A Three-Tiered Approach to Integrating Information Retrieval and Natural Language Processing [*]

Carl Weir, Tim Finin, Robin McEntire, and Barry Silk
Unisys Center for Advanced Information Technology
Paoli Pennsylvania

May, 1991

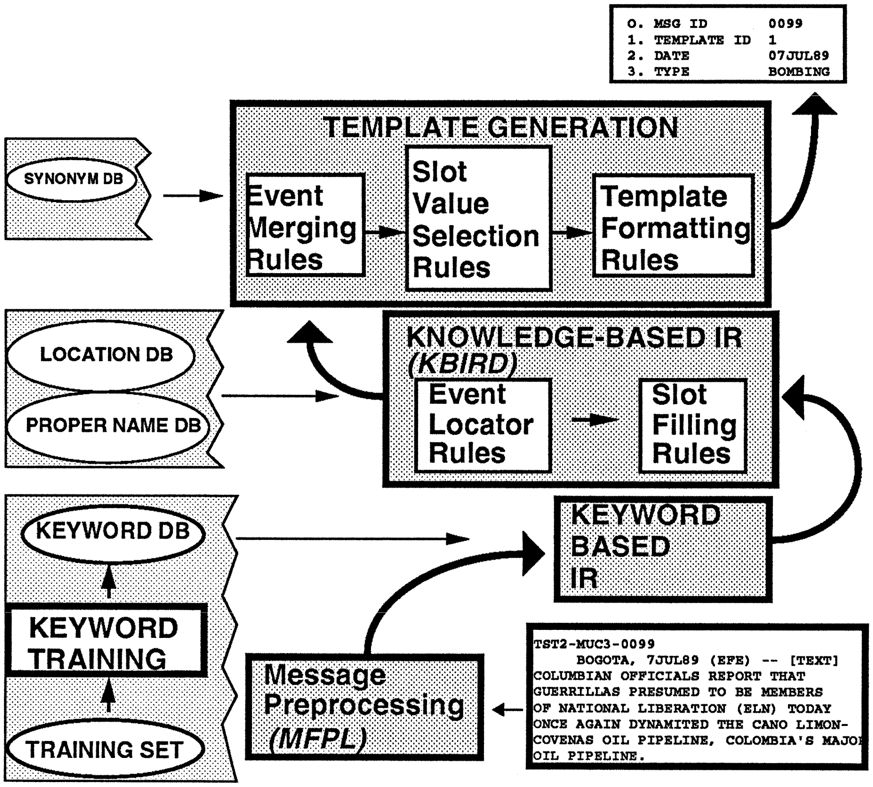
Abstract

This paper presents a three-tiered approach to text processing in which a novel and quite powerful knowledge-based form of information retrieval plays a central role. This approach was used in our participation in the the NOSC-sponsored *Third Message Understanding Conference* (MUC-3). Our three-tiered approach to text processing can be defined in terms of three processing components: a keyword analysis system that is used to predict the occurrence of terrorist act descriptions; the knowledge-based information retrieval system KBIRD which is used to instantiate templates for the terrorist act descriptions detected by the keyword analysis system; and a natural language processing system called PUNDIT, which KBIRD provides with key segments of text on which to perform a detailed linguistic analysis in order to extract information about grammatical and thematic roles.

Submitted to the American Society for Information Science *Workshop on Language and Information Processing* to be held on October 27, 1991 in Washington, D.C.

Point of contact: Carl Weir, Unisys Center for Advanced Information Technology, P.O. Box 517, Paoli PA 19139, weir@prc.unisys.com, voice: 215-648-2369 , fax: 215-648-2288

[*] This work was partially supported by DARPA Contract MDA-903-89-C-0041.



```
0. MSG ID          0099
1. TEMPLATE ID     1
2. DATE            07JUL89
3. TYPE            BOMBING
```

```
TST2-MUC3-0099
     BOGOTA, 7JUL89 (EFE) -- [TEXT]
COLUMBIAN OFFICIALS REPORT THAT
GUERRILLAS PRESUMED TO BE MEMBERS
OF NATIONAL LIBERATION (ELN) TODAY
ONCE AGAIN DYNAMITED THE CANO LIMON-
COVENAS OIL PIPELINE, COLOMBIA'S MAJOR
OIL PIPELINE.
```

Figure 2: The system flow chart for the version of system used in the second evaluation text for MUC-3

Our three-tiered approach to text processing can be defined in terms of three main processing components: a keyword analysis system that is used to predict the occurrence of terrorist act descriptions; the knowledge-based information retrieval system KBIRD which is used to instantiate templates for the terrorist act descriptions detected by the keyword analysis system; and the PUNDIT natural language processing system [2], which KBIRD provides with key segments of text on which to perform a detailed linguistic analysis in order to extract information about grammatical and thematic roles.

In the actual system used in the MUC-3 evaluation, the PUNDIT component was not integrated. The organization of the this system is shown in Figure 2. Here the following main software components are described:

- **Message Preprocessor** – The initial stage takes the raw text and performs low-level preprocessing, recognizing the formatted portions (e.g. text date) and producing output appropriate for Prolog.

- **Keyword Based IR** – An initial analysis based on key words and word pairs which predicts the existance of zero or more event classes in the text.

- **KBIRD** – A rule-based system which locates instances of predicted event classes in the text, instantiates candidate templates for each, and fills in appropriate slots.

- **Template Generator** – A module which (i) identifies unique events described in the text and merges all templates describing them (ii) selects the best candidate slot

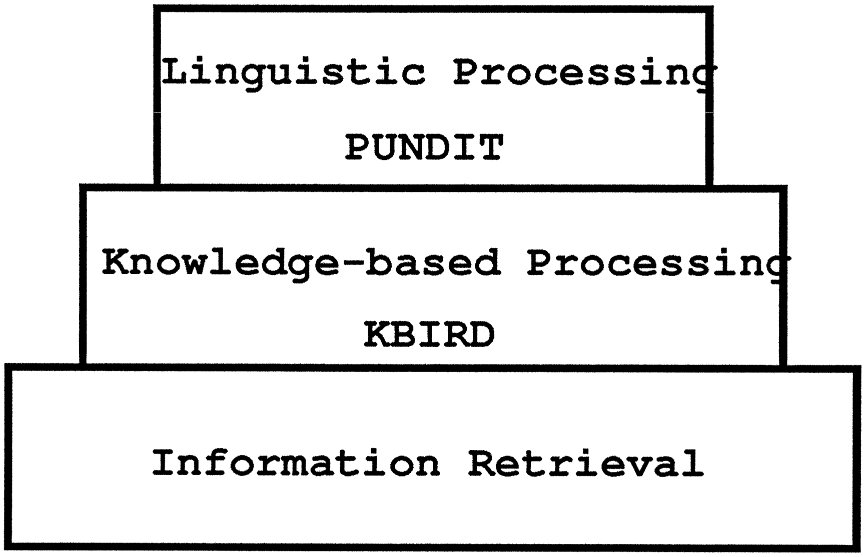


Figure 1: Our three-tiered approach to text processing can be defined in terms of three main processing components: a keyword analysis system; KBIRD, a knowledge-based component; and PUNDIT, a natural language processing system.

1 Introduction

This paper presents a three-tiered approach to text processing in which a novel and quite powerful knowledge-based form of information retrieval plays a central role. This approach was used in our participation in the the NOSC-sponsored *Third Message Understanding Conference* (MUC-3) [7]. Research groups participating in the MUC-3 conference must evaluate the performance of their text processing systems on a black-box, template (database record) generation task. To perform this task, a text processing system must extract information about different types of terrorist acts from newspaper articles and radio broadcasts. Relevant data about terrorist acts including when and where they occurred, who perpetrated them, what weapons were used, who or what were the targets, and so forth, constitute the content of templates used to represent them in a database. The knowledge-based form of information retrieval which plays a key role in our three-tiered approach allows us to define an interesting level of text analysis that falls somewhere between what is possible with standard IR techniques and deep linguistic analysis.

Our three-tiered approach to text processing can be defined in terms of three processing components: a keyword analysis system that is used to predict the occurrence of terrorist act descriptions; the knowledge-based information retrieval system KBIRD which is used to instantiate templates for the terrorist act descriptions detected by the keyword analysis system; and a natural language processing system called PUNDIT [2, 4], which KBIRD provides with key segments of text on which to perform a detailed linguistic analysis in order to extract information about grammatical and thematic roles. In the remainder of this paper, these components are described in more detail.

2 Approach and System Description

fillers for each slot and (iii) produces the final template output form.

Each of these components makes use of various databases and knowledge-bases.

2.1 Message Pre-processing

The message pre-processing component is a special low-level processor which parses the text into its component parts and generates output in a form compatible with the KBIRD rule processing system (i.e., as a set of Prolog terms). This processor is a special C program which was generated using an *Application Specific Language* called MFPL (*Message Format Processing Language*) [6]. MFPL was specifically designed as a high-level language for processing the formatted portions of electronic messages. In addition to producing a representation of the text in Prolog terms, this module identifies and encodes sentence boundaries, paragraph boundaries, and the standard formatted portions of the text (e.g., date, time, location, etc.).

2.2 Keyword Analysis

The keyword analysis component of the Unisys MUC-3 system predicts when various types of terrorist acts (bombings, murders, kidnappings, and so forth) have been referred to in a text. The probability of an act of a given type having occurred is determined by a search for words, word stems, and pairs of words and pairs of word stems, that are associated with types of acts.[1] The probability of a such a word (or word stem, or word pair or stem pair) occurring in a text for which an act of a given type is associated is determined as follows.

The frequency of presence for a given word W (or word stem ...) in texts for which a terrorist act of a given type T occurs is computed ($f(W,T)$), as is the presence of the word in any text at all in the complete corpus ($f(W,C)$). The probability of the word appearing in texts for which a terrorist act of a given type occurs

$$\frac{f(W,T)}{f(T,C)}$$

and the probability of the word occurring in any text

$$\frac{f(W,C)}{|C|}$$

are calculated, and these two values are used to determine the conditional probability of the word (or word stem ...) predicting the given type of terrorist act.

$$\mathcal{P}(W,T) = \frac{\left(\frac{f(W,T)}{f(T,C)}\right)}{\left(\frac{f(W,C)}{|C|}\right)},$$

Only words with relatively high probabilities of predicting a given type of terrorist act are searched for in a text, and words that do not occur frequently enough in the text corpus based on some empirically-derived threshold are not used.

Training. A database of key words, two-word phrases, word stems and two-stem phrases was compiled from the DEV corpus using a collection of GAWK scripts. After some experimentation, we decided not to use the word stem and stem-pair data in the second test. Currently, an event class, T, is predicted for a text if it contains any single word, W, where $\mathcal{P}(W,T) > .65$ or if it contains two words W_1 and W_2 where $\mathcal{P}(W_1,T) > .55$ and $\mathcal{P}(W_2,T) > .55$.

[1] The keyword analysis system uses a rule-driven word-stemmer based on one developed by Chris Paice (Landcaster, UK) [3].

Antecedent Format	Operator Description
A ˆ *B*	*A* is contiguous with *B*.
A , *B*	*A* is in the same text as *B*.
A .. *B*	*A* is in the same sentence as *B*.
A.. > *B*	*A* is in the same sentence as and precedes *B*.
A ... *B*	*A* is in the same paragraph as *B*.
A ..+ *B*	*A* is in the same region as *B*.

Figure 3: Some of the KBIRD operators used in the MUC-3 application

2.3 KBIRD

Once a set of terrorist acts have been predicted, the task of generating templates describing those acts falls to the knowledge-based information retrieval component called KBIRD.

KBIRD is a rule-based system for concept-spotting in free text [9, 8, 10]. KBIRD rules are forward-chaining horn clauses whose antecedents are constituents discovered and recorded in a chart data structure and whose consequents are newly inferred constituents—concepts (or facts)—to be added to the chart. The antecedents and consequents of KBIRD rules can include arbitrary Prolog goals just as in *Definite Clause Grammars* [5].

It is tempting to think of a set of KBIRD rules as implementing a kind of bottom up chart parser, but there are several interesting differences. One distinctive feature is that the concepts that KBIRD rules infer are associated with a specific region of text, a region which is the maximal cumulative span of the regions of text associated with each expression in a given rule's antecedent. Moreover, these regions can be explicitly reasoned about by subsequent KBIRD rules.

In typical natural language parsers, there is an implicit constraint that adjacent constituents in a rule must be realized by contiguous strings of text in the input. KBIRD allows one to write rules which specify other constraints on the relative positions of the strings which realize rule constituents. The antecedent of a KBIRD rule may consist of several facts (words or concepts) that are the arguments of operators of the following sort. New operators are easy to define in KBIRD. Figure 3 gives a sample of some of the KBIRD operators used in this application

KBIRD rules are compiled into a combination of Prolog backward chaining rules and forward chaining rules in Pfc [1]. A simple optimizer is applied to the output of this compilation process to improve performance. KBIRD has many additional features which are inherited from the Pfc rule language, such as the ability to write non-monotonic rules which specify that no occurrence of a certain constituent or concept be found in a given region.

Some examples of KBIRD rules are shown in Figure 4. The first rule states that if the wordstem "MURDER*" has been found in the text, then a fact should be added to the factbase stating that a potential_murder_event has been found. The second rule illustrate KBIRD's ability to recognize phrases, asserting that if the string "ARMY OF NATIONAL LIBERATION" is discovered, a fact should be added to the factbase stating that a terrorist organization exists in the text at the same location as the string. The remaining four rules contain examples of operations on concepts derived from the text. The third rule, for ex-

```
1.  "MURDER*" ⟹ potential_murder_event.
2.  "ARMY" ^ "OF" ^ "NATIONAL" ^ "LIBERATION"
       ⟹ terrorist_organization.
3.  terrorist_event(E) ..  potential_victim(V)
       ⟹ victim(E,V).
4.  bombing_event ⟹ terrorist_event(bombing).
5.  peasant ⟹ potential_victim.
6.  government_person ⟹ potential_victim.
```

Figure 4: Examples of KBIRD rules

ample, asserts that if a terrorist event E is found in the same sentence as a potential victim V, then a fact should be added to the factbase indicating that V is the actual victim of E.

Several additional features of the KBIRD rule language should be mentioned, all of which appear in the following rule which attempts to find individual perpetrators in the text:

```
generic_perpetrator(A)@P,
[~unlikely_perpetrator(Name)],
{get_full_text_at_loc(P,Name)}
     ==> potential_ind_perpetrator(A, Name).
```

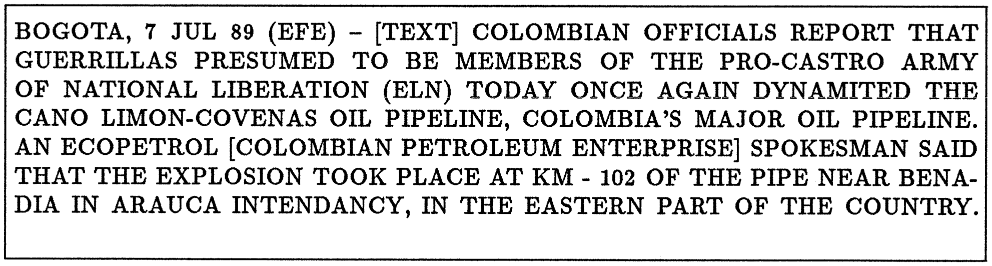
In the first antecedent, the location of the text region matching **generic_perpetrator(A)** is bound to the logic variable P with the @ operator. This allows the location to be explicitly constrainted later in the rule. If a rule antecedent is enclosed in square brackets, as is the second one, then its location is ignored. This condition also shows the use of the tilde as the negation operator. Thus, this second antecedent specifies that it is not the case that Name has been determined to be an "unlikely perpetrator" anywhere else in the text. The final antecedent in this rule is encased in curly brackets, which indicates that it is a Prolog constraint which must be met.

2.4 Template Generator

The *Template Generator* has three tasks – to select the actual templates to be produced as output, to choose between candidate slot fillers if more than one has been found, and to print the template in the proper format.

Template Selection. If no event has been predicted, then an "irrelevant template" is created. If several events of the same type have been created, the template generator will attempt to merge them using a set of heuristics which hypothesize that two event descriptions refer to the same event. Some of the general heuristics used for merging events of the same class are:

- Merge two events if there is a significant overlap in the text regions found by the *event locator rules.*
- Merge two events if they share human targets whose scores are above a certain threshold.
- Merge two events if they share physical targets whose scores are above a certain threshold.



BOGOTA, 7 JUL 89 (EFE) - [TEXT] COLOMBIAN OFFICIALS REPORT THAT GUERRILLAS PRESUMED TO BE MEMBERS OF THE PRO-CASTRO ARMY OF NATIONAL LIBERATION (ELN) TODAY ONCE AGAIN DYNAMITED THE CANO LIMON-COVENAS OIL PIPELINE, COLOMBIA'S MAJOR OIL PIPELINE. AN ECOPETROL [COLOMBIAN PETROLEUM ENTERPRISE] SPOKESMAN SAID THAT THE EXPLOSION TOOK PLACE AT KM - 102 OF THE PIPE NEAR BENADIA IN ARAUCA INTENDANCY, IN THE EASTERN PART OF THE COUNTRY.

Figure 5: The first two sentences of text TST2-MUC3-0099 from the MUC-3 second test set

Slot Filler Selection. After merging events, the template generator must select the final slot filler values. The KBIRD rules which propose slot fillers attach a score (an integer between and 100) to each candidate which represents the system's confidence in that value. If multiple candidate fillers exist for a given template, several general heuristics are used to select among them:

- Candidate slot values with scores below a given threshold are dropped from consideration.
- A set of synonymous expressions are dropped in favor of their canonical expression.
- If one candidate expression is a substring of another, then the shorter one is dropped.
- A generic description (e.g., *vehicles*) is dropped in favor of one or more subsumed ones (e.g., *ambulance, truck*).
- If a slot can only take a single value then the candidate receiving the highest value is selected.

2.5 PUNDIT

The PUNDIT natural language processing system has been under development at Unisys for the last five years and is capable of performing a detailed linguistic analysis of an input text. Unlike KBIRD, PUNDIT abstracts away from the actual strings used to convey information in a text at the very beginning of its analysis process by determining to which syntactic properties and domain concepts the lexical items in the text correspond. These syntactic properties and domain concepts are then processed without much attention being paid to their physical location in the text. In KBIRD, on the other hand, everything that is manipulated, even concepts that have been asserted, are explicitly associated with regions of text.

A key capability that the deeper linguistic processing of PUNDIT can provide is the determination of the grammatical and thematic roles of expressions in a text. Thus, it can determine that in the sentence *"Castellar is the second mayor that has been murdered in Colombia in the last 3 days"* that *Castellar* is the subject of the copular verb in the matrix clause, and that *Castellar* should inherit properties asserted of the predicate nominal argument. It can also recognize the passive voice of the relative/subordinate clause headed by *that* and thus that it is Castellar that has been murdered (as the second mayor) in Columbia.

It would be possible to build a KBIRD rulebase that performs the sort of detailed linguistic analysis now being performed by PUNDIT. Merging KBIRD and PUNDIT in this way would minimize the complications F1of integrating the text analyses that they perform. However, such a merger would very likely reduce the modularity of the three-tiered approach to text processing that we have been following.

Sent	Type	Prob	Keys
1	murder	63	[63,PRESUMED,5:6]
1	bombing	83	[83,DYNAMITED,22:23] [58,PIPELINE,27:28] [58,PIPELINE,32:33]
2	bombing	81	[62,ECOPETROL,35:36] [81,EXPLOSION,45:46]
2	bombing	89	[71,COLOMBIAN,PETROLEUM,37:39] [71,PETROLEUM,ENTERPRISE,38:40] [89,THE,EXPLOSION,44:46] [85,EXPLOSION,TOOK,45:47]
3	bombing	83	[83,CRUDE,85:86] [58,PIPELINE,91:92]
5	bombing	83	[83,CRUDE,150:151] [58,PIPELINE,156:157]
6	bombing	62	[62,ECOPETROL,173:174]
8	bombing	83	[83,DYNAMITED,201:202] [58,PIPELINE,205:206] [55,DAMAGE,236:237] [62,ECOPETROL,239:240]

Figure 6: Keyword analysis predicts a murder event and a bombing event for text TST2-MUC3-0099.

3 An Example

In this section, we will show the result of processing text TST2-MUC3-0099 of the MUC-3 test corpus. The first two sentences of this text are shown in Figure 5. The initial keyword analysis predicts two event classes – murder with a probability of 63% and bombing with a probability of 89%. Figure 6 shows the particular words and word pairs which gave rise to these predicted event types. The last column in this table contains triples consisting of a probability, a word or two-word phrase and the location in the text. Given our current thresholds, the prediction of a *murder* event was judged to be too weak for further consideration. Figure 7 shows the *bombing* template produced for this text.

The *bombing* template generated contained three errors – in slots five, seven and sixteen. For slot five (perpetrator individuals), the system generated a '-' (the response for "no information") rather than "GUERRILLAS". The KBIRD processor did, in fact, identify the correct answer with a probability of 95%, but it wasn't extracted by the template generator (an apparent bug). For slot seven (perpetrator confidence) we did not key off the keyword *presumed* found in the text, which should have lead to the correct fill of *suspected or accused by authorities.* Instead, we used the default fill for the slot of *reported as fact.* For slot sixteen (location of incident), KBIRD correctly deduced a location of COLOMBIA with ARAUCA as an INTENDANCY, as well as ARAUCA as a RIVER, with equal likelihood. The template generator arbitrarily chose the RIVER answer.

4 Conclusions

The value of our three-tiered approach is two-fold. First, the domain in which we are currently working is so well-defined that a deep linguistic analysis is rarely needed. Using linguistic analysis sparingly and perhaps not at all in some texts provides a dramatic improvement in processing time. Second, in the MUC-3 evaluation task we have discovered

Slot	*Description*	*Filler*
0	message id	TST2-MUC3-0099
1	template id	1
2	date of incident	07 JUL 89
3	type of incident	BOMBING
4	category of incident	TERRORIST ACT
5	perpetrator: id of indiv	-
6	perpetrator: id of org(s)	”PRO-CASTRO ARMY OF NATIONAL LIBERATION”
7	perpetrator: confidence	REPORTED AS FACT: ”PRO-CASTRO ARMY OF NATIONAL LIBERATION”
8	physical target: id(s)	”OIL PIPELINE”
9	physical target: total num	1
10	physical target: type(s)	ENERGY: ”OIL PIPELINE”
11	human target: id(s)	-
12	human target: total num	-
13	human target: type(s)	-
14	target: foreign nation(s)	-
15	instrument: type(s)	*
16	location of incident	COLOMBIA: ARAUCA (RIVER)
17	effect on physical target	SOME DAMAGE: ”OIL PIPELINE”
18	effect on human target(s)	-

Figure 7: The first template for text TST2-MUC3-0099

that a small amount of modeling effort, i.e., writing KBIRD rules, produces a significant improvement in our ability to extract pertinent information. Since KBIRD is a forward chaining rule-driven methodology, the creation, modification and removal of rules is a very easy and intuitive process.

The three-tiered approach of combining traditional information retrieval and linguistic analysis techniques with the type of analysis that our knowledge-based information retrieval system, KBIRD, provides offers significant advantages to solving common text processing problems. The modularity of this approach allows us to utilize advances made in keyword analysis and NLP technology with relative ease.

References

[1] Tim Finin, Rich Fritzson, and Dave Matuzsek. Adding forward chaining and truth maintenance to prolog. In *Fifth IEEE Conference on Artificial Intelligence Application*, pages 123–130, March 1989.

[2] L. Hirschman, M. Palmer, J. Dowding, D. Dahl, M. Linebarger, R. Passonneau, F.-M. Lang, C. Ball, and C. Weir. The PUNDIT natural-language processing system. In *AI Systems in Government Conf.* Computer Society of the IEEE, March 1989.

[3] Chris Paice. Another stemmer. *SIGIR Forum*, Fall 1990.

[4] Rebecca Passonneau, Carl Weir, Tim Finin, and Martha Palmer. Integrating natural language processing and knowledge based processing, 1990. Eighth National Conference on Artifical Intelligence, AAAI 90.

[5] Fernando C.N. Pereira and David H.D. Warren. Definite clause grammars for language analysis—a survey of the formalism and a comparison with augmented transition networks. *Artificial Intelligence*, 13(3):231–278, 1980.

[6] Bob Pollack. Message format processing language. Manual, Unisys Center for Advanced Information Technology, August 1989. Version 2.1.

[7] Beth M. Sundheim. Third message understanding conference (MUC-3): Phase 1 status report. In *Proceedings of the DARPA Speech and Natural Language Workshop*, February 1991.

[8] Carl Weir, Tim Finin, Robin McEntire, and Barry Silk. The Unisys MUC-3 text understanding system. In *Proceedings of the Third Message Understanding Conference*, San Diego, May 1991.

[9] Carl Weir, Tim Finin, Barry Silk, Marcia Linebarger, and Robin McEntire. Knowledge-based strategies for robust text-understanding. Poster paper presented at the Eighth Annual Intelligence Community AI/Advance Computing Symposium, March 1991.

[10] Carl Weir, Robin McEntire, Barry Silk, and Tim Finin. MUC-3 test results and analysis. In *Proceedings of the Third Message Understanding Conference*, San Diego, May 1991.