [Paper > Ŷear]; [@publicationYear] 1.40450

4. [Paper > Ŝubject]; [@subject] || [Paper < Paper > Institution]; [cites, institution] || [Paper > Ŷear]; [@publicationYear] 1.40429

5. [Paper > Institution > Country]; [institution, country] || [Paper > Person > Institution]; [author, institution] || [Paper > Ŷear]; [@publicationYear] 1.34051

6. [Paper > Ŝubject]; [@subject] || [Paper > Institution]; [institution] || [Paper > D̂ate]; [@publicationYear] 1.31058

7. [Paper > Ŝubject]; [@subject] || [Paper > Editor < Publication > Institution]; [author, editor, institution] || [Paper > Ŷear]; [@publicationYear] 1.25632

8. [Paper > Ŝubject]; [@subject] || [Paper > Book > Publisher]; [cites, publisher] || [Paper > Ŷear]; [@publicationYear] 1.20063

9. [Paper > Author > Ŝubject]; [author, @subject] || [Paper > Institution]; [institution] || [Paper > Ŷear]; [@publicationYear] 1.19173

(IM 10) ?y/paper, has, ?x/Venues || ?y, published by, ?z/Organization || ?y, published in, ?u/Year

0. [Paper > Venue]; [venue] || [Paper > Institution]; [institution] || [Paper > Ŷear]; [@publicationYear] 10.35954

1. [Paper > Venue]; [venue] || [Paper > Person > Institution]; [author, institution] || [Paper > Ŷear]; [@publicationYear] 8.97175

2. [Paper > Venue]; [venue] || [Paper > Institution]; [institution] || [Paper > D̂ate]; [@publicationYear] 7.07148

3. [Paper > Venue]; [journal] || [Paper > Institution]; [institution] || [Paper > Ŷear]; [@publicationYear] 6.96049

4. [Paper > Venue]; [venue] || [Paper > Publication > Institution]; [cites, institution] || [Paper > Ŷear]; [@publicationYear] 6.87055

5. [Paper > Venue]; [venue] || [Paper < Paper > Institution]; [cites, institution] || [Paper > Ŷear]; [@publicationYear] 6.86949

6. [Paper > Venue]; [conference] || [Paper > Institution]; [institution] || [Paper > Ŷear]; [@publicationYear] 6.71231

7. [Paper > Venue]; [venue] || [Paper > Editor < Publication > Institution]; [author, editor, institution] || [Paper > Ŷear]; [@publicationYear] 6.50264

8. [Paper > Venue]; [venue] || [Paper > Institution]; [institution] || [Paper > Publication > Ŷear]; [cites, @publicationYear] 6.31100

9. [Paper > Venue]; [venue] || [Paper > Institution]; [institution] || [Paper < Paper > Ŷear]; [cites, @publicationYear] 6.30913

Appendix C

TOP-10 INTERPRETATIONS OF 99 DBPEDIA TESTCASES

This appendix details top-10 interpretations of the 99 DBPedia testcases, which are produced using the hybrid similarity and the parameters learned from the DBLP+ dataset. The 33 natural language questions corresponding to the 99 testcases can be referenced in the Table 9.11.

Each query may contain one or more relations, which are separated by “||”. Each relation in the query is mapped to a schema path. The top-10 interpretations are numbered from 0 to 9. An interpretations is serialized into a string, following a specific syntax. Inside the first brackets are the classes on the schema path and the “<” or “>” between two classes shows the direction of the property connecting the classes. Inside the second brackets are the properties on the schema path, which are in order with “<” or “>” in the first brackets. The ending number at each interpretation line is the fitness score of the interpretation.

(DS 1) Brooklyn/Bridge, crosses, ?x/River

0. [Bridge > River]; [crosses] 9.38807
1. [Bridge > Stream]; [crosses] 6.33271
2. [Bridge > C̃ross < River]; [crosses, riverMouth] 5.10397
3. [Bridge > Stream > River]; [crosses, riverMouth] 4.82787
4. [Bridge > River]; [river] 4.14279

5. [Bridge > C̃ross < Stream]; [crosses, riverMouth] 3.44130

6. [Bridge > Stream > Stream]; [crosses, riverMouth] 3.25500

7. [Bridge > Lake]; [crosses] 3.16446

8. [Bridge > Stream]; [river] 2.79263

9. [Bridge > T̃ributary]; [crosses] 2.51006

(DS 2) Abraham Lincoln/Person, died, ?x/Place

0. [Person > Place]; [deathPlace] 10.29601

1. [Person > PopulatedPlace]; [deathPlace] 8.02293

2. [MilitaryPerson > Place]; [deathPlace] 6.59922

3. [Person > L̃ocation]; [deathPlace] 6.59287

4. [Person > Place > Place]; [deathPlace, location] 6.46622

5. [MilitaryPerson > PopulatedPlace]; [deathPlace] 5.14085

6. [Person > R̃egion < Place]; [deathPlace, region] 5.07034

7. [Person > B̃attle > Place]; [deathPlace, place] 4.74140

8. [Person > R̃egion]; [deathPlace] 4.68545

9. [Person > Place > PopulatedPlace]; [deathPlace, location] 4.35680

(DS 3) Obama/President, wife, ?x/Person

0. [President > Person]; [spouse] 5.70799

1. [President < Person]; [spouse] 5.63143

2. [President < Person]; [president] 5.05304

3. [President > Person]; [president] 4.11711

4. [P̃resenter < Person]; [spouse] 3.82739

5. [P̃resenter > Person]; [spouse] 3.76419

6. [D̃irector < Person]; [spouse] 3.64943

7. [D̃irector > Person]; [spouse] 3.57920

8. [President > S̃pouse < Person]; [spouse, spouse] 3.14995

9. [Governor > Person]; [spouse] 2.54065

(DS 4) Nile/River, starts, ?x/Country

0. [Stream > Settlement > Country]; [startPoint, country] 1.49303

1. [River > Country]; [sourceCountry] 1.29701

2. [River > Place > Country]; [sourcePlace, country] 1.16381

3. [River > Settlement > Country]; [mouthPlace, country] 1.05154

4. [River > Place > Country]; [source, country] 1.01772

5. [River < River > Country]; [sourcePlace, country] 0.97016

6. [River > R̃egion]; [sourcePlace] 0.87977

7. [Stream > Place > Country]; [sourcePlace, country] 0.78617

8. [River > R̃egion]; [sourceRegion] 0.75057

9. [River > S̃tate]; [sourcePlace] 0.74943

(DS 5) Ape/Cave, located in, ?x/Place

0. [Cave > Place]; [location] 7.45401

1. [Cave > L̃ocation]; [location] 4.64778

2. [Cave > PopulatedPlace]; [location] 4.46449

3. [Cave > L̃ocation < PopulatedPlace]; [location, region] 1.45560

4. [Lake > Place]; [location] 1.27032

5. [Cave > L̃ocation < PopulatedPlace]; [location, country] 1.02334

6. [Mountain > Place]; [locatedInArea] 0.92078

7. [Mountain > L̃ocation < Place]; [locatedInArea, location] 0.74736

8. [Mountain > Place]; [mountainRange] 0.69261

9. [Mountain > Place > Place]; [locatedInArea, location] 0.64737

(DS 6) ?x/Protein

0. [Protein] 1.00000

1. [Ĉhromosome] 0.35752

2. [ChemicalCompound] 0.33897

3. [Muscle] 0.14731

4. [Ṽariant] 0.12458

5. [Brain] 0.11722

6. [Drug] 0.11214

7. [Nerve] 0.10568

8. [Fungus] 0.10443

9. [Insect] 0.10352

(DS 7) Claudia Schiffer/Person, height, ?x/Number

0. [Person > N̂umber]; [@height] 12.42687

1. [Person > N̂umber]; [@weight] 3.87524

2. [Person > Person > N̂umber]; [child, @height] 3.37939

3. [Person < Person > N̂umber]; [child, @height] 3.37939

4. [Person > Person > N̂umber]; [spouse, @height] 2.90157

5. [C̃hild > N̂umber]; [@elevation] 2.40223

6. [C̃hild < Person > N̂umber]; [child, @height] 1.99647

7. [S̃pouse > N̂umber]; [@height] 1.93811

8. [C̃hild < Place > N̂umber]; [location, @elevation] 1.76821

9. [C̃hild > N̂umber]; [@point] 1.65839

(DS 8) IBM/Company, revenue, ?x/Number

0. [Company > N̂umber]; [@revenue] 10.96768
1. [Organisation > N̂umber]; [@revenue] 5.27515
2. [Company > N̂umber]; [@netIncome] 4.73824
3. [Company > N̂umber]; [@operatingIncome] 4.07247
4. [Company > Company > N̂umber]; [parentCompany, @revenue] 4.04295
5. [Company < Company > N̂umber]; [parentCompany, @revenue] 4.04295
6. [Õrganization > N̂umber]; [@revenue] 2.91640
7. [D̃istributor > N̂umber]; [@revenue] 2.62752
8. [Non-ProfitOrganisation > N̂umber]; [@revenue] 2.59472
9. [Organisation > N̂umber]; [@netIncome] 2.27331

(DS 9) Limerick Lake/Lake, located in, ?x/Country

0. [Lake > Country]; [country] 10.90295
1. [Lake > Country]; [location] 8.76767
2. [Lake > S̃tate]; [country] 6.02262
3. [Lake > R̃egion]; [country] 5.70758
4. [Lake > S̃tate]; [location] 5.59084
5. [River > Country]; [country] 5.38094
6. [Lake > R̃egion]; [location] 4.89424
7. [Lake > Place > Country]; [country, location] 4.16781
8. [Lake > Ãrea]; [country] 3.84045
9. [Lake < Place > Country]; [location, country] 3.35212

(DS 10) Walt Disney/Company, created, ?x/TV show

0. [Company < TelevisionShow]; [creator] 4.60312
1. [Company < TelevisionShow]; [producer] 4.31560
2. [Company > TelevisionShow]; [product] 3.94430
3. [Company < TelevisionShow]; [developer] 2.98290
4. [Organisation < TelevisionShow]; [creator] 2.30347
5. [Organisation < TelevisionShow]; [producer] 2.11891
6. [Õrganization < TelevisionShow]; [creator] 1.93444
7. [Organisation > TelevisionShow]; [product] 1.89239
8. [D̃istributor < TelevisionShow]; [creator] 1.77068
9. [Company < Broadcast < TelevisionShow]; [owningCompany, creator] 1.75570

(DS 11) Annapurna/Mountain, height, ?x/Number

0. [Mountain > N̂umber]; [@elevation] 7.86171
1. [Mountain > N̂umber]; [@point] 5.72618

2. [Mountain > Place > N̂umber]; [mountainRange, @elevation] 3.91529

3. [Mountain > MountainRange > N̂umber]; [mountainRange, @maximumElevation] 3.63629

4. [Mountain < Mountain > N̂umber]; [parentMountainPeak, @elevation] 3.45214

5. [Island > N̂umber]; [@elevation] 1.81120

6. [Lake > N̂umber]; [@elevation] 1.77965

7. [River > N̂umber]; [@elevation] 1.61796

8. [MountainRange > N̂umber]; [@maximumElevation] 1.60722

9. [Island > N̂umber]; [@point] 1.50265

(DS 12) Jackson/President, involved in, ?x/War || Jackson/President, in, United States/Country

0. [President > MilitaryConflict]; [battle] || [President > Country]; [country] 3.66535

1. [President > MilitaryConflict]; [battle] || [President < Country]; [leaderName] 3.51814

2. [President > B̃attle]; [battle] || [President > Country]; [country] 3.06357

3. [President < B̃attle]; [leaderName] || [President > Country]; [country] 3.01167

4. [President < Person > MilitaryConflict]; [president, battle] || [President > Country]; [country] 2.94648

5. [President > B̃attle]; [battle] || [President < Country]; [leaderName] 2.94053

6. [President < B̃attle]; [leaderName] || [President < Country]; [leaderName] 2.89071

7. [President > MilitaryConflict]; [battle] || [President < Person > Country]; [president, country] 2.84424

8. [President < Person > MilitaryConflict]; [president, battle] || [President < Country]; [leaderName] 2.82814

9. [President > MilitaryConflict]; [battle] || [President > S̃tate]; [country] 2.72651

(DS 13) WikiLeaks/Website, author, ?x/Person

0. [Website > Person]; [author] 7.11753

1. [H̃omepage < Work > Person]; [homepage, writer] 1.51544

2. [H̃omepage < Work > Person]; [homepage, author] 1.24049

3. [H̃omepage < Person]; [homepage] 0.91738

4. [H̃omepage < TelevisionShow > Person]; [homepage, creator] 0.87192

5. [Õrganization < Single > Person]; [writer, writer] 0.80652

6. [Magazine > Person]; [editor] 0.78623

7. [H̃omepage < Magazine > Person]; [homepage, editor] 0.54092

8. [Õrganization < Work > Person]; [writer, author] 0.50999

9. [Magazine > Person]; [publisher] 0.50492

(DS 14) Czech Republic/Country, , ?x/Currency

0. [Country > Currency]; [currency] 8.53040

1. [Country < Currency]; [usingCountry] 5.95008

2. [Country < PopulatedPlace > Currency]; [country, currency] 4.73777

3. [S̃tate > Currency]; [currency] 3.96135

4. [S̃tate < Currency]; [usingCountry] 3.10364

5. [R̃egion > Currency]; [currency] 3.07612

6. [R̃egion < Currency]; [usingCountry] 2.66191

7. [S̃tate < PopulatedPlace > Currency]; [country, currency] 2.62782

8. [R̃egion < PopulatedPlace > Currency]; [country, currency] 2.44207

9. [Ãrea < Currency]; [usingCountry] 1.96875

(DS 15) Berlin/City, area code, ?x/Number

0. [City > N̂umber]; [@areaCode] 11.41993

1. [City < Place > N̂umber]; [location, @areaCode] 6.77350

2. [City > N̂umber]; [@areaTotal] 6.08377

3. [Town > N̂umber]; [@areaCode] 5.77152

4. [City > N̂umber]; [@areaLand] 5.08282

5. [City < Place > N̂umber]; [city, @areaCode] 4.93212

6. [D̃istrict > N̂umber]; [@areaCode] 4.27591

7. [Village > N̂umber]; [@areaCode] 3.55128

8. [Ãrea > N̂umber]; [@areaCode] 3.45415

9. [M̃unicipality > N̂umber]; [@areaCode] 3.40110

(DS 16) ?x/Person, owner, Universal Studios/Organization

0. [Person < Organisation]; [owner] 7.43729

1. [Person < Organisation]; [owningCompany] 5.48728

2. [Person > Organisation]; [team] 4.09509

3. [Person < Organisation]; [foundationPerson] 4.03868

4. [Person < Organisation]; [foundationOrganisation] 4.00728

5. [Person > Õrganization]; [party] 3.75825

6. [Person < Company]; [owner] 3.18776

7. [Person > Õrganization]; [country] 3.04121

8. [Person > Organisation > Õrganization]; [team, location] 2.93385

9. [Person < Company]; [owningCompany] 2.52002

(DS 17) Yenisei/River, flows through, ?x/Country

0. [River > Country]; [country] 6.82908

1. [Stream > Country]; [country] 6.58291

2. [River < BodyOfWater > Country]; [inflow, country] 4.13891

3. [River > S̃tate]; [country] 3.78502

4. [River < BodyOfWater > Country]; [outflow, country] 3.70885

5. [River > BodyOfWater > Country]; [riverMouth, country] 3.69907

6. [River < River > Country]; [riverMouth, country] 3.68730

7. [River > R̃egion]; [country] 3.66067

8. [Stream > S̃tate]; [country] 3.64865

9. [Stream > R̃egion]; [country] 3.52875

(DS 18) Battle of Gettysburg/Event, , ?x/Date

0. [Event > D̂ate]; [@date] 11.88194

1. [Event > D̂ate]; [@startDate] 8.67892

2. [SportsEvent > D̂ate]; [@date] 8.30619

3. [WrestlingEvent > D̂ate]; [@date] 7.36225

4. [Event < MilitaryConflict > D̂ate]; [partOf, @date] 7.22790

5. [Event > MilitaryConflict > D̂ate]; [partOf, @date] 7.22790

6. [Event > D̂ate]; [@landingDate] 6.58207

7. [MixedMartialArtsEvent > D̂ate]; [@date] 6.57140

8. [SportsEvent < MixedMartialArtsEvent > D̂ate]; [previousEvent, @date] 4.41896

9. [SportsEvent > MixedMartialArtsEvent > D̂ate]; [previousEvent, @date] 4.41896

(DS 19) ?x/Mountain, in, Germany/Country

0. [Mountain > Country]; [locatedInArea] 5.95838

1. [Mountain > Country]; [country] 5.83352

2. [Mountain > Place > Country]; [mountainRange, country] 5.57732

3. [Mountain > PopulatedPlace > Country]; [locatedInArea, country] 4.90624

4. [Mountain > Place > Country]; [parentMountainPeak, country] 4.05986

5. [Mountain > S̃tate]; [locatedInArea] 3.51131

6. [Mountain > R̃egion]; [locatedInArea] 3.40528

7. [Mountain > R̃egion]; [mountainRange] 3.30305

8. [Mountain > S̃tate]; [country] 3.23391

9. [Mountain > R̃egion]; [country] 3.13396

(DS 20) ?x/Soccer Club, in, Spain/Country

0. [SoccerClub < Person > Country]; [team, country] 4.15591

1. [SoccerClub < Person > Country]; [managerClub, country] 3.42663

2. [SoccerClub > SoccerLeague > Country]; [league, country] 2.74107

3. [SoccerClub < Person > Country]; [team, stateOfOrigin] 2.71852

4. [SoccerClub < Person > S̃tate]; [team, country] 2.29894

5. [SoccerClub < Person > R̃egion]; [team, country] 2.13964

6. [SoccerClub < Person > Country]; [managerClub, stateOfOrigin] 2.00406

7. [SoccerClub < Person > S̃tate]; [managerClub, country] 1.89300

8. [SoccerClub < Person > R̃egion]; [team, region] 1.84132

9. [SoccerClub < Person > R̃egion]; [managerClub, country] 1.71704

(DS 21) Philippines/Country, , ?x/Official Language

0. [Country > Language]; [officialLanguage] 6.33903

1. [Country < Book > Language]; [country, language] 5.19436

2. [Country > Language]; [language] 5.11652

3. [Country < Language]; [spokenIn] 4.48234

4. [Country < PopulatedPlace > Language]; [country, officialLanguage] 3.41293

5. [S̃tate > Language]; [officialLanguage] 3.12119

6. [S̃tate < Book > Language]; [country, language] 2.93592

7. [R̃egion < Book > Language]; [country, language] 2.82673

8. [S̃tate > Language]; [language] 2.65266

9. [R̃egion > Language]; [officialLanguage] 2.56396

(DS 22) New York/City, mayor, ?x/Person

0. [City > Person]; [leaderName] 4.20158

1. [D̃istrict > Person]; [leaderName] 1.97405

2. [City < Place > Person]; [location, leaderName] 1.96406

3. [City < Company > Person]; [location, keyPerson] 1.94952

4. [City < Person]; [hometown] 1.91772

5. [City < Organisation < Person]; [city, managerClub] 1.85307

6. [Ãrea > Person]; [leaderName] 1.51637

7. [M̃unicipality > Person]; [leaderName] 1.42686

8. [City > P̃arty]; [leaderName] 1.36869

9. [C̃ounty > Person]; [leaderName] 1.26052

(DS 23) ?x/Person, designed, Brooklyn/Bridge

0. [Person < Bridge]; [architect] 2.72538

1. [Person < Building]; [architect] 0.96118

2. [Person > C̃ross < Bridge]; [birthPlace, crosses] 0.93530

3. [Person < HistoricBuilding]; [architect] 0.75010

4. [Person > C̃ross < Bridge]; [deathPlace, crosses] 0.71094

5. [Person > S̃tructure]; [birthPlace] 0.58969

6. [Person > S̃tructure]; [deathPlace] 0.52647

7. [Person > S̃tructure]; [restingPlace] 0.36793

8. [MilitaryPerson > C̃ross < Bridge]; [birthPlace, crosses] 0.34848

9. [MilitaryPerson > C̃ross < Bridge]; [deathPlace, crosses] 0.34442

(DS 24) Karakoram/Mountain Range, , ?x/Highest Place

0. [MountainRange > Place]; [highestPlace] 4.11190

1. [MountainRange < Place]; [mountainRange] 4.00812

2. [MountainRange < Place]; [location] 2.67753
3. [MountainRange < Mountain < Place]; [mountainRange, highestPlace] 2.20607
4. [MountainRange < Mountain > Place]; [mountainRange, mountainRange] 1.72620
5. [MountainRange > PopulatedPlace]; [highestPlace] 1.58063
6. [MountainRange > L̃ocation]; [highestPlace] 1.25366
7. [MountainRange > PopulatedPlace]; [highestMountain] 0.87173
8. [MountainRange < L̃ocation]; [mountainRange] 0.73896
9. [MountainRange > L̃ocation]; [highestMountain] 0.67763

(DS 25) Forbes/Organization, homepage, ?x/Homepage

0. [Organisation > H̃omepage]; [homepage] 13.62638
1. [Õrganization > H̃omepage]; [homepage] 8.75129
2. [Õrganization < Company > H̃omepage]; [location, homepage] 7.20481
3. [EducationalInstitution > H̃omepage]; [homepage] 6.47221
4. [Non-ProfitOrganisation > H̃omepage]; [homepage] 6.42342
5. [Company > H̃omepage]; [homepage] 6.01166
6. [Organisation < Person > H̃omepage]; [team, homepage] 5.19744
7. [Õrganization < Company > H̃omepage]; [foundationOrganisation, homepage] 5.03959
8. [Õrganization < Settlement > H̃omepage]; [country, homepage] 4.54715
9. [Õrganization < Place > H̃omepage]; [owningOrganisation, homepage] 4.38874

(DS 26) ?x/Company, , computer software/Industry

0. [Company > Ĩndustry]; [industry] 12.47260
1. [Company > Ĩndustry]; [location] 9.32496
2. [Company > Ĩndustry]; [service] 6.49195
3. [Company > Company > Ĩndustry]; [parentCompany, industry] 5.24804
4. [Company < Company > Ĩndustry]; [parentCompany, industry] 5.24804
5. [Organisation > Ĩndustry]; [industry] 5.11098
6. [Organisation > Ĩndustry]; [location] 3.88074
7. [Company > Ĩndustry < Company]; [industry, industry] 3.42729
8. [Organisation > Ĩndustry]; [service] 3.11470
9. [Õrganization > Ĩndustry]; [industry] 2.70199

(DS 27) Bruce Carver/Person, died, ?x/Reason

0. [Person > G̃round]; [deathPlace] 4.06231
1. [Person > C̃ause]; [deathPlace] 2.82273
2. [MilitaryPerson > G̃round]; [deathPlace] 2.54760
3. [Person > C̃ause]; [deathCause] 2.40886
4. [Person > Ĩnterest]; [deathPlace] 2.05727

5. [MilitaryPerson > C̃ause]; [deathPlace] 1.77215
6. [Person > Place > G̃round]; [deathPlace, location] 1.57548
7. [Person > T̃eam > G̃round]; [deathPlace, ground] 1.45750
8. [Person < Person > G̃round]; [child, deathPlace] 1.31114
9. [Person > Person > G̃round]; [child, deathPlace] 1.31114

(DM 1) *y/Actor, , Charmed/TV Show || *y/Actor, has, ?x/Official Website

0. [Actor < TelevisionShow]; [starring] || [Actor > H̃omepage]; [homepage] 2.57329
1. [Actor < TelevisionShow]; [showJudge] || [Actor > H̃omepage]; [homepage] 2.12593
2. [Actor < TelevisionShow]; [producer] || [Actor > H̃omepage]; [homepage] 1.87919
3. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [starring, homepage] 1.79692
4. [Actor < TelevisionShow]; [director] || [Actor > H̃omepage]; [homepage] 1.70761
5. [Actor < TelevisionShow]; [composer] || [Actor > H̃omepage]; [homepage] 1.67116
6. [Actor < TelevisionShow]; [starring] || [Actor < Work > P̂age]; [starring, @pages] 1.61913
7. [Actor < TelevisionShow]; [showJudge] || [Actor < Work > H̃omepage]; [starring, homepage] 1.48452
8. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [writer, homepage] 1.42810
9. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [producer, homepage] 1.36509

(DM 2) Richard Nixon/Person, daughter, *y/Person || *y/Person, married to, ?x/Person

0. [Person > Person]; [child] || [Person > Person]; [spouse] 8.12454
1. [Person > Person]; [child] || [Person < Person]; [spouse] 8.12449
2. [Person < Person]; [child] || [Person > Person]; [spouse] 8.12449
3. [Person < Person]; [child] || [Person < Person]; [spouse] 8.12444
4. [Person > Person]; [parent] || [Person > Person]; [spouse] 6.74569
5. [Person > Person]; [parent] || [Person < Person]; [spouse] 6.74565
6. [Person < Person]; [parent] || [Person > Person]; [spouse] 6.74563
7. [Person < Person]; [parent] || [Person < Person]; [spouse] 6.74559
8. [Person > C̃hild < Person]; [child, child] || [Person > Person]; [spouse] 6.38246
9. [Person > C̃hild < Person]; [child, child] || [Person < Person]; [spouse] 6.38242

(DM 3) ?x/City, the largest city in, Egypt/Country || Egypt/Country, capital, ?x/City

0. [City > Country]; [country] || [Country > City]; [capital] 6.90110
1. [City < Country]; [largestCity] || [Country > City]; [capital] 6.71758
2. [City < PopulatedPlace > Country]; [largestCity, country] || [Country > City]; [capital] 6.30455
3. [City < Place > Country]; [location, country] || [Country > City]; [capital] 5.77375
4. [City < EducationalInstitution > Country]; [city, country] || [Country > City]; [capital] 5.58313
5. [City > Country]; [country] || [Country < PopulatedPlace > City]; [country, capital] 5.25397
6. [City < Country]; [largestCity] || [Country < PopulatedPlace > City]; [country, capital] 5.11426
7. [City < PopulatedPlace > Country]; [largestCity, country] || [Country < PopulatedPlace > City]; [country, capital] 4.79980

8. [City > Country]; [country] || [Country < City]; [country] 4.64972
9. [City < Country]; [largestCity] || [Country < City]; [country] 4.52607

(DM 4) Garry Marshall/Person, directed, ?x/Film || ?x/Film, starring, Julia Roberts/Person

0. [Person < Film]; [director] || [Film > Person]; [starring] 10.70593
1. [Person < Film]; [director] || [Film > MilitaryPerson]; [starring] 6.47013
2. [Person < Film]; [director] || [Film > C̃hild < Person]; [starring, child] 6.30669
3. [Person < Film]; [director] || [Film > C̃hild]; [starring] 6.23642
4. [Person < Film]; [director] || [Film > S̃pouse < Person]; [starring, spouse] 6.19369
5. [Person < Film]; [director] || [Film > Person > Person]; [starring, child] 6.18593
6. [Person > S̃pouse < Film]; [spouse, director] || [Film > Person]; [starring] 6.11280
7. [Person > Film]; [notableWork] || [Film > Person]; [starring] 6.10212
8. [Person < Film]; [starring] || [Film > Person]; [starring] 6.05440
9. [Person < Film]; [director] || [Film > Actor > Person]; [starring, spouse] 6.01945

(DM 5) Manhattan/Bridge, has, *y/Type || ?x/Bridge, has, *y/Type

0. [Bridge > T̃ype]; [type] || [Bridge > T̃ype]; [type] 5.38549
1. [Bridge > T̃ype]; [type] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 4.25581
2. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > T̃ype]; [type] 4.25581
3. [Bridge > T̃ype]; [type] || [Bridge > BodyOfWater > T̃ype]; [crosses, location] 3.67317
4. [Bridge > BodyOfWater > T̃ype]; [crosses, location] || [Bridge > T̃ype]; [type] 3.67317
5. [Bridge > T̃ype]; [type] || [S̃tructure > T̃ype]; [type] 3.51981
6. [S̃tructure > T̃ype]; [type] || [Bridge > T̃ype]; [type] 3.51981
7. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 3.36309
8. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > BodyOfWater > T̃ype]; [crosses, location] 2.90267
9. [Bridge > BodyOfWater > T̃ype]; [crosses, location] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 2.90267

(DM 6) ?x/organization, , telecommunication/Industry || ?x/organization, located in, Belgium/Country

0. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [country] 12.57979
1. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [location] 12.11848
2. [Organisation > Ĩndustry]; [location] || [Organisation > Country]; [country] 10.96171
3. [Organisation > Ĩndustry]; [location] || [Organisation > Country]; [location] 10.55974
4. [Organisation > Ĩndustry]; [industry] || [Organisation > PopulatedPlace > Country]; [state, country] 9.81938
5. [Organisation > Ĩndustry]; [industry] || [Organisation > S̃tate]; [country] 9.41061
6. [Organisation > Ĩndustry]; [industry] || [Organisation > R̃egion]; [country] 9.20425
7. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [state] 9.13270
8. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [regionServed] 9.09443
9. [Organisation > Ĩndustry]; [industry] || [Organisation > S̃tate]; [location] 9.00882

(DS 1) Brooklyn/Bridge, crosses, ?x/River

0. [Bridge > River]; [crosses] 9.38807
1. [Bridge > Stream]; [crosses] 6.33271
2. [Bridge > C̃ross < River]; [crosses, riverMouth] 5.10397
3. [Bridge > Stream > River]; [crosses, riverMouth] 4.82787
4. [Bridge > River]; [river] 4.14279
5. [Bridge > C̃ross < Stream]; [crosses, riverMouth] 3.44130
6. [Bridge > Stream > Stream]; [crosses, riverMouth] 3.25500
7. [Bridge > Lake]; [crosses] 3.16446
8. [Bridge > Stream]; [river] 2.79263
9. [Bridge > T̃ributary]; [crosses] 2.51006

(DS 2) Abraham Lincoln/President, died, ?x/Place

0. [President > Place]; [deathPlace] 7.32670
1. [President > PopulatedPlace]; [deathPlace] 5.71046
2. [President > L̃ocation]; [deathPlace] 4.68455
3. [President < Person > Place]; [president, deathPlace] 4.04038
4. [D̃irector > Place]; [deathPlace] 3.53347
5. [Governor > Place]; [deathPlace] 3.51689
6. [President > Place > Place]; [deathPlace, location] 3.49691
7. [P̃resenter > Place]; [deathPlace] 3.48260
8. [President > R̃egion]; [deathPlace] 3.18714
9. [President < Person > PopulatedPlace]; [president, deathPlace] 3.14542

(DS 3) Obama/President, wife, ?x/Person

0. [President > Person]; [spouse] 5.70799
1. [President < Person]; [spouse] 5.63143
2. [President < Person]; [president] 5.05304
3. [President > Person]; [president] 4.11711
4. [P̃resenter < Person]; [spouse] 3.82739
5. [P̃resenter > Person]; [spouse] 3.76419
6. [D̃irector < Person]; [spouse] 3.64943
7. [D̃irector > Person]; [spouse] 3.57920
8. [President > S̃pouse < Person]; [spouse, spouse] 3.14995
9. [Governor > Person]; [spouse] 2.54065

(DS 4) Nile/River, starts, ?x/Country

0. [Stream > Settlement > Country]; [startPoint, country] 1.49303
1. [River > Country]; [sourceCountry] 1.29701

2. [River > Place > Country]; [sourcePlace, country] 1.16381

3. [River > Settlement > Country]; [mouthPlace, country] 1.05154

4. [River > Place > Country]; [source, country] 1.01772

5. [River < River > Country]; [sourcePlace, country] 0.97016

6. [River > R̃egion]; [sourcePlace] 0.87977

7. [Stream > Place > Country]; [sourcePlace, country] 0.78617

8. [River > R̃egion]; [sourceRegion] 0.75057

9. [River > S̃tate]; [sourcePlace] 0.74943

(DS 5) Ape/Cave, , ?x/Place

0. [Cave > Place]; [location] 7.45401

1. [Cave > L̃ocation]; [location] 4.64778

2. [Cave > PopulatedPlace]; [location] 4.46449

3. [Cave > L̃ocation < PopulatedPlace]; [location, region] 1.45560

4. [Lake > Place]; [location] 1.27032

5. [Cave > L̃ocation < PopulatedPlace]; [location, country] 1.02334

6. [Mountain > Place]; [locatedInArea] 0.92078

7. [Mountain > L̃ocation < Place]; [locatedInArea, location] 0.74736

8. [Mountain > Place]; [mountainRange] 0.69261

9. [Mountain > Place > Place]; [locatedInArea, location] 0.64737

(DS 6) ?x/Protein

0. [Protein] 1.00000

1. [Ĉhromosome] 0.35752

2. [ChemicalCompound] 0.33897

3. [Muscle] 0.14731

4. [Ṽariant] 0.12458

5. [Brain] 0.11722

6. [Drug] 0.11214

7. [Nerve] 0.10568

8. [Fungus] 0.10443

9. [Insect] 0.10352

(DS 7) Claudia Schiffer/Person, tall, ?x/Number

0. [Person > N̂umber]; [@height] 9.59783

1. [Person > Person > N̂umber]; [child, @height] 2.74845

2. [Person < Person > N̂umber]; [child, @height] 2.74845

3. [Person > Person > N̂umber]; [spouse, @height] 2.35983

4. [Person < Person > N̂umber]; [spouse, @height] 2.35983

5. [C̃hild > N̂umber]; [@long] 1.84444

6. [C̃hild < HistoricPlace > N̂umber]; [location, @long] 1.72418

7. [C̃hild < Person > N̂umber]; [child, @height] 1.62372

8. [S̃pouse > N̂umber]; [@height] 1.49689

9. [S̃ubject < Place > N̂umber]; [location, @long] 1.45111

(DS 8) IBM/Company, revenue, ?x/Number

0. [Company > N̂umber]; [@revenue] 10.96768

1. [Organisation > N̂umber]; [@revenue] 5.27515

2. [Company > N̂umber]; [@netIncome] 4.73824

3. [Company > N̂umber]; [@operatingIncome] 4.07247

4. [Company > Company > N̂umber]; [parentCompany, @revenue] 4.04295

5. [Company < Company > N̂umber]; [parentCompany, @revenue] 4.04295

6. [Õrganization > N̂umber]; [@revenue] 2.91640

7. [D̃istributor > N̂umber]; [@revenue] 2.62752

8. [Non-ProfitOrganisation > N̂umber]; [@revenue] 2.59472

9. [Organisation > N̂umber]; [@netIncome] 2.27331

(DS 9) Limerick/Lake, country, ?x/Country

0. [Lake > Country]; [country] 10.90295

1. [Lake > Country]; [location] 8.76767

2. [River > Country]; [country] 6.07391

3. [Lake > S̃tate]; [country] 6.02262

4. [Lake > R̃egion]; [country] 5.70758

5. [Lake > S̃tate]; [location] 5.59084

6. [Lake > R̃egion]; [location] 4.89424

7. [Lake > Place > Country]; [country, location] 4.16781

8. [Stream > Country]; [country] 4.05188

9. [Lake > Ãrea]; [country] 3.84045

(DS 10) Walt Disney/Company, created, ?x/Television show

0. [Company < TelevisionShow]; [creator] 4.60312

1. [Company < TelevisionShow]; [producer] 4.31560

2. [Company > TelevisionShow]; [product] 3.94430

3. [Company < TelevisionShow]; [developer] 2.98290

4. [Organisation < TelevisionShow]; [creator] 2.30347

5. [Organisation < TelevisionShow]; [producer] 2.11891

6. [Õrganization < TelevisionShow]; [creator] 1.93444

7. [Organisation > TelevisionShow]; [product] 1.89239

8. [D̃istributor < TelevisionShow]; [creator] 1.77068

9. [Company < Broadcast < TelevisionShow]; [owningCompany, creator] 1.75570

(DS 11) Annapurna/Mountain, height, ?x/Number

0. [Mountain > N̂umber]; [@elevation] 7.86171

1. [Mountain > N̂umber]; [@point] 5.72618

2. [Mountain > Place > N̂umber]; [mountainRange, @elevation] 3.91529

3. [Mountain > MountainRange > N̂umber]; [mountainRange, @maximumElevation] 3.63629

4. [Mountain < Mountain > N̂umber]; [parentMountainPeak, @elevation] 3.45214

5. [Island > N̂umber]; [@elevation] 1.81120

6. [Lake > N̂umber]; [@elevation] 1.77965

7. [River > N̂umber]; [@elevation] 1.61796

8. [MountainRange > N̂umber]; [@maximumElevation] 1.60722

9. [Island > N̂umber]; [@point] 1.50265

(DS 12) Jackson/U.S. President, involved in, ?x/War

0. [President > MilitaryConflict]; [battle] 1.51647

1. [President > B̃attle]; [battle] 1.05940

2. [President < B̃attle]; [leaderName] 0.83415

3. [President < Person > MilitaryConflict]; [president, battle] 0.82293

4. [Governor > MilitaryConflict]; [battle] 0.61919

5. [President < Person > B̃attle]; [president, battle] 0.59298

6. [President < MilitaryConflict]; [commander] 0.58291

7. [Governor > B̃attle]; [battle] 0.53490

8. [PrimeMinister < B̃attle]; [leaderName] 0.47020

9. [President > Person > MilitaryConflict]; [president, battle] 0.44900

(DS 13) ?x/Author, , WikiLeaks/Website

0. [Ãuthor < Website]; [author] 8.65513

1. [Person < Website]; [author] 1.88654

2. [Artist < Website]; [author] 1.54471

3. [Writer > Ãuthor < Website]; [influencedBy, author] 0.25260

4. [Writer > Ãuthor < Website]; [influenced, author] 0.18328

5. [Writer > Ãuthor < Website]; [birthPlace, author] 0.14291

6. [Writer > G̃enre < Website]; [genre, language] 0.09146

7. [Scientist > Ãuthor < Website]; [birthPlace, author] 0.05618

8. [Scientist > Ãuthor < Website]; [nationality, author] 0.04374

9. [Scientist > Ãuthor < Website]; [deathPlace, author] 0.03309

(DS 14) Czech Republic/Country, , ?x/Currency

0. [Country > Currency]; [currency] 8.53040
1. [Country < Currency]; [usingCountry] 5.95008
2. [Country < PopulatedPlace > Currency]; [country, currency] 4.73777
3. [S̃tate > Currency]; [currency] 3.96135
4. [S̃tate < Currency]; [usingCountry] 3.10364
5. [R̃egion > Currency]; [currency] 3.07612
6. [R̃egion < Currency]; [usingCountry] 2.66191
7. [S̃tate < PopulatedPlace > Currency]; [country, currency] 2.62782
8. [R̃egion < PopulatedPlace > Currency]; [country, currency] 2.44207
9. [Ãrea < Currency]; [usingCountry] 1.96875

(DS 15) Berlin/City, , ?x/Area Code

0. [City > Ârea code]; [@areaCode] 11.47158
1. [City < Place > Ârea code]; [location, @areaCode] 6.85407
2. [Town > Ârea code]; [@areaCode] 5.79764
3. [City < Place > Ârea code]; [city, @areaCode] 5.01270
4. [D̃istrict > Ârea code]; [@areaCode] 4.36728
5. [Village > Ârea code]; [@areaCode] 3.58156
6. [Ãrea > Ârea code]; [@areaCode] 3.47083
7. [M̃unicipality > Ârea code]; [@areaCode] 3.44617
8. [City > PopulatedPlace > Ârea code]; [country, @areaCode] 3.40843
9. [D̃istrict < Place > Ârea code]; [district, @areaCode] 3.20034

(DS 16) ?x/Person, owner, Universal Studios/Company

0. [Person < Company]; [owner] 6.64435
1. [Person < Company]; [owningCompany] 5.25257
2. [Person < Company]; [foundationPerson] 3.95117
3. [Person < Company]; [keyPerson] 3.85356
4. [Person < Organisation]; [owner] 3.56820
5. [Person < Organisation]; [owningCompany] 2.63263
6. [Person < Company]; [foundationOrganisation] 2.52712
7. [Person < Organisation]; [foundationPerson] 1.93762
8. [Person < Organisation]; [keyPerson] 1.86701
9. [Person > Organisation]; [managerClub] 1.86169

(DS 17) Yenisei/River, flows, ?x/Country

0. [River > Country]; [country] 6.82908
1. [Stream > Country]; [country] 6.58291

2. [River < BodyOfWater > Country]; [inflow, country] 4.13891

3. [River > S̃tate]; [country] 3.78502

4. [River < BodyOfWater > Country]; [outflow, country] 3.70885

5. [River > BodyOfWater > Country]; [riverMouth, country] 3.69907

6. [River < River > Country]; [riverMouth, country] 3.68730

7. [River > R̃egion]; [country] 3.66067

8. [Stream > S̃tate]; [country] 3.64865

9. [Stream > R̃egion]; [country] 3.52875

(DS 18) Gettysburg/Battle, , ?x/Date

0. [B̃attle > D̂ate]; [@date] 10.68902

1. [B̃attle < Person > D̂ate]; [battle, @birthDate] 7.79949

2. [B̃attle < MilitaryConflict > D̂ate]; [partOf, @date] 7.34063

3. [B̃attle < MilitaryPerson > D̂ate]; [battle, @deathDate] 7.25877

4. [B̃attle > MilitaryConflict > D̂ate]; [partOf, @date] 6.29978

5. [MilitaryConflict > D̂ate]; [@date] 2.95422

6. [MilitaryConflict < Person > D̂ate]; [battle, @birthDate] 1.89645

7. [MilitaryConflict < MilitaryPerson > D̂ate]; [battle, @deathDate] 1.76131

8. [B̃attle < MilitaryPerson > Ŷear]; [battle, @serviceStartYear] 1.35416

9. [B̃attle < MilitaryUnit > Ŷear]; [battle, @activeYearsEndYear] 1.32953

(DS 19) ?x/Mountain, in, Germany/Country

0. [Mountain > Country]; [locatedInArea] 5.95838

1. [Mountain > Country]; [country] 5.83352

2. [Mountain > Place > Country]; [mountainRange, country] 5.57732

3. [Mountain > PopulatedPlace > Country]; [locatedInArea, country] 4.90624

4. [Mountain > Place > Country]; [parentMountainPeak, country] 4.05986

5. [Mountain > S̃tate]; [locatedInArea] 3.51131

6. [Mountain > R̃egion]; [locatedInArea] 3.40528

7. [Mountain > R̃egion]; [mountainRange] 3.30305

8. [Mountain > S̃tate]; [country] 3.23391

9. [Mountain > R̃egion]; [country] 3.13396

(DS 20) ?x/Soccer Club, in, Spain/Country

0. [SoccerClub < Person > Country]; [team, country] 4.15591

1. [SoccerClub < Person > Country]; [managerClub, country] 3.42663

2. [SoccerClub > SoccerLeague > Country]; [league, country] 2.74107

3. [SoccerClub < Person > Country]; [team, stateOfOrigin] 2.71852

4. [SoccerClub < Person > S̃tate]; [team, country] 2.29894

5. [SoccerClub < Person > R̃egion]; [team, country] 2.13964

6. [SoccerClub < Person > Country]; [managerClub, stateOfOrigin] 2.00406

7. [SoccerClub < Person > S̃tate]; [managerClub, country] 1.89300

8. [SoccerClub < Person > R̃egion]; [team, region] 1.84132

9. [SoccerClub < Person > R̃egion]; [managerClub, country] 1.71704

(DS 21) Philippines/Country, , ?x/Official Language

0. [Country > Language]; [officialLanguage] 6.33903

1. [Country < Book > Language]; [country, language] 5.19436

2. [Country > Language]; [language] 5.11652

3. [Country < Language]; [spokenIn] 4.48234

4. [Country < PopulatedPlace > Language]; [country, officialLanguage] 3.41293

5. [S̃tate > Language]; [officialLanguage] 3.12119

6. [S̃tate < Book > Language]; [country, language] 2.93592

7. [R̃egion < Book > Language]; [country, language] 2.82673

8. [S̃tate > Language]; [language] 2.65266

9. [R̃egion > Language]; [officialLanguage] 2.56396

(DS 22) New York/City, mayor, ?x/Person

0. [City > Person]; [leaderName] 4.20158

1. [D̃istrict > Person]; [leaderName] 1.97405

2. [City < Place > Person]; [location, leaderName] 1.96406

3. [City < Company > Person]; [location, keyPerson] 1.94952

4. [City < Person]; [hometown] 1.91772

5. [City < Organisation < Person]; [city, managerClub] 1.85307

6. [Ãrea > Person]; [leaderName] 1.51637

7. [M̃unicipality > Person]; [leaderName] 1.42686

8. [City > P̃arty]; [leaderName] 1.36869

9. [C̃ounty > Person]; [leaderName] 1.26052

(DS 23) ?x/Person, designed, Brooklyn/Bridge

0. [Person < Bridge]; [architect] 2.72538

1. [Person < Building]; [architect] 0.96118

2. [Person > C̃ross < Bridge]; [birthPlace, crosses] 0.93530

3. [Person < HistoricBuilding]; [architect] 0.75010

4. [Person > C̃ross < Bridge]; [deathPlace, crosses] 0.71094

5. [Person > S̃tructure]; [birthPlace] 0.58969

6. [Person > S̃tructure]; [deathPlace] 0.52647

7. [Person > S̃tructure]; [restingPlace] 0.36793

8. [MilitaryPerson > C̃ross < Bridge]; [birthPlace, crosses] 0.34848

9. [MilitaryPerson > C̃ross < Bridge]; [deathPlace, crosses] 0.34442

(DS 24) Karakoram/Place, highest, ?x/Place

0. [Place > Place]; [highestPlace] 6.35592

1. [Place < Place]; [highestPlace] 6.35582

2. [Place > Place]; [highestMountain] 5.11853

3. [Place < Place]; [highestMountain] 5.11843

4. [Place > Place]; [highestRegion] 4.47268

5. [Place > PopulatedPlace]; [highestPlace] 3.89268

6. [PopulatedPlace < Place]; [highestPlace] 3.89258

7. [Place > PopulatedPlace]; [highestMountain] 3.86343

8. [PopulatedPlace < Place]; [highestMountain] 3.86333

9. [Place > PopulatedPlace]; [highestRegion] 3.47290

(DS 25) Forbes/Magazine, , ?x/Homepage

0. [Magazine > H̃omepage]; [homepage] 9.84085

1. [Newspaper > H̃omepage]; [homepage] 3.77063

2. [P̃ublisher > H̃omepage]; [homepage] 3.59728

3. [Magazine > Company > H̃omepage]; [publisher, homepage] 3.58770

4. [P̃ublisher < Company > H̃omepage]; [location, homepage] 2.81021

5. [P̃ublisher < Software > H̃omepage]; [publisher, homepage] 2.76114

6. [Company > H̃omepage]; [homepage] 1.68699

7. [Company < Software > H̃omepage]; [publisher, homepage] 1.35058

8. [Journalist > H̃omepage]; [homepage] 1.10043

9. [Writer > H̃omepage]; [homepage] 1.05295

(DS 26) ?x/Company, in, computer software/Industry

0. [Company > Ĩndustry]; [industry] 12.47260

1. [Company > Ĩndustry]; [location] 9.32496

2. [Company > Ĩndustry]; [service] 6.49195

3. [Company > Company > Ĩndustry]; [parentCompany, industry] 5.24804

4. [Company < Company > Ĩndustry]; [parentCompany, industry] 5.24804

5. [Organisation > Ĩndustry]; [industry] 5.11098

6. [Organisation > Ĩndustry]; [location] 3.88074

7. [Company > Ĩndustry < Company]; [industry, industry] 3.42729

8. [Organisation > Ĩndustry]; [service] 3.11470

9. [Õrganization > Ĩndustry]; [industry] 2.70199

(DS 27) Bruce Carver/Person, died from, ?x/Reason

0. [Person > G̃round]; [deathPlace] 4.06231
1. [Person > C̃ause]; [deathPlace] 2.82273
2. [MilitaryPerson > G̃round]; [deathPlace] 2.54760
3. [Person > C̃ause]; [deathCause] 2.40886
4. [Person > Ĩnterest]; [deathPlace] 2.05727
5. [MilitaryPerson > C̃ause]; [deathPlace] 1.77215
6. [Person > Place > G̃round]; [deathPlace, location] 1.57548
7. [Person > T̃eam > G̃round]; [deathPlace, ground] 1.45750
8. [Person < Person > G̃round]; [child, deathPlace] 1.31114
9. [Person > Person > G̃round]; [child, deathPlace] 1.31114

(DM 1) *y/Actors, , Charmed/Television Show || *y/Actors, , ?x/Official Website

0. [Actor < TelevisionShow]; [starring] || [Actor > H̃omepage]; [homepage] 2.57329
1. [Actor < TelevisionShow]; [showJudge] || [Actor > H̃omepage]; [homepage] 2.20386
2. [Actor < TelevisionShow]; [producer] || [Actor > H̃omepage]; [homepage] 1.87919
3. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [starring, homepage] 1.79692
4. [Actor < TelevisionShow]; [director] || [Actor > H̃omepage]; [homepage] 1.70761
5. [Actor < TelevisionShow]; [composer] || [Actor > H̃omepage]; [homepage] 1.67116
6. [Actor < TelevisionShow]; [starring] || [Actor < Work > P̂age]; [starring, @pages] 1.61913
7. [Actor < TelevisionShow]; [showJudge] || [Actor < Work > H̃omepage]; [starring, homepage] 1.53894
8. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [writer, homepage] 1.42810
9. [Actor < TelevisionShow]; [showJudge] || [Actor < Work > P̂age]; [starring, @pages] 1.38668

(DM 2) Richard Nixon/President, daughter, *y/Person || *y/Person, married to, ?x/Person

0. [President > Person]; [child] || [Person > Person]; [spouse] 6.53598
1. [President > Person]; [child] || [Person < Person]; [spouse] 6.53594
2. [President < Person]; [child] || [Person > Person]; [spouse] 6.26694
3. [President < Person]; [child] || [Person < Person]; [spouse] 6.26690
4. [President < Person]; [president] || [Person > Person]; [spouse] 6.18157
5. [President < Person]; [president] || [Person < Person]; [spouse] 6.18154
6. [President > Person]; [president] || [Person > Person]; [spouse] 5.57981
7. [President > Person]; [president] || [Person < Person]; [spouse] 5.57978
8. [President > Person]; [child] || [Person > S̃pouse < Person]; [spouse, spouse] 5.05292
9. [P̃resenter < Person]; [child] || [Person > Person]; [spouse] 4.94444

(DM 3) Egypt/Country, largest, ?x/City || Egypt/Country, capital, ?x/City

0. [Country > City]; [largestCity] || [Country > City]; [capital] 5.81765
1. [Country < PopulatedPlace > City]; [country, largestCity] || [Country > City]; [capital] 5.45990

2. [Country > City]; [largestCity] || [Country < PopulatedPlace > City]; [country, capital] 4.42912

3. [Country < PopulatedPlace > City]; [country, largestCity] || [Country < PopulatedPlace > City]; [country, capital] 4.15675

4. [Country > City]; [largestCity] || [Country < City]; [country] 3.91973

5. [Country < PopulatedPlace > City]; [country, largestCity] || [Country < City]; [country] 3.67868

6. [Country < Place > City]; [location, largestCity] || [Country > City]; [capital] 3.44586

7. [R̃egion > City]; [largestCity] || [R̃egion > City]; [capital] 3.32478

8. [Country > City]; [largestCity] || [Country < Place > City]; [country, location] 3.27945

9. [S̃tate > City]; [largestCity] || [S̃tate > City]; [capital] 3.27126

(DM 4) ?x/Film, directed, Garry Marshall/Person || ?x/Film, starring, Julia Roberts/Person

0. [Film > Person]; [director] || [Film > Person]; [starring] 10.70599

1. [Film > Person]; [director] || [Film > MilitaryPerson]; [starring] 6.47017

2. [Film > Person]; [director] || [Film > C̃hild < Person]; [starring, child] 6.30672

3. [Film > Person]; [director] || [Film > C̃hild]; [starring] 6.23646

4. [Film > Person]; [director] || [Film > S̃pouse < Person]; [starring, spouse] 6.19373

5. [Film > Person]; [director] || [Film > Person > Person]; [starring, child] 6.18597

6. [Film > S̃pouse < Person]; [director, spouse] || [Film > Person]; [starring] 6.11280

7. [Film < Person]; [notableWork] || [Film > Person]; [starring] 6.10200

8. [Film > Person]; [starring] || [Film > Person]; [starring] 6.05452

9. [Film > Person]; [director] || [Film > Actor > Person]; [starring, spouse] 6.01949

(DM 5) Manhattan/Bridge, has, *y/Type || ?x/Bridge, has, *y/Type

0. [Bridge > T̃ype]; [type] || [Bridge > T̃ype]; [type] 5.38549

1. [Bridge > T̃ype]; [type] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 4.25581

2. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > T̃ype]; [type] 4.25581

3. [Bridge > T̃ype]; [type] || [Bridge > BodyOfWater > T̃ype]; [crosses, location] 3.67317

4. [Bridge > BodyOfWater > T̃ype]; [crosses, location] || [Bridge > T̃ype]; [type] 3.67317

5. [Bridge > T̃ype]; [type] || [S̃tructure > T̃ype]; [type] 3.51981

6. [S̃tructure > T̃ype]; [type] || [Bridge > T̃ype]; [type] 3.51981

7. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 3.36309

8. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > BodyOfWater > T̃ype]; [crosses, location] 2.90267

9. [Bridge > BodyOfWater > T̃ype]; [crosses, location] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 2.90267

(DM 6) ?x/organization, , telecommunication/Industry || ?x/organization, located, Belgium/Country

0. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [country] 12.57979

1. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [location] 12.11848

2. [Organisation > Ĩndustry]; [location] || [Organisation > Country]; [country] 10.96171

3. [Organisation > Ĩndustry]; [location] || [Organisation > Country]; [location] 10.55974

4. [Organisation > Ĩndustry]; [industry] || [Organisation > PopulatedPlace > Country]; [state, country] 9.81938

5. [Organisation > Ĩndustry]; [industry] || [Organisation > S̃tate]; [country] 9.41061

6. [Organisation > Ĩndustry]; [industry] || [Organisation > R̃egion]; [country] 9.20425

7. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [state] 9.13270

8. [Organisation > Ĩndustry]; [industry] || [Organisation > Country]; [regionServed] 9.09443

9. [Organisation > Ĩndustry]; [industry] || [Organisation > S̃tate]; [location] 9.00882

(DS 1) Brooklyn/Bridge, cross, ?x/River

0. [Bridge > River]; [crosses] 9.38807

1. [Bridge > Stream]; [crosses] 6.33271

2. [Bridge > C̃ross < River]; [crosses, riverMouth] 5.10397

3. [Bridge > Stream > River]; [crosses, riverMouth] 4.82787

4. [Bridge > River]; [river] 4.14279

5. [Bridge > C̃ross < Stream]; [crosses, riverMouth] 3.44130

6. [Bridge > Stream > Stream]; [crosses, riverMouth] 3.25500

7. [Bridge > Lake]; [crosses] 3.16446

8. [Bridge > Stream]; [river] 2.79263

9. [Bridge > T̃ributary]; [crosses] 2.51006

(DS 2) Abraham Lincoln/Person, died, ?x/Place

0. [Person > Place]; [deathPlace] 10.29601

1. [Person > PopulatedPlace]; [deathPlace] 8.02293

2. [MilitaryPerson > Place]; [deathPlace] 6.59922

3. [Person > L̃ocation]; [deathPlace] 6.59287

4. [Person > Place > Place]; [deathPlace, location] 6.46622

5. [MilitaryPerson > PopulatedPlace]; [deathPlace] 5.14085

6. [Person > R̃egion < Place]; [deathPlace, region] 5.07034

7. [Person > B̃attle > Place]; [deathPlace, place] 4.74140

8. [Person > R̃egion]; [deathPlace] 4.68545

9. [Person > Place > PopulatedPlace]; [deathPlace, location] 4.35680

(DS 3) Obama/President, , *y/Wife || *y/Wife, called, ?x/name

0. [President < S̃pouse]; [president] || [S̃pouse > N̂ame]; [@name] 5.35065

1. [President > S̃pouse]; [spouse] || [S̃pouse > N̂ame]; [@name] 5.25250

2. [President > S̃pouse]; [president] || [S̃pouse > N̂ame]; [@name] 5.17682

3. [President < S̃pouse]; [spouse] || [S̃pouse > N̂ame]; [@name] 4.75462

4. [President < S̃pouse]; [president] || [S̃pouse < Person > N̂ame]; [spouse, @name] 4.38813

5. [President > S̃pouse]; [spouse] || [S̃pouse < Person > N̂ame]; [spouse, @name] 4.30764

6. [President > S̃pouse]; [president] || [S̃pouse < Person > N̂ame]; [spouse, @name] 4.24557

7. [President < S̃pouse]; [president] || [S̃pouse > Person > N̂ame]; [spouse, @name] 4.14954

8. [President > S̃pouse]; [spouse] || [S̃pouse > Person > N̂ame]; [spouse, @name] 4.07343
9. [President < S̃pouse]; [president] || [S̃pouse > N̂ame]; [@birthName] 4.05785

(DS 4) Nile/River, starts, ?x/Country

0. [Stream > Settlement > Country]; [startPoint, country] 1.49303
1. [River > Country]; [sourceCountry] 1.29701
2. [River > Place > Country]; [sourcePlace, country] 1.16381
3. [River > Settlement > Country]; [mouthPlace, country] 1.05154
4. [River > Place > Country]; [source, country] 1.01772
5. [River < River > Country]; [sourcePlace, country] 0.97016
6. [River > R̃egion]; [sourcePlace] 0.87977
7. [Stream > Place > Country]; [sourcePlace, country] 0.78617
8. [River > R̃egion]; [sourceRegion] 0.75057
9. [River > S̃tate]; [sourcePlace] 0.74943

(DS 5) Ape Cave/Cave, location, ?x/Place

0. [Cave > Place]; [location] 7.45401
1. [Cave > L̃ocation]; [location] 4.64778
2. [Cave > PopulatedPlace]; [location] 4.46449
3. [Cave > L̃ocation < PopulatedPlace]; [location, region] 1.45560
4. [Lake > Place]; [location] 1.27032
5. [Cave > L̃ocation < PopulatedPlace]; [location, country] 1.02334
6. [Mountain > Place]; [locatedInArea] 0.92078
7. [Mountain > L̃ocation < Place]; [locatedInArea, location] 0.74736
8. [Mountain > Place]; [mountainRange] 0.69261
9. [Mountain > Place > Place]; [locatedInArea, location] 0.64737

(DS 6) ?x/Protein

0. [Protein] 1.00000
1. [Ĉhromosome] 0.35752
2. [ChemicalCompound] 0.33897
3. [Muscle] 0.14731
4. [Ṽariant] 0.12458
5. [Brain] 0.11722
6. [Drug] 0.11214
7. [Nerve] 0.10568
8. [Fungus] 0.10443
9. [Insect] 0.10352

(DS 7) Claudia Schiffer/Person, height, ?x/Number

0. [Person > N̂umber]; [@height] 12.42687

1. [Person > N̂umber]; [@weight] 3.87524

2. [Person > Person > N̂umber]; [child, @height] 3.37939

3. [Person < Person > N̂umber]; [child, @height] 3.37939

4. [Person > Person > N̂umber]; [spouse, @height] 2.90157

5. [C̃hild > N̂umber]; [@elevation] 2.40223

6. [C̃hild < Person > N̂umber]; [child, @height] 1.99647

7. [S̃pouse > N̂umber]; [@height] 1.93811

8. [C̃hild < Place > N̂umber]; [location, @elevation] 1.76821

9. [C̃hild > N̂umber]; [@point] 1.65839

(DS 8) IBM/Company, revenue, ?x/Number

0. [Company > N̂umber]; [@revenue] 10.96768

1. [Organisation > N̂umber]; [@revenue] 5.27515

2. [Company > N̂umber]; [@netIncome] 4.73824

3. [Company > N̂umber]; [@operatingIncome] 4.07247

4. [Company > Company > N̂umber]; [parentCompany, @revenue] 4.04295

5. [Company < Company > N̂umber]; [parentCompany, @revenue] 4.04295

6. [Õrganization > N̂umber]; [@revenue] 2.91640

7. [D̃istributor > N̂umber]; [@revenue] 2.62752

8. [Non-ProfitOrganisation > N̂umber]; [@revenue] 2.59472

9. [Organisation > N̂umber]; [@netIncome] 2.27331

(DS 9) Limerick Lake/Lake, location, ?x/Country

0. [Lake > Country]; [country] 10.90295

1. [Lake > Country]; [location] 8.76767

2. [Lake > S̃tate]; [country] 6.02262

3. [Lake > R̃egion]; [country] 5.70758

4. [Lake > S̃tate]; [location] 5.59084

5. [River > Country]; [country] 5.38094

6. [Lake > R̃egion]; [location] 4.89424

7. [Lake > Place > Country]; [country, location] 4.16781

8. [Lake > Ãrea]; [country] 3.84045

9. [Lake < Place > Country]; [location, country] 3.35212

(DS 10) Walt Disney/Company, created, ?x/TV show

0. [Company < TelevisionShow]; [creator] 4.60312

1. [Company < TelevisionShow]; [producer] 4.31560

2. [Company > TelevisionShow]; [product] 3.94430

3. [Company < TelevisionShow]; [developer] 2.98290

4. [Organisation < TelevisionShow]; [creator] 2.30347

5. [Organisation < TelevisionShow]; [producer] 2.11891

6. [Õrganization < TelevisionShow]; [creator] 1.93444

7. [Organisation > TelevisionShow]; [product] 1.89239

8. [D̃istributor < TelevisionShow]; [creator] 1.77068

9. [Company < Broadcast < TelevisionShow]; [owningCompany, creator] 1.75570

(DS 11) Annapurna/Mountain, height, ?x/Number

0. [Mountain > N̂umber]; [@elevation] 7.86171

1. [Mountain > N̂umber]; [@point] 5.72618

2. [Mountain > Place > N̂umber]; [mountainRange, @elevation] 3.91529

3. [Mountain > MountainRange > N̂umber]; [mountainRange, @maximumElevation] 3.63629

4. [Mountain < Mountain > N̂umber]; [parentMountainPeak, @elevation] 3.45214

5. [Island > N̂umber]; [@elevation] 1.81120

6. [Lake > N̂umber]; [@elevation] 1.77965

7. [River > N̂umber]; [@elevation] 1.61796

8. [MountainRange > N̂umber]; [@maximumElevation] 1.60722

9. [Island > N̂umber]; [@point] 1.50265

(DS 12) Jackson/U.S. President, involved, ?x/Wars

0. [President > MilitaryConflict]; [battle] 1.51647

1. [President > B̃attle]; [battle] 1.05940

2. [President < B̃attle]; [leaderName] 0.83415

3. [President < Person > MilitaryConflict]; [president, battle] 0.82293

4. [Governor > MilitaryConflict]; [battle] 0.61919

5. [President < Person > B̃attle]; [president, battle] 0.59298

6. [President < MilitaryConflict]; [commander] 0.58291

7. [Governor > B̃attle]; [battle] 0.53490

8. [PrimeMinister < B̃attle]; [leaderName] 0.47020

9. [President > Person > MilitaryConflict]; [president, battle] 0.44900

(DS 13) ?x/Author, , WikiLeaks/Website

0. [Ãuthor < Website]; [author] 8.65513

1. [Person < Website]; [author] 1.88654

2. [Artist < Website]; [author] 1.54471

3. [Writer > Ãuthor < Website]; [influencedBy, author] 0.25260

4. [Writer > Ãuthor < Website]; [influenced, author] 0.18328

5. [Writer > Ãuthor < Website]; [birthPlace, author] 0.14291

6. [Writer > G̃enre < Website]; [genre, language] 0.09146

7. [Scientist > Ãuthor < Website]; [birthPlace, author] 0.05618

8. [Scientist > Ãuthor < Website]; [nationality, author] 0.04374

9. [Scientist > Ãuthor < Website]; [deathPlace, author] 0.03309

(DS 14) Czech Republic/Country, , ?x/Currency

0. [Country > Currency]; [currency] 8.53040

1. [Country < Currency]; [usingCountry] 5.95008

2. [Country < PopulatedPlace > Currency]; [country, currency] 4.73777

3. [S̃tate > Currency]; [currency] 3.96135

4. [S̃tate < Currency]; [usingCountry] 3.10364

5. [R̃egion > Currency]; [currency] 3.07612

6. [R̃egion < Currency]; [usingCountry] 2.66191

7. [S̃tate < PopulatedPlace > Currency]; [country, currency] 2.62782

8. [R̃egion < PopulatedPlace > Currency]; [country, currency] 2.44207

9. [Ãrea < Currency]; [usingCountry] 1.96875

(DS 15) Berlin/City, area code, ?x/Number

0. [City > N̂umber]; [@areaCode] 11.41993

1. [City < Place > N̂umber]; [location, @areaCode] 6.77350

2. [City > N̂umber]; [@areaTotal] 6.08377

3. [Town > N̂umber]; [@areaCode] 5.77152

4. [City > N̂umber]; [@areaLand] 5.08282

5. [City < Place > N̂umber]; [city, @areaCode] 4.93212

6. [D̃istrict > N̂umber]; [@areaCode] 4.27591

7. [Village > N̂umber]; [@areaCode] 3.55128

8. [Ãrea > N̂umber]; [@areaCode] 3.45415

9. [M̃unicipality > N̂umber]; [@areaCode] 3.40110

(DS 16) ?x/Person, owns, Universal Studios/Organization

0. [Person < Organisation]; [owner] 7.13920

1. [Person < Organisation]; [owningCompany] 5.59395

2. [Person > S̃pouse < Organisation]; [spouse, location] 3.39649

3. [Person < Company]; [owner] 3.06000

4. [Person < Company]; [owningCompany] 2.56901

5. [Person < T̃eam]; [owner] 2.52109

6. [Person < Company < Organisation]; [foundationPerson, owner] 2.05418

7. [Person > Organisation]; [managerClub] 2.02226

8. [Person < Building > Õrganization]; [owner, location] 1.82612
9. [S̃ubject < Company < Organisation]; [location, owner] 1.53229

(DS 17) Yenisei/River, flows, ?x/Country

0. [River > Country]; [country] 6.82908
1. [Stream > Country]; [country] 6.58291
2. [River < BodyOfWater > Country]; [inflow, country] 4.13891
3. [River > S̃tate]; [country] 3.78502
4. [River < BodyOfWater > Country]; [outflow, country] 3.70885
5. [River > BodyOfWater > Country]; [riverMouth, country] 3.69907
6. [River < River > Country]; [riverMouth, country] 3.68730
7. [River > R̃egion]; [country] 3.66067
8. [Stream > S̃tate]; [country] 3.64865
9. [Stream > R̃egion]; [country] 3.52875

(DS 18) Battle of Gettysburg/Battle, took place, ?x/Year

0. [B̃attle < SoccerPlayer > Ŷear]; [birthPlace, @activeYearsEndYear] 4.72769
1. [B̃attle < Person > Ŷear]; [birthPlace, @activeYearsStartYear] 4.52519
2. [B̃attle < Person > Ŷear]; [birthPlace, @birthYear] 4.11618
3. [B̃attle < Person > Ŷear]; [deathPlace, @activeYearsEndYear] 3.91427
4. [B̃attle < Person > Ŷear]; [deathPlace, @activeYearsStartYear] 3.81221
5. [B̃attle < Person > D̂ate]; [birthPlace, @activeYearsStartYear] 1.49867
6. [B̃attle < SoccerPlayer > D̂ate]; [birthPlace, @activeYearsEndYear] 1.44701
7. [B̃attle > T̃ime zone]; [place] 1.34959
8. [B̃attle < Person > D̂ate]; [deathPlace, @activeYearsStartYear] 1.26254
9. [B̃attle < Person > D̂ate]; [birthPlace, @birthYear] 1.24291

(DS 19) ?x/Mountain, ,Germany/Country

0. [Mountain > Country]; [locatedInArea] 5.95838
1. [Mountain > Country]; [country] 5.83352
2. [Mountain > Place > Country]; [mountainRange, country] 5.57732
3. [Mountain > PopulatedPlace > Country]; [locatedInArea, country] 4.90624
4. [Mountain > Place > Country]; [parentMountainPeak, country] 4.05986
5. [Mountain > S̃tate]; [locatedInArea] 3.51131
6. [Mountain > R̃egion]; [locatedInArea] 3.40528
7. [Mountain > R̃egion]; [mountainRange] 3.30305
8. [Mountain > S̃tate]; [country] 3.23391
9. [Mountain > R̃egion]; [country] 3.13396

(DS 20) ?x/Soccer Club, in, Spain/Country

0. [SoccerClub < Person > Country]; [team, country] 4.15591

1. [SoccerClub < Person > Country]; [managerClub, country] 3.42663

2. [SoccerClub > SoccerLeague > Country]; [league, country] 2.74107

3. [SoccerClub < Person > Country]; [team, stateOfOrigin] 2.71852

4. [SoccerClub < Person > S̃tate]; [team, country] 2.29894

5. [SoccerClub < Person > R̃egion]; [team, country] 2.13964

6. [SoccerClub < Person > Country]; [managerClub, stateOfOrigin] 2.00406

7. [SoccerClub < Person > S̃tate]; [managerClub, country] 1.89300

8. [SoccerClub < Person > R̃egion]; [team, region] 1.84132

9. [SoccerClub < Person > R̃egion]; [managerClub, country] 1.71704

(DS 21) Philippines/Country, official language, ?x/Languages

0. [Country > Language]; [officialLanguage] 8.25635

1. [Country > Language]; [language] 6.66408

2. [Country < Book > Language]; [country, language] 6.49483

3. [Country < Language]; [spokenIn] 5.83811

4. [Country < PopulatedPlace > Language]; [country, officialLanguage] 4.44521

5. [S̃tate > Language]; [officialLanguage] 4.12613

6. [S̃tate < Book > Language]; [country, language] 3.62707

7. [S̃tate < Book > Language]; [language, language] 3.60078

8. [R̃egion > Language]; [officialLanguage] 3.52106

9. [S̃tate > Language]; [language] 3.50675

(DS 22) New York/City, mayor, ?x/Person

0. [City > Person]; [leaderName] 4.20158

1. [D̃istrict > Person]; [leaderName] 1.97405

2. [City < Place > Person]; [location, leaderName] 1.96406

3. [City < Company > Person]; [location, keyPerson] 1.94952

4. [City < Person]; [hometown] 1.91772

5. [City < Organisation < Person]; [city, managerClub] 1.85307

6. [Ãrea > Person]; [leaderName] 1.51637

7. [M̃unicipality > Person]; [leaderName] 1.42686

8. [City > P̃arty]; [leaderName] 1.36869

9. [C̃ounty > Person]; [leaderName] 1.26052

(DS 23) ?x/Person, designed, Brooklyn/Bridge

0. [Person < Bridge]; [architect] 2.72538

1. [Person < Building]; [architect] 0.96118

2. [Person > C̃ross < Bridge]; [birthPlace, crosses] 0.93530

3. [Person < HistoricBuilding]; [architect] 0.75010

4. [Person > C̃ross < Bridge]; [deathPlace, crosses] 0.71094

5. [Person > S̃tructure]; [birthPlace] 0.58969

6. [Person > S̃tructure]; [deathPlace] 0.52647

7. [Person > S̃tructure]; [restingPlace] 0.36793

8. [MilitaryPerson > C̃ross < Bridge]; [birthPlace, crosses] 0.34848

9. [MilitaryPerson > C̃ross < Bridge]; [deathPlace, crosses] 0.34442

(DS 24) Karakoram/Mountain Range, highest, ?x/Place

0. [MountainRange > Place]; [highestPlace] 6.16786

1. [MountainRange > Place]; [highestMountain] 4.28680

2. [MountainRange < Mountain < Place]; [mountainRange, highestPlace] 3.30910

3. [MountainRange > PopulatedPlace]; [highestPlace] 3.30468

4. [MountainRange > PopulatedPlace]; [highestMountain] 3.26433

5. [MountainRange > Mountain > Place]; [highestPlace, locatedInArea] 2.96694

6. [MountainRange > P̃osition < Place]; [highestPosition, highestPosition] 2.75530

7. [MountainRange > L̃ocation]; [highestPlace] 2.62088

8. [MountainRange > L̃ocation]; [highestMountain] 2.53732

9. [MountainRange > Mountain > PopulatedPlace]; [highestPlace, locatedInArea] 2.30226

(DS 25) Forbes/Company, homepage, ?x/Website

0. [Company > H̃omepage]; [homepage] 4.55716

1. [Organisation > H̃omepage]; [homepage] 2.37773

2. [Company < Broadcast > H̃omepage]; [owningCompany, homepage] 2.02823

3. [Company < Company > H̃omepage]; [parentCompany, homepage] 1.92544

4. [Company > Company > H̃omepage]; [parentCompany, homepage] 1.92544

5. [Company < TelevisionShow > H̃omepage]; [company, homepage] 1.77151

6. [Õrganization > H̃omepage]; [homepage] 1.52705

7. [EducationalInstitution > H̃omepage]; [homepage] 1.47976

8. [Company < Website]; [author] 1.35299

9. [Non-ProfitOrganisation > H̃omepage]; [homepage] 1.25509

(DS 26) ?x/Company, belongs to , computer software/Industry

0. [Company > Ĩndustry]; [industry] 12.47260

1. [Company > Ĩndustry]; [location] 9.32496

2. [Company > Ĩndustry]; [service] 6.49195

3. [Company > Company > Ĩndustry]; [parentCompany, industry] 5.24804

4. [Company < Company > Ĩndustry]; [parentCompany, industry] 5.24804

5. [Organisation > Ĩndustry]; [industry] 5.11098
6. [Organisation > Ĩndustry]; [location] 3.88074
7. [Company > Ĩndustry < Company]; [industry, industry] 3.42729
8. [Organisation > Ĩndustry]; [service] 3.11470
9. [Õrganization > Ĩndustry]; [industry] 2.70199

(DS 27) Bruce Carver/Person, died by, ?x/Disease

0. [Person > Disease]; [deathCause] 6.78930
1. [C̃hild > Disease]; [deathCause] 2.17679
2. [S̃pouse > Disease]; [deathCause] 1.57630
3. [Person > C̃ause]; [deathPlace] 0.96695
4. [Person > C̃ause]; [deathCause] 0.82517
5. [MilitaryPerson > C̃ause]; [deathPlace] 0.60706
6. [Person > G̃enus]; [deathPlace] 0.52145
7. [S̃pouse < Person > Disease]; [deathPlace, deathCause] 0.32262
8. [S̃pouse < Person > Disease]; [spouse, deathCause] 0.31655
9. [C̃hild > C̃ause]; [deathPlace] 0.28296

(DM 1) *y/Actors, , Charmed/TV Show || *y/Actors, official website, ?x/Websites

0. [Actor < TelevisionShow]; [starring] || [Actor > H̃omepage]; [homepage] 2.89037
1. [Actor < TelevisionShow]; [showJudge] || [Actor > H̃omepage]; [homepage] 2.38788
2. [Actor < TelevisionShow]; [producer] || [Actor > H̃omepage]; [homepage] 2.11075
3. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [starring, homepage] 2.01833
4. [Actor < TelevisionShow]; [director] || [Actor > H̃omepage]; [homepage] 1.91802
5. [Actor < TelevisionShow]; [composer] || [Actor > H̃omepage]; [homepage] 1.87708
6. [Actor < TelevisionShow]; [starring] || [Actor < Work > P̂age]; [starring, @pages] 1.75027
7. [Actor < TelevisionShow]; [showJudge] || [Actor < Work > H̃omepage]; [starring, homepage] 1.66744
8. [Actor < TelevisionShow]; [starring] || [Actor < Work > H̃omepage]; [writer, homepage] 1.60407
9. [Actor < TelevisionShow]; [starring] || [Actor > H̃omepage < WorldHeritageSite]; [homepage, homepage] 1.57738

(DM 2) Richard Nixon/Person, daughter, *y/Person || *y/Person, married to, ?x/Person

0. [Person > Person]; [child] || [Person > Person]; [spouse] 8.12454
1. [Person > Person]; [child] || [Person < Person]; [spouse] 8.12449
2. [Person < Person]; [child] || [Person > Person]; [spouse] 8.12449
3. [Person < Person]; [child] || [Person < Person]; [spouse] 8.12444
4. [Person > Person]; [parent] || [Person > Person]; [spouse] 6.74569
5. [Person > Person]; [parent] || [Person < Person]; [spouse] 6.74565
6. [Person < Person]; [parent] || [Person > Person]; [spouse] 6.74563
7. [Person < Person]; [parent] || [Person < Person]; [spouse] 6.74559

8. [Person > C̃hild < Person]; [child, child] || [Person > Person]; [spouse] 6.38246

9. [Person > C̃hild < Person]; [child, child] || [Person < Person]; [spouse] 6.38242

(DM 3) Egypt/Country, largest city, ?x/City || Egypt/Country, capital, ?x/City

0. [Country < City]; [country] || [Country > City]; [capital] 6.90104

1. [Country > City]; [largestCity] || [Country > City]; [capital] 6.71764

2. [Country < PopulatedPlace > City]; [country, largestCity] || [Country > City]; [capital] 6.30455

3. [Country < Place > City]; [country, location] || [Country > City]; [capital] 5.77375

4. [Country < EducationalInstitution > City]; [country, city] || [Country > City]; [capital] 5.58313

5. [Country < City]; [country] || [Country < PopulatedPlace > City]; [country, capital] 5.25393

6. [Country > City]; [largestCity] || [Country < PopulatedPlace > City]; [country, capital] 5.11430

7. [Country < PopulatedPlace > City]; [country, largestCity] || [Country < PopulatedPlace > City]; [country, capital] 4.79980

8. [Country < City]; [country] || [Country < City]; [country] 4.64968

9. [Country > City]; [largestCity] || [Country < City]; [country] 4.52611

(DM 4) Garry Marshall/Director, directed, ?x/Movies || ?x/Movies, starring, Julia Roberts/Actress

0. [D̃irector < Film]; [director] || [Film > Actor]; [starring] 7.50218

1. [D̃irector < Film]; [director] || [Film > Comedian]; [starring] 5.57283

2. [D̃irector < Film]; [director] || [Film > AdultActor]; [starring] 5.04387

3. [D̃irector < Film]; [director] || [Film > C̃omposer]; [starring] 4.91115

4. [D̃irector < Film]; [director] || [Film > Artist]; [starring] 4.81937

5. [D̃irector < Film]; [starring] || [Film > Actor]; [starring] 4.66071

6. [D̃irector < Film]; [director] || [Film > Wrestler]; [starring] 4.15413

7. [D̃irector < Film]; [director] || [Film > Boxer]; [starring] 3.93192

8. [M̃anager < Film]; [director] || [Film > Actor]; [starring] 3.92532

9. [D̃irector < Film]; [director] || [Film > Athlete]; [starring] 3.87065

(DM 5) Manhattan/Bridge, , *y/Type || ?x/Bridge, ,*y/Type

0. [Bridge > T̃ype]; [type] || [Bridge > T̃ype]; [type] 5.38549

1. [Bridge > T̃ype]; [type] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 4.25581

2. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > T̃ype]; [type] 4.25581

3. [Bridge > T̃ype]; [type] || [Bridge > BodyOfWater > T̃ype]; [crosses, location] 3.67317

4. [Bridge > BodyOfWater > T̃ype]; [crosses, location] || [Bridge > T̃ype]; [type] 3.67317

5. [Bridge > T̃ype]; [type] || [S̃tructure > T̃ype]; [type] 3.51981

6. [S̃tructure > T̃ype]; [type] || [Bridge > T̃ype]; [type] 3.51981

7. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 3.36309

8. [Bridge > BodyOfWater > T̃ype]; [crosses, type] || [Bridge > BodyOfWater > T̃ype]; [crosses, location] 2.90267

9. [Bridge > BodyOfWater > T̃ype]; [crosses, location] || [Bridge > BodyOfWater > T̃ype]; [crosses, type] 2.90267

(DM 6) ?x/telecommunication organization, located in, Belgium/Country

0. [Organisation > Country]; [country] 8.92118

1. [Organisation > Country]; [location] 8.27889

2. [Õrganization < Settlement > Country]; [country, country] 5.37715

3. [Õrganization > Country]; [country] 5.36800

4. [Organisation > S̃tate]; [country] 5.32525

5. [Organisation > R̃egion]; [country] 5.09427

6. [Organisation > S̃tate]; [location] 4.88023

7. [Organisation > S̃tate]; [state] 4.71649

8. [Organisation > Country]; [state] 4.70191

9. [Organisation > Country]; [regionServed] 4.66259

REFERENCES

[1] E. Agichtein and L. Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the Fifth ACM Conference on Digital Libraries*, pages 85–94, New York, NY, USA, 2000. ACM.

[2] E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo. *sem 2013 shared task: Semantic textual similarity. In **SEM 2013: The Second Joint Conference on Lexical and Computational Semantics*. Association for Computational Linguistics, 2013.

[3] E. Agirre, M. Diab, D. Cer, and A. Gonzalez-Agirre. Semeval-2012 task 6: a pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, pages 385–393. Association for Computational Linguistics, 2012.

[4] I. Androutsopoulos, G. Ritchie, and P. Thanisch. Natural language interfaces to databases – an introduction. *Natural Language Engineering*, 1(01):29–81, 1995.

[5] P. Auxerre and R. Inder. Masque modular answering system for queries in english - user's manual. Technical report, Artificial Intelligence Applications Institute, University of Edinburgh, 1986.

[6] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni. Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

[7] M. Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. The wacky wide web:

A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226, 2009.

[8] C. Bizer, T. Heath, and T. Berners-Lee. Linked data-the story so far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(3):1–22, 2009.

[9] D. Bollegala, Y. Matsuo, and M. Ishizuka. Measuring semantic similarity between words using web search engines. In *Proc. WWW*, 2007.

[10] S. Brin. Extracting patterns and relations from the world wide web. In *International Workshop on The World Wide Web and Databases*, WebDB '98, pages 172–183, London, UK, UK, 1999. Springer-Verlag.

[11] A. Budanitsky and G. Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, 2006.

[12] J. Bullinaria and J. Levy. Extracting semantic representations from word cooccurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526, 2007.

[13] R. Bunescu and R. Mooney. A shortest path dependency kernel for relation extraction. In *Conf. on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 724–731, 2005.

[14] C. Burgess, K. Livesay, and K. Lund. Explorations in context space: Words, sentences, discourse. *Discourse Processes*, 25:211–257, 1998.

[15] M. J. Cafarella, C. Re, D. Suciu, O. Etzioni, and M. Banko. Structured querying of web text data: A technical challenge. In *In CIDR*, pages 225–234, 2007.

[16] H. Chen, M. Lin, and Y. Wei. Novel association measures using web search with double checking. In *Proc. COLING/ACL 2006*, pages 1009–1016, 2006.

[17] K. Church and P. Hanks. Word association norms, mutual information and lexicography. In *Proc. 27th Annual Conf. of the ACL*, pages 76–83, 1989.

[18] P. Cimiano, P. Haase, and J. Heizmann. Porting natural language interfaces between domains: an experimental user study with the ORAKEL system. In *Proc. 12th Int. Conf. on Intelligent User Interfaces*, pages 180–189. ACM, 2007.

[19] T. Coelho, P. Pereira Calado, L. Vieira Souza, B. Ribeiro-Neto, and R. Muntz. Image retrieval using multiple evidence ranking. *IEEE Trans. on Knowl. and Data Eng.*, 16(4):408–417, 2004.

[20] S. Cohen, J. Mamou, Y. Kanza, and Y. Sagiv. XSearch: A Semantic Search Engine for XML. In *VLDB*, 2003.

[21] M. J. Collins. *Head-driven statistical models for natural language parsing*. PhD thesis, University of Pennsylvania, 1999.

[22] J. R. Curran and M. Moens. Improvements in automatic thesaurus extraction. In *Proc. Workshop on Unsupervised Lexical Acquisition*, pages 59–66, Philadelphia, PA, USA, 2002.

[23] I. Dagan, S. Marcus, and S. Markovitch. Contextual word similarity and estimation from sparse data. In *Proceedings of ACL-93*, pages 164–171, Columbus, Ohio, 1993.

[24] D. Damljanovic, M. Agatonovic, and H. Cunningham. FREyA: An interactive way of querying Linked Data using natural language. In *1st Workshop on Question Answering over Linked Data*, pages 125–138, 2011.

[25] Dbpedia 3.6 downloads. http://wiki.dbpedia.org/Downloads36?v=ebv, 2011.

[26] M.-C. de Marneffe, B. MacCartney, and C. D. Manning. Generating typed dependency parses from phrase structure parses. In *5th Int. Conf. on Language Resources and Evaluation*, pages 449–454, 2006.

[27] B. Dolan, C. Quirk, and C. Brockett. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04. Association for Computational Linguistics, 2004.

[28] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[29] O. Erling and I. Mikhailov. RDF support in the virtuoso DBMS. In *Networked Knowledge - Networked Media*, volume 221, pages 7–24. Springer, 2009.

[30] T. Finin. *Semantic Interpretation of Compound Nominals*. PhD thesis, University of Illinois, 1980.

[31] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131, 2002.

[32] L. A. Flemming. Putnam's Word Book. http://www.gutenberg.org/files/13188/13188-8.txt, 1913.

[33] A. Freitas, J. de Oliveira, S. O'Riain, E. Curry, and J. P. da Silva. Querying linked data using semantic relatedness: A vocabulary independent approach. In *16th Int. Conf. Applications of Natural Language to Information Systems*, pages 40–51. Springer, 2011.

[34] W. Gale, K. Church, and D. Yarowsky. One sense per discourse. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 233–237, Harriman, NY, 1992.

[35] R. Ge and R. J. Mooney. A statistical semantic parser that integrates syntax and semantics. In *Proc. of CoNLL-05*, pages 9–16, Ann Arbor, MI, 2005.

[36] G. Grefenstette. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, Boston, USA, 1994.

[37] B. Grosz, D. Appelt, P. Martin, and F. Pereira. Team: an experiment in the design of transportable natural-language interfaces. *Artificial Intelligence*, 32(2):173–243, 1987.

[38] L. Han and T. Finin. UMBC webbase corpus. http://ebiq.org/r/351, 2013.

[39] L. Han, T. Finin, and A. Joshi. Schema-free structured querying of dbpedia data. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2090–2093. ACM, 2012.

[40] L. Han, T. Finin, P. McNamee, A. Joshi, and Y. Yesha. Improving Word Similarity by Augmenting PMI with Estimates of Word Polysemy. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1307–1322, 2013.

[41] L. Han, A. L. Kashyap, T. Finin, J. Mayfield, and J. Weese. UMBC_EBIQUITY-CORE: Semantic Textual Similarity Systems. In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics*. Association for Computational Linguistics, June 2013.

[42] Z. Harris. *Mathematical Structures of Language*. Wiley, New York, USA, 1968.

[43] M. Hart. Project gutenberg electronic books. http://www.gutenberg.org/wiki/Main_Page, 1997.

[44] G. Hendrix, E. Sacerdoti, D. Sagalowicz, and J. Slocum. Developing a natural language interface to complex data. *TODS*, 3(2):105–147, 1978.

[45] D. Higgins. Which statistics reflect semantics? rethinking synonymy and word similarity. In *Proceedings of the International Conference on Linguistic Evidence*, pages 265–284, Tübingen, Germany, 2004.

[46] D. Hindle. Noun classification from predicate-argument structures. In *Proc. Annual Meeting of the ACL*, pages 268–275, Pittsburg PA, 1990.

[47] V. Hristidis and Y. Papakonstantinou. Discover: Keyword search in relational databases. In *VLDB*, pages 670–681, 2002.

[48] M. Jarmasz and S. Szpakowicz. Roget's thesaurus and semantic similarity. In *Proc. Int. Conf. on Recent Advances in Natural Language Processing*, pages 212–219, Borovets, Bulgaria, 2003.

[49] J. Jiang and D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. Int. Conf. on Research in Computational Linguistics*, 1997.

[50] N. Kaji and M. Kitsuregawa. Using hidden markov random fields to combine distributional and pattern-based word clustering. In *Proc. of the 22nd Int. Conf. on Computational Linguistics*, pages 401–408, 2008.

[51] N. Kambhatla. Combining lexical, syntactic and semantic features with maximum entropy models. In *Proceedings of ACL*, 2004.

[52] B. Katz and J. Lin. Selectively using relations to improve precision in question answering. In *Proc. of the EACL-2003 Workshop on Natural Language Processing for Question Answering*, 2003.

[53] D. Kauchak and R. Barzilay. Paraphrasing for automatic evaluation. In *HLT-NAACL '06*, pages 455–462, 2006.

[54] I. Kaur and A. J. Hornof. A comparison of LSA, wordnet and PMI-IR for predicting user click behavior. In *Proc. ACM CHI 2005 Human Factors in Computing Systems Conf.*, pages 51–60, New York, 2005. ACM Press.

[55] T. Landauer and S. Dumais. A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. In *Psychological Review, 104*, pages 211–240, 1997.

[56] Y. Lei, V. Uren, and E. Motta. Semsearch: A search engine for the semantic web. In *15th Int. Conf. on Knowledge Engineering and Knowledge Management*, pages 238–245. Springer, 2006.

[57] J. P. Levy and J. A. Bullinaria. Learning lexical properties from word usage patterns: Which context words should be used? In *Sixth Neural Computation and Psychology Workshop: Connectionist Models of Learning, Development and Evolution*, pages 273–282, London, 2001. Springer.

[58] J. P. Levy, J. A. Bullinaria, and M. Patel. Explorations in the derivation of semantic representations from word co-occurrence statistics. *South Pacific Journal of Psychology*, 10:99–111, 1998.

[59] Y. Li, Z. Bandar, and D. McLean. An approach for measuring semantic similarity be-

tween words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):871–882, 2003.

[60] Y. Li, H. Yang, and H. Jagadish. Constructing a generic natural language interface for an xml database. In *EDBT*, pages 737–754, 2006.

[61] Y. Li, C. Yu, and H. V. Jagadish. Schema-free XQuery. In *VLDB*, pages 72–83, 2004.

[62] D. Lin. Automatic retrieval and clustering of similar words. In *Proc. 17th Int. Conf. on Computational Linguistics*, pages 768–774, Montreal, CN, 1998.

[63] D. Lin. Dependency-based evaluation of minipar. In *Workshop on the Evaluation of Parsing Systems*, 1998.

[64] D. Lin. An information-theoretic definition of similarity. In *Proc. Int. Conf. on Machine Learning*, 1998.

[65] D. Lin and P. Pantel. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360, 2001.

[66] V. Lopez, M. Pasin, and E. Motta. Aqualog: An ontology-portable question answering system for the semantic web. In *Proc. European Semantic Web Conf.*, pages 546–562, 2005.

[67] V. Lopez, V. Uren, M. Sabou, and E. Motta. Cross Ontology Query Answering on the Semantic Web: An Initial Evaluation. In *Proc. 5th Int. Conf. on Knowledge Capture*. ACM, 2009.

[68] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, US, 1999.

[69] C. T. Meadow. *Text Information Retrieval Systems*. Academic Press, Inc., 1992.

[70] D. Metzler, S. Dumais, and C. Meek. Similarity measures for short segments of text. In *Proceedings of the 29th European conference on IR research*, pages 16–27. Springer-Verlag, 2007.

[71] R. Mihalcea, C. Corley, and C. Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In *Proc. 21st National Conf. on Artificial Intelligence*, pages 775–780, Boston, Mass., 2006.

[72] G. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):41, 1995.

[73] G. Miller and W. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

[74] S. Mohammad, B. Dorr, and G. Hirst. Computing word-pair antonymy. In *Proc. Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-2008)*, October 2008.

[75] S. Mohammad and G. Hirst. Distributional measures of concept-distance: A task oriented evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 35–43, 2006.

[76] R. Navigli. Meaningful clustering of senses helps boost word sense disambiguation performance. In *Proc. 21th Int. Conf. on Computational Linguistics*, pages 105–112, Sydney, Australia., 2006.

[77] S. Pado and M. Lapata. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199, 2007.

[78] P. Pantel and D. Lin. Discovering word senses from text. In *Proc. ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, pages 613–619, Edmonton, Canada, 2002.

[79] M. Patel, J. A. Bullinaria, and J. P. Levy. Extracting semantic representations from large text corpora. In *Fourth Neural Computation and Psychology Workshop: Connectionist Representations*, pages 199–212, London, 1997. Springer.

[80] A.-M. Popescu, O. Etzioni, and H. Kautz. Towards a theory of natural language interfaces to databases. In *Proc. 8th Int. Conf. on Intelligent User Interfaces*, pages 149–157. ACM, 2003.

[81] Poweraqua question answering system. http://poweraqua.open.ac.uk:8080/poweraqualinked.

[82] Qald-1 open challenge test phase: Evaluation results. http://bit.ly/QALD11.

[83] 1st workshop on question answering over linked data. http://www.sc.cit-ec.uni-bielefeld.de/qald-1, 2011.

[84] R. Rapp. Word sense discovery based on sense descriptor dissimilarity. In *Proc. 9th Machine Translation Summit*, pages 315–322, 2003.

[85] P. Resnik. Using information content to evaluate semantic similarity. In *Proc. 14th Int. Joint Conf. on Artificial Intelligence*, 1995.

[86] P. Resnik. Semantic similarity in a taxonomy: An information based measure and its application to problems of ambiguity in natural language. *Journal of Aritificial Intelligence Research*, 11:95–130, 1999.

[87] T. Rose, M. Stevenson, and M. Whitehead. The reuters corpus volume 1 - from yesterday's news to tomorrow's language resources. In *In Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 29–31, 2002.

[88] H. Rubenstein and J. Goodenough. Contextual correlates of synonymy. *CACM*, 8(10):627–633, 1965.

[89] M. Sahami and T. D. Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 377–386. ACM, 2006.

[90] F. Saric, G. Glavas, M. Karan, J. Snajder, and B. D. Basic. Takelab: systems for measuring semantic text similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, pages 441–448. Association for Computational Linguistics, 2012.

[91] Y. Shinyama and S. Sekine. Preemptive information extraction using unrestricted relation discovery. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 304–311, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[92] F. Smadja. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19(1):143–178, 1993.

[93] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in twitter to improve information filtering. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 841–842. ACM, 2010.

[94] Stanford WebBase project. http://dbpubs.stanford.edu:8091/ testbed/doc2/WebBase/.

[95] A. Termehchy and M. Winslett. Using structural information in xml keyword search effectively. *TODS*, 36(01):4:1–4:39, 2011.

[96] E. Terra and C. L. A. Clarke. Frequency estimates for statistical word similarity measures. In *Proc. Human Language Technology and North American Chapter of the ACL Conf. 2003*, pages 244–251, 2003.

[97] B. Thompson and F. Thompson. Introducing ask, a simple knowledgeable system. In *1st Conf. on Applied Natural Language Processing*, pages 17–24, 1983.

[98] K. Toutanova, D. Klein, C. Manning, W. Morgan, A. Rafferty, and M. Galley. Stanford log-linear part-of-speech tagger. http://nlp.stanford.edu/software/tagger.shtml, 2000.

[99] K. Toutanova, D. Klein, C. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL*, pages 173–180, 2003.

[100] T. Tran, P. Cimiano, S. Rudolph, and R. Studer. Ontology-based Interpretation of Keywords for Semantic Search. In *Proc. of the 6th ISWC*, pages 523–536. Springer, 2007.

[101] Trueknowledge (evi) online system. http://trueknowledge.com/.

[102] W. Tunstall-Pedoe. True knowledge: Open-domain question answering using structured knowledge and inference. *AI Magazine*, 31(3):80–92, 2010.

[103] P. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proc. 12th European Conf. on Machine Learning*, pages 491–502, 2001.

[104] P. Turney, M. Littman, J. Bigham, and V. Shnayder. Combining independent modules to solve multiple-choice synonym and analogy problems. In *Proc. RANLP-2003*, pages 482–489, 2003.

[105] G. Varelas, E. Voutsakis, P. Raftopoulou, E. Petrakis, and E. Milios. Semantic similarity methods in wordnet and their application to information retrieval on the web. In *Proc. ACM Workshop on Web Information and Data Management*, Bremen, Germany, 2005.

[106] C. Wang, M. Xiong, Q. Zhou, and Y. Yu. PANTO: A Portable Natural Language Interface to Ontologies. In *Proc. Semantic Web: Research and Applications*, pages 473–487. Springer, 2007.

[107] J. Weeds and D. Weir. Finding and evaluating sets of nearest neighbours. In *Proc. 2nd Int. Conf. on Corpus Linguistics*, Lancaster, UK, 2003.

[108] J. E. Weeds. *Measures and applications of lexical distributional similarity*. PhD thesis, University of Sussex, September 2003.

[109] W. Woods, R. Kaplan, and B. Nash-Webber. The lunar sciences natural language information system. Technical Report 2378, BBN, Cambridge MA, 1972.

[110] H. Wu and M. Zhou. Synonymous collocation extraction using translation information. In *Proc. 41st Annual Meeting of the Association for Computational Linguistics*, pages 120–127, Sapporo, Japan, 2003.

[111] Y. Xu and Y. Papakonstantinou. Efficient Keyword Search for Smallest LCAs in XML Databases. In *SIGMOD*, pages 527–538, 2005.

[112] D. Yang and D. M. Powers. Automatic thesaurus construction. In *Proc. 31st Australasian Conf. on Computer Science*, volume 74, pages 147–156, 2008.

[113] D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the ACL*, pages 189–196, 1995.