Identifying and Isolating Text Classification Signals from Domain and Genre Noise for Sentiment Analysis

by

Justin Martineau

Advisory Committee:
Dr. Tim Finin, Chair/Advisor
Dr. Tim Oates
Dr. Marie desJardins
Dr. James Mayfield
Dr. Akshay Java

© Copyright by
Justin Martineau
20011

ABSTRACT

Title of dissertation: Identifying and Isolating
Text Classification Signals
from Domain and Genre Noise
for Sentiment Analysis
Justin Martineau

Dissertation directed by: Tim Finin
Department of Computer Science

Sentiment analysis is the automatic detection and measurement of sentiment in text segments by machines. This problem is generally divided into three tasks: a sentiment detection task, a topic detection task, and a sentiment measurement task. The first task attempts to determine whether the author is being objective or whether they are expressing a value judgment on the topic. The second task attempts to determine the topic of the sentiment. The third task attempts to determine whether the author approves or disapproves of the topic and by how much.

The main difficulty in solving these tasks arises from noise in the author's sentiment signal that is caused by the variety of different domains (topics) and genres (communication media). The limitless scope of possible domains, plus the effort require to hand label data for sentiment analysis tasks, implies a lack of labeled data for any given domain. To further exacerbate this problem, any of these domains can occur in any known genre, creating further noise in the communicator's

sentiment signal. Genres that are either freely available or have high data volumes, including web pages, blogs, news feeds, Twitter posts, Facebook updates, and SMS messages are very interesting. These data sources cover a broad range of topics and are primarily unedited, thus noise and domain dependence are very important issues for sentiment analysis.

Sentiment analysis techniques are widely applicable to both government and private sector problems. On the government side, textual sentiment analysis in blogs could help identify terrorists, terrorist supporters, and potential suicide victims. These techniques can also be used to influence people around the world, and to measure the effectiveness of advertising campaigns. Textual sentiment analysis can be used for market research, financial investments, and politics.

To support these kinds of applications on blogs, news, and other text messages, I developed and evaluated techniques to identify and rank transferable discriminative sentimental terms, and used them on other domains of interest to classify the author's sentiment about the topic of his writing. These techniques for a four step process. The first step is to determine the sentimental orientation of terms and score their strength in a set of known domains using Delta IDF. The second step is to determine and score how well these terms should transfer to the target domain using my Domain Independence Verification Algorithm (DIVA). The third step is to create a weight vector for the target domain using the Delta IDF weights and the DIVA scores. The final step is to transform documents into term frequency weighted document vectors and then classify them using the result of their dot product with the target domain's weight vector.

Delta IDF works well because I designed it to correct certain problems with the existing best practice. Term frequency times inverse document frequency (TFIDF) is a very successful mainstream practice in the field of Information Retrieval (IR). It was designed to reduce the importance of common English words while simultaneously boosting the importance of context-specific words. It is effective at identifying topical words for a document, but it was not designed to identify discriminative vocabulary for specific classification tasks. Nevertheless, many sentiment analysis researchers still use TFIDF scores as their initial term frequency weights for machine learning algorithms. I designed Delta TFIDF [25] to correct this mismatch between design goals. Classifying documents by their dot product with an in-domain Delta IDF weight vector produces statistically better results than using a Support Vector Machine (SVM).

DIVA is an effective technique for identifying and scoring terms that will transfer well to a target domain because it relies upon my newly developed concept of sentimental domain independence. Sentimental domain independence is the degree to which a term's sentimental bias remains unchanged in multiple domains. This definition is useful because it exposes a statistical or counting problem. Before this dissertation, researchers lacked a clear, concrete definition of domain independence, a measure of domain independence, and statistics about domain independence. This definition is an important part of my solution. While it is hard to determine what effects an increase or decrease in document level frequency counts will have on a given term's sentimental bias, it is much easier to determine if a term's sentimental bias will remain largely unchanged in the target domain if we can verify that it

changes little in other domains. The definition, statistics, and algorithm for sentimental domain-independent terms are some of my major contributions to the field of domain adaption and sentiment analysis.

I evaluated this approach as a classification problem. I evaluated Delta IDF using ten-fold cross-validation on different domains and genres against state-of-the-art SVM baselines. These results are statistically superior using two-tailed tests. Other researchers have since published further tests showing Delta IDF works well on additional domains and genres [33]. My results with DIVA on multiple domains against state-of-the-art domain adaptation techniques were clearly superior. In fact, DIVA is so powerful that it can outperform in-domain Delta IDF baselines by leveraging greater amounts of out of domain data. Not only is it possible to accurately transfer models between domains, it is advantageous.

My research made both algorithmic and theoretical contributions to the fields of domain adaption and sentiment analysis. First, I provided algorithms to discover and weight discriminative classification task specific features within a domain. Second, I produced algorithms to score how well these features transfer to a new target domain. Third, I laid out a general theory for the kinds of information and the types of noise they produce in text classification problems. Fourth, I defined sentimental domain independence and statistically described it. This dissertation gives future readers a firm theoretical foundation and practical algorithms to build on top of for a wide array of identified classification problems and for future research in the field.

Table of Contents

List of Figures iv

List of Abbreviations viii

1 Introduction 1
1.1 Thesis Statement . . . 1
1.2 Theoretical Foundation . . . 1
1.2.1 Classification Signal: Sentiment Signal . . . 2
1.2.2 Domain Signal . . . 3
1.2.3 Genre Signal . . . 5
1.3 Problem Definition . . . 6
1.4 Scope . . . 10
1.5 Contributions . . . 11

2 Motivation and Applications 13

3 State of the Art 17
3.1 Observations about Sentiment . . . 17
3.2 Challenges . . . 19
3.3 Sentiment Analysis . . . 22
3.3.1 Vector Space Analysis . . . 22
3.3.2 Vector Space Analysis Shortfalls . . . 24
3.4 Domain Adaptation . . . 28

4 Delta IDF: Identifying In-domain Problem Specific Information 33
4.1 Introduction . . . 33
4.2 Approach . . . 36
4.3 Evaluation . . . 44
4.3.1 Delta IDF weights for SVMs . . . 44
4.3.2 Dot Product Classification . . . 48
4.4 Discussion and Conclusions . . . 54

5 DIVA: Domain Independence Verification Algorithm for Sentiment Analysis 56
5.1 Introduction . . . 56
5.2 Related Work . . . 58
5.3 Dataset Description . . . 62
5.3.1 Collection and Tokenization . . . 62
5.3.2 Sentimental Domain Dependence Statistics . . . 63
5.4 Approach . . . 64
5.5 Evaluation . . . 68
5.5.1 Single Domain Adaption . . . 68
5.5.2 Domain-Independent Classification . . . 70
5.6 Future Work . . . 75
5.7 Conclusion . . . 77

6 Novel Feature Discovery in Unfamiliar Domains 79
6.1 Introduction and Related Work 79
6.2 Experimental Design 80
6.3 Results and Discussion 83
6.4 Future Work 85
6.5 Conclusion 86

7 Genre Adaptation 87
7.1 Introduction and Related Work 88
7.2 Genre Adaptation Experiments 89
7.2.1 TREC Topic Sentiment Prediction 89
7.2.2 Irish News Financial Prediction 91
7.3 Discussion and Future Work 92
7.4 Conclusion 94

8 Conclusion 95
8.1 Contributions 95
8.2 Future Work 98
8.3 Summary 99

A In-Domain Delta IDF Model Visualizations 101

B Novel Features Visualizations 114

Bibliography 127

List of Figures

4.1 Top Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red. 44

5.1 Log count of the number of features that are primarily negative in X domains and primarily positive in Y domains. Features are increasingly negative going to the right corner along the X axis, increasingly positive going to the left corner along the Y axis, and more widely used among the domains going down. 64

5.2 Top Delta IDF features for Books. Font size indicates relative strength. Positive features are in blue. Negative features are in red. 66

5.3 Top Delta IDF features found in the union of Books, DVDs, and Electronics with the same sentimental orientation. Font size indicates relative strength. Positive features are in blue. Negative features are in red. 66

6.1 Top domain-specific Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red. 83

A.1 Top Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red. 101

A.2 Top Delta IDF features found in the **automotive** domain. Positive Features are in blue. Negative Features are in red. 102

A.3 Top Delta IDF features found in the **baby products** domain. Positive Features are in blue. Negative Features are in red. 102

A.4 Top Delta IDF features found in the **beauty products** domain. Positive Features are in blue. Negative Features are in red. 103

A.5 Top Delta IDF features found in the **books** domain. Positive Features are in blue. Negative Features are in red. 103

A.6 Top Delta IDF features found in the **cameras** domain. Positive Features are in blue. Negative Features are in red. 104

A.7 Top Delta IDF features found in the **cell phones and cellular service** domain. Positive Features are in blue. Negative Features are in red. 104

A.8 Top Delta IDF features found in the **computer and video games** domain. Positive Features are in blue. Negative Features are in red. 105

A.9 Top Delta IDF features found in the **DVDs** domain. Positive Features are in blue. Negative Features are in red. 105

A.10 Top Delta IDF features found in the **electronics** domain. Positive Features are in blue. Negative Features are in red. 106

A.11 Top Delta IDF features found in the **gourmet food** domain. Positive Features are in blue. Negative Features are in red. 106

A.12 Top Delta IDF features found in the **groceries** domain. Positive Features are in blue. Negative Features are in red. 107

A.13 Top Delta IDF features found in the **health and personal fitness** domain. Positive Features are in blue. Negative Features are in red. 107

A.14 Top Delta IDF features found in the **jewelry and watches** domain. Positive Features are in blue. Negative Features are in red. 108

A.15 Top Delta IDF features found in the **kitchen and housewares** domain. Positive Features are in blue. Negative Features are in red. 108

A.16 Top Delta IDF features found in the **magazines** domain. Positive Features are in blue. Negative Features are in red. 109

A.17 Top Delta IDF features found in the **music** domain. Positive Features are in blue. Negative Features are in red. 109

A.18 Top Delta IDF features found in the **musical instruments** domain. Positive Features are in blue. Negative Features are in red. 110

A.19 Top Delta IDF features found in the **office products** domain. Positive Features are in blue. Negative Features are in red. 110

A.20 Top Delta IDF features found in the **outdoor living** domain. Positive Features are in blue. Negative Features are in red. 111

A.21 Top Delta IDF features found in the **software** domain. Positive Features are in blue. Negative Features are in red. 111

A.22 Top Delta IDF features found in the **sports and outdoors** domain. Positive Features are in blue. Negative Features are in red. 112

A.23 Top Delta IDF features found in the **tools and hardware** domain. Positive Features are in blue. Negative Features are in red. 112

A.24 Top Delta IDF features found in the **toys and games** domain. Positive Features are in blue. Negative Features are in red. 113

A.25 Top Delta IDF features found in the **video** domain. Positive Features are in blue. Negative Features are in red. 113

B.1 Top domain specific Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red. 114

B.2 Top domain specific Delta IDF features found in the **automotive** domain. Positive Features are in blue. Negative Features are in red. 115

B.3 Top domain specific Delta IDF features found in the **baby products** domain. Positive Features are in blue. Negative Features are in red. 115

B.4 Top domain specific Delta IDF features found in the **beauty products** domain. Positive Features are in blue. Negative Features are in red. 116

B.5 Top domain specific Delta IDF features found in the **books** domain. Positive Features are in blue. Negative Features are in red. 116

B.6 Top domain specific Delta IDF features found in the **cameras** domain. Positive Features are in blue. Negative Features are in red. 117

B.7 Top domain specific Delta IDF features found in the **cell phones and cellular service** domain. Positive Features are in blue. Negative Features are in red. . . 117

B.8 Top domain specific Delta IDF features found in the **computer and video games** domain. Positive Features are in blue. Negative Features are in red. . . . 118

B.9 Top domain specific Delta IDF features found in the **DVDs** domain. Positive Features are in blue. Negative Features are in red. 118

B.10 Top domain specific Delta IDF features found in the **electronics** domain. Positive Features are in blue. Negative Features are in red. 119

B.11 Top domain specific Delta IDF features found in the **gourmet food** domain. Positive Features are in blue. Negative Features are in red. 119

B.12 Top domain specific Delta IDF features found in the **groceries** domain. Positive Features are in blue. Negative Features are in red. 120

B.13 Top domain specific Delta IDF features found in the **health and personal fitness** domain. Positive Features are in blue. Negative Features are in red. 120

B.14 Top domain specific Delta IDF features found in the **jewelry and watches** domain. Positive Features are in blue. Negative Features are in red. 121

B.15 Top domain specific Delta IDF features found in the **kitchen and housewares** domain. Positive Features are in blue. Negative Features are in red. 121

B.16 Top domain specific Delta IDF features found in the **magazines** domain. Positive Features are in blue. Negative Features are in red. 122

B.17 Top domain specific Delta IDF features found in the **music** domain. Positive Features are in blue. Negative Features are in red. 122

B.18 Top domain specific Delta IDF features found in the **musical instruments** domain. Positive Features are in blue. Negative Features are in red. 123

B.19 Top domain specific Delta IDF features found in the **office products** domain. Positive Features are in blue. Negative Features are in red. 123

B.20 Top domain specific Delta IDF features found in the **outdoor living** domain. Positive Features are in blue. Negative Features are in red. 124

B.21 Top domain specific Delta IDF features found in the **software** domain. Positive Features are in blue. Negative Features are in red. 124

B.22 Top domain specific Delta IDF features found in the **sports and outdoors** domain. Positive Features are in blue. Negative Features are in red. 125

B.23 Top domain specific Delta IDF features found in the **tools and hardware** domain. Positive Features are in blue. Negative Features are in red. 125

B.24 Top domain specific Delta IDF features found in the **toys and games** domain. Positive Features are in blue. Negative Features are in red. 126

B.25 Top domain specific Delta IDF features found in the **video** domain. Positive Features are in blue. Negative Features are in red. 126

List of Abbreviations

NLP	Natural Language Processing
IR	Information Retrieval
ML	Machine Learning
BoW	Bag of Words
LSA	Latent Semantic Analysis
SVD	Singular Value Decomposition
EVD	Eigenvalue Decomposition
SVM	Support Vector Machine
EM	Expectation Maximization
PMI	Pointwise Mutual Information
TFIDF	Term Frequency Inverse Document Frequency
DIVA	Domain Independence Verification Algorithm

Chapter 1

Introduction

1.1 Thesis Statement

This dissertation shows how to significantly improve sentiment analysis accuracy for domains with little or no labeled data by characterizing and categorizing textual expressions into three fundamental types of information: genre, domain, and problem specific knowledge; it describes new algorithms that identify, isolate, and transfer only the problem-specific information.

1.2 Theoretical Foundation

This theoretical framework is an enabling contribution that describes how a message is composed of multiple sources of information. It explains how these different information sources affect a communicator's message. These distinctions allow me to better isolate problem specific information by better understanding the sources of non-problem specific information.

In any written work, authors send three types of signals. These signals are genre signals, domain signals, and classification task signals. A given word or phrase can contribute information to any combination of these three types of signals. This thesis provides methods to identify, characterize, and isolate the problem specific

signal.

When a person decides to communicate he must choose a communications medium to use. This communications medium is the genre he or she has selected to use. The genre selected can be either a speech or a text genre. This research is limited to textual genres. Textual genres include: blogs, news papers, forums, Twitter, Facebook, books, SMS messages, and user reviews. Every authors' message has a general topic area. This topic area is the domain the author has chosen to write in.

1.2.1 Classification Signal: Sentiment Signal

The problem or classification task signal is a direct result of the classification problem being addressed. This task signal is what the reader or the machine learning researcher is interested in. The task signal is not necessarily, but is often, the main point the author is trying to convey.

In this thesis the problem is sentiment analysis. Sentiment analysis is the automatic detection and measurement of sentiment in text segments by machines. In sentiment analysis the goal is to determine the author's opinion about his topic. Authors can either approve of, disapprove of, have mixed feelings about, or express no opinion about, the topic of their writing.

To simplify this problem, most researchers break it into two problems. They preprocess their documents using subjectivity detection to ensure that only opinionated documents are considered. Then they use polarity detection to determine

if those documents are primarily positive or negative. This results in casting the mixed sentiment examples into the class they are closest to. Most polarity detection tasks use a numeric score to rank not only which sentiment is expressed, but also how strongly that sentiment is expressed. Using subjectivity and then polarity detection will reveal how strongly opinionated any given document is.

Text classification problems like subjectivity and polarity detection are focused upon discriminative words or phrases that indicate the document's class. Words and phrases exhibit some combination of genre, domain, and problem signals. For each word or phrase it is important to measure the strength of these three signals. In most domains and genres only a small fraction of terms have any bearing on the author's expressed sentiment. If a term does not express a clear problem-specific signal then it is useless for that problem. After identifying classification task-specific terms in a given dataset it is important to understand how they interact with the genre signal and the domain signal.

1.2.2 Domain Signal

The domain signal is primarily determined by the subject or topic of the message. When writing on a subject a certain set of topical words tend to be used. Similar topics will use a similar set of topical words, while dissimilar topics will use far fewer of these words.

Terms exhibit a degree of domain independence. Some terms are more likely to appear for a given topic than for other topics, while other terms are far more

generic and will appear in almost every topic regardless of how similar the topics are. In a movie review domain-specific terms will tend to be focused upon things like the actors, directors, special effects, characters, pacing, and plot. Similarly, in a book review the domain signal will center on things like the characters, pacing, and plot. However, books do not have actors, or special effects, but the author can be thought of as roughly analogous to the director. This highlights an important point for domain adaptation research: With each signal, the meaning of some things transfers well, some things get dropped, and some things change.

The most important thing about these fluctuations is their affect upon the task-specific signal. If a term is not present in the target domain then the information transferred is impotent: it does neither harm nor good. If the term was not present in the source domain then there is no information about it to transfer, leaving no information about its problem-specific value. In Chapter 6 I show how to identify and determine the polarity of these terms using their association to other task-specific features that have been transferred. These terms add very little classification accuracy to the original model because this process looks for new terms that highly agree with the original model transferred. These terms have other uses because they are associated with the general sentiment being expressed and specific to the domain at hand. Consequently, they tend to be facets of the domain that the author approves of or disapproves of. In short, they tell why the author felt the way they did about the topic.

If the term is present in both domains then it will either keep the same general problem specific value or it will not. If its value remains substantially unchanged

then it is safe to use the term as is. However, if its value radically changes then something else must be done. Determining which case is in effect is crucial, and actually very simple; just look at how constant the term's problem-specific signal is among multiple other domains. Later I will describe an algorithm called DIVA (Domain Independence Verification Algorithm) to do this. By carefully selecting verification domains that are like the target domain it is possible to identify and rank a higher degree of domain-specific terms for the problem.

1.2.3 Genre Signal

The genre signal is primarily defined by the data source or medium that the author uses to communicate. Writing styles change with the publishing medium. Books, news articles, blog posts, surveys, product reviews, status updates, and tweets have very different styles. Length, stop-word and near stop-word usage, personal pronoun usage, abbreviations, formatting, degree of editing, and the tendency to stay on topic are affected by the publishing medium. This creates a unique signal that I call the genre signal.

The different styles exhibited in these genres have varying degrees of similarity. Books are similar to news articles because they are both highly edited, but have vastly different lengths and personal pronoun usage. Furthermore, books, due to their length tend to span a greater topical breadth. Tweets on the other hand are very short, unedited, and highly abbreviated, but tend to stay on topic.

Slight genre differences can create great variance in the problem specific sig-

nal. The degree of misspellings, which is commonly associated with the degree of editing, affects term based methods by varying noise. Stop word and personal pronoun frequency affects the frequency of many sentimental indicators. Vector space representations for term frequencies treat abbreviations and the abbreviated word as different terms. This produces different levels of noise when the genre signal changes. The amount written on a topic is a sentimental indicator, genre specific changes in the average are important. Off topic, or off domain, information should always be discarded.

1.3 Problem Definition

When there is enough information about the problem in the domain and genre that the researcher is working on, then there are two main approaches to sentiment detection and other text analysis problems: the statistical approach and the grammatical parsing approach. Statistical sentiment analysis approaches tend to follow a supervised machine learning framework. In this framework text data is manually labeled by humans into a set of distinct classes. A supervised learning algorithm will use a subset of these labeled documents to build a model of the training data. This model contains the knowledge the algorithm learned when examining the example data points and their associated class labels. In conjunction with the classification algorithm this model can be used to label documents. A model's accuracy is evaluated by how closely the labels it predicts match the human being's class label for a document that the model was not trained on. Sentiment classifiers built on

data from a single domain and genre pair typically provide just under 90% accuracy on that pair of domain and genre; this is not perfect, but is sufficient for most applications.

The statistical approach is better suited to work on informal genres than the grammatical parsing approach. Since the vast amount of freely available data on the web is informal, selecting the statistical approach is more appropriate. While grammatical approaches can take advantage of known properties and rules pertaining to language structure, part of speech information, and theta roles for words (the number of and type of arguments that a verb takes) [12], they encounter greater difficulty with free text because it is often informally written, poorly structured, and rife with spelling and grammatical errors. This noise frequently violates rule-based approaches, causing parsers to fail or return incorrect results.

The problem with statistical machine learning approaches is that while they do much better with informal text, they tend to generalize poorly when used on data that is different from the data they were trained on, because meaning and sentiment are context dependent. For example, the line, "Jolting, heart pounding, you're in for one hell of a bumpy ride!" means two very different things when said about a car as opposed to when said about a movie. Documents that have been labeled for sentiment in one domain, and sentimental features extracted from those documents in that domain, do not necessarily share the same sentiment in another domain.

The limitless scope of possible domains creates a lack of labeled data for any given domain. To further exacerbate this problem, any of these domains can occur in any known genre further dividing the available pool of labeled data for the specific

domain and genre pair the researcher is working on. It is infeasible to gather and label enough data to train supervised models that are reasonably accuracy for every pair of domain and genre. Even if such a pool of data were to be made available building and tuning a classifier for every domain and genre requires an unrealistic amount of processing power. Furthermore, selecting which classifier to use given a document would pose additional challenges. The field of transductive transfer learning seeks to address these research challenges.

Domain adaptation, also known as transductive transfer learning, is the machine learning task of building a model for a target data distribution given labeled data for the problem from a different data distribution.

Cross domain training was an early attempt to accomplish this. Cross domain training is a common technique used to compensate for a lack of existing labeled training data. This technique involves training a classifier for the target dataset on labeled data from another dataset. Results are generally much lower than training on a data set from a similar sized labeled data set from the target domain. Experiments with SVMs show that similar domains show a decrease in accuracy of about ten percent, but dissimilar domains suffer even greater penalties dropping accuracy to just above random chance [48]. Using a mixture of multiple domains seems to provide mixed results based upon how well each individual domain does for cross domain training [1].

This dissertation reduces the transductive transfer learning problem down to two key questions:

1. What is the bias in the source domain(s) of the term to transfer?

2. From this bias what can be concluded about its bias in the target domain?

By posing the problem in this way domain adaptation researchers can profitably interact with machine learning researchers, since the field of machine learning provides numerous techniques to answer the first question. The second question is the defining problem for transfer learning. A simple baseline approach would assume that the item's bias will not change between the source and target domains. This assumption invariably leads to lackluster accuracy as the cross domain training approach described above illustrates.

Given that feature bias will change between domains and that the relative frequency of the feature between the source and target domains is observable, what can be concluded about the feature bias in the target domain? If the feature was biased in the source domain will it still be as biased in the target domain if it becomes less/more common? A simple examination of feature bias versus feature frequency on any sentiment analysis domain is sufficient to cast doubt upon this approach. TF and IDF are not very good predictors for feature bias strength. [26] [25] This point will be covered in greater detail when Delta IDF is described. Determining the bias of a feature in the target domain using the bias in the source domain is hard; it is much easier, and still quite useful, to determine if the bias for the feature is likely to change between the two domains.

My research shows how to make classifiers generalize better to data that is less like the original training data by isolating and removing noise caused by domain and

genre information. This allows me to produce general purpose sentiment models that are at least as accurate as single domain models even though my models were trained on data drawn from distributions than cover different general topic areas than the test distribution.

1.4 Scope

Sentiment analysis is the automatic detection and measurement of sentiment in text segments by machines. Given the growth of social media and the type of applications sentiment analysis is frequently used upon, there are many other fields that intersect with sentiment analysis. Therefore, it is important to specify the scope of my dissertation.

Improving classifier generalization is the most important and significant part of this thesis. Concretely, this dissertation focuses on using machine learning to solve the sentiment analysis task on domains with little or no labeled data. Improvements in sentiment analysis techniques are important, but improvements on classifier generalizability are more significant. Testing classifier generalization improvements for every classification problem is too much to evaluate. While I believe my approach will improve generalization accuracy for problems other than sentiment analysis, seriously advancing this claim requires more data and experiments than I am prepared to show.

The experiments on Delta IDF show that the techniques designed to improve sentiment polarity and subjectivity detection generalize well to a few other binary

textual classification problems. Please keep in mind that these other classification problems are not the main focus of the work. Instead take these results as indicators that my approach will improve generalization accuracy for many different types of classification problems. These results also help support the sentiment analysis experiments by showing that the approach works on other data sets and that the accurate sentiment analysis classification results are not a product of continually creating new techniques until one happens to randomly do well on the data at hand.

1.5 Contributions

This dissertation makes several discoveries about text analysis and domain adaptation that lead to concrete measures and algorithms, which are supported by classification results on the de facto standard evaluation datasets. These contributions are briefly described as follows:

1. Discoveries:

   (a) Textual expression is influenced by three different factors: genre, domain, and problem factors.

   (b) Defined sentimental domain and genre independence.

   (c) Two tasks are common to textual supervised machine learning based domain adaptation approaches: Determining source domain feature bias scores, and determining their validity scores for the target domain. Doing just these two tasks well is sufficient to achieve high accuracy. Exploiting the strong document level co-occurrence between domain-independent

and domain-specific terms is unnecessary and will not improve accuracy for document level classification beyond the level achieved by domain-independent features.

2. Measures/Algorithms:

   (a) Delta IDF - a technique to more accurately determine the polarity and bias of words and phrases for binary text classification problems including subjectivity detection and sentiment polarity classification.

   (b) DIVA - the first quantifiable measure of problem specific domain independence. Empirically demonstrated that using DIVA with a simple classification rule will produce out-of-domain sentiment polarity classification results that are at least as good as state of the art in-domain techniques.

3. Supporting Contributions:

   (a) Produced the strongest in-domain baseline for all 25 Amazon product categories of the de facto standard domain adaptation dataset.

   (b) Out-of-domain training with DIVA classification is more accurate than any published technique for the 25 category Amazon product review dataset.

Chapter 2

Motivation and Applications

Text analysis techniques are widely applicable to government, private sector, and research problems. On the government side, textual sentiment analysis in blogs could help identify terrorists, terrorist supporters, and suicide victims. These techniques can also be used to influence people around the world, and to measure the effectiveness of our advertising campaigns. On the private sector side, textual sentiment analysis can be used for market research, financial investments, and politics. Sentiment analysis is critical to creating human like AI, and very useful for human computer interaction problems.

In a recent study, the New York Police Department found that there is a "radicalization process in the West that drives 'unremarkable' people to become terrorists" [43] "The majority of these individuals started as 'unremarkable' - they had 'unremarkable' jobs, had lived 'unremarkable' lives and had little, if any criminal history" [43]. Furthermore, these individuals use "al-Qaeda as their inspiration and ideological reference point." At risk individuals are male, middle class, educated, and are skilled with computers, making blogs a natural place to share their ideas. This sharing and ideological reinforcement from like-minded individuals occurs immediately before they start planning an attack. It is during this stage that textual sentiment analysis tools can help identify potential terrorists.

Similarly, the government could use textual sentiment analysis to search for other at-risk individuals who are contemplating suicide. Many suicide victims announce their intentions to commit suicide long before they attempt to commit suicide. Some warning signs for suicide are "Talking about death, dying, or suicide", along with feelings of hopelessness, rage, anger, or anxiety [42], journal writing is also a preparatory behavior for suicide [41]. Journal style blogs are very common, and would presumably show these sentimental warning signs for suicide.

Attempts to influence the opinions and beliefs of people in other countries can benefit greatly from textual sentiment analysis. Knowing what the local people feel before you start a campaign to change popular sentiment and beliefs is very important. If another country were to influence us, it would be more effective if they were to dress their message up in American clothing, ideas, and cultural icons, which we feel positively about. Similarly, if we want to get others to accept our messages we need to put a local flavor on them, and associate them with things that the local people feel positively about. By using sentiment analysis on local blogs, newspapers, and other information sources, we can get a better picture of what local people feel. These same techniques can be used to determine how well our message has reached the target audience.

In the private sector, companies are very interested in determining what customers feel about their products. Instead of using costly surveys, companies could monitor blogs,since customers will gladly post their feelings and experiences with products they've bought or used. Textual sentiment analysis techniques can determine which products people like, and why. Take, for example, using the point wise

mutual information of words with known sentimental language to classify reviews in multiple domains [47]. Customers can also benefit from these techniques to help them find out what products other customers like without shifting through a large number of posts.

Consumer and investor sentiment affects stocks prices. Investors need to know what other investors and experts feel about stocks, companies, products, and share prices. Studies show that when investor sentiment is low, "subsequent returns are relatively high for small stocks, young stocks, high volatility stocks, unprofitable stocks, non-dividend-paying stocks, extreme growth stocks, and distressed stocks. When sentiment is high, on the other hand, these categories of stock earn relatively low subsequent returns" [2]. Knowing how investors feel in general about the market, and specifically about any given stock, is useful investment information. This information can be extracted from blog text using sentiment analysis techniques.

Sentiment detection tools for blogs and other social media, tailored to politics, allow politicians to better represent and lead people in today's society. Sentiment detection on blogs can augment existing opinion polling techniques. Researchers are already working on using forums and discussion boards as a data source for political sentiment detection. Political discussions in threaded forums and discussion boards display a different community structure than linked blog posts. Comment posters tend to quote their opposition, as opposed to bloggers, who link to friends and like minded individuals. [29] Understanding how the genre and domain interact with sentiment is a crucial part of this thesis.

Sentiment analysis is essential for creating intelligent agents. Even skeptical

researchers admit that emotions "can be useful in organisms or machines which sometimes require very rapid reactions to occur faster than normal processes of perception, reasoning, deliberation, and planning," and that these reactions "can be triggered by a relatively stupid, but largely trainable, pattern recognition system" [44]. However, they deny the necessity of emotion claiming that emotions would not be necessary if other systems "could operate faster, and with more complete information" [44]. These researchers gloss over fundamental aspects of reality, neglecting to mention that the world is only partially observable and that they must compete with other agents in the world. These agents will already have spent large amounts of processing power to formulate plans of which the current agent is unaware of and must react quickly to. Furthermore, these agents will eventually need to interact with real people in emotional states, the ability to detect and analyze these states will play a factor in how successful those agents are.

While it might be possible to build an intelligent agent completely devoid of emotion, it would not seem very human. Humans want to interact with machines at a human level. Interacting with computers that understand sentiment improves productivity. Users who could express their feelings of frustration interactively with a computer through a conversational text agent designed to show sympathy could continue to use their software longer before stopping due to fatigue [19].

Chapter 3

State of the Art

A lot of research has gone into determining the sentiment of movie, book, and product reviews due to the availability of data. New datasets in the political domain have appeared that promise to add another deep line of work to the field. This research, as well as my own investigations, have revealed several key challenges and useful observations pertaining to sentiment analysis. Before discussing any specific piece of research, an overview of the many challenges and interesting observations is in order.

3.1 Observations about Sentiment

1. Sentiment has differing levels of intensity. The words: good, great, excellent, fantastic, and nice all usually express positive sentiment, but some express a stronger positive opinion than others. Manually scoring these terms is a subjective process.

2. Sentiment can be mixed: Life is filled with bittersweet experiences. People hold positive and negative opinions about things. There are many terms and phrases that express positive and negative opinions at the same time. A diva is a very talented opera singer who's appeal draws in a larger crowd, but divas are renowned for being hard to work with.

3. Sentiment has a source and a target. People and groups of people have opinions about things.

4. Sentimental terms tend to be near their targets. Saying that "the car is good" or that "it is a good car" is far more common and sounds better than other formulations with more words between good and car.

5. In English, opposing sentiments that are used in the same sentence are usually separated by indicator words such as "but" that define their scope.

6. Sentiment can change over time.

7. Sentimental expressions can be very creative.

8. Low-frequency word occurrences can be very indicative of sentiment.

9. Sentimental associations for words are context-dependent to varying degrees.

10. Sentimental term strength exhibits some degree of "guilt by association."

11. Sentimental expression is language dependent.

12. Sentimental expression is culturally dependent.

13. Sentimental words are not limited to any specific part of speech, but adjectives are generally more sentimental.

14. Tense affects sentiment. Since sentiment can change over time, opinions written in the past tense are not the same as opinions written in the present tense.

15. Plain TFIDF does a poor job of identifying sentimental language.

16. Sentiment-bearing sections of documents tend to occur near each other [36].

17. Sentimental expression frequently involves comparing and contrasting things. For example, "Hillary Clinton was a better candidate than Barack Obama."

3.2 Challenges

Numerous opportunities exist for sentimental text analytics, but there are many challenges to automatic sentiment detection in online media. The first challenge posed is by the data source, or genre, itself. News articles, blogs, forums, user reviews are not all equal and should not be treated as if they were. With the exception of news articles, the vast majority of this content is informal, poorly structured, and rife with spelling and grammatical errors. Blogs usually discuss multiple topics, frequently in the same post, from a limitless set of domains. Additionally, blogs have a rich link structure to exploit. Likewise, forums stray off topic easily, and have a unique thread structure. User reviews, such as movie and book reviews, on the other hand don't have any interesting link structure to utilize, tend to stay on topic, but have a lot of extraneous noise from plot summaries.

Not only does the style and structure vary between genres, the content does as well. The topics covered in each of these different genres of documents frequently span different domains. News articles do not discuss the same things that movie reviews do. Even though movie and product reviews both come from review data sources they are still substantially different. The line, "Jolting, heart pounding, you're in for one hell of a bumpy ride!" means something entirely different when

it is said about a car than when it is said about a movie. Building a system that can take advantage of available domain knowledge while still performing well in the general case is the central problem this thesis addresses.

However, there are other worthy challenges for sentiment analysis researchers to solve. These challenges include adjusting for sentiment drift, sarcasm detection, dealing with negation, identifying the target of sentiment, doing coreference resolution, and determining what level of granularity to do sentiment analysis at.

People's feelings change. Aggregating opinions without detecting that a substantial shift in sentiment has occurred is a mistake. The sentiment attached to words drifts and changes with time. Consider how sentiment attached to the word "surge" has changed before, during, and after "the surge" in Iraq. Detecting and accounting for sentiment drift and changing opinions are important challenges for sentiment researchers.

Sarcasm also poses a major problem for most sentiment detection systems. Identifying that an expressed sentiment is too strongly oriented, and out of character with the rest of the document, is a difficult challenge for machines and humans alike.

Dealing with negation is another challenge for automatic sentiment analysis. Many of the techniques used (such as SVMs) do not take account of the order words occur in. Simply placing a single "not" in front of a sentence can change the entire meaning. Tagging negated words with a "not" and using bigram analysis have both been tried as potential solutions. Each has their own problems, but negation tagging is less effective than bigrams [39]. Bigrams do not capture negation very well, since they do not cover the entire span of negated things. Additionally, bigram analysis

is very domain specific and requires labeled training data.

People express sentiment about things. Determining the target of an expressed sentiment is especially important in blogs and forums since they can cover many different topics at once. When authors compare and contrast two things, determining the sentiment's target is crucial.

Since we need to determine whether the sentiment is on target, co-reference resolution is an issue. If the two objects with the same name aren't really the same object, but co-reference resolution was not used, then this information is missing, so the two different objects will be treat as the same object, resulting in errors. If a single thing has multiple names, co-reference resolution should be used. If it is not, then the signal for the target will be randomly cut into multiple pieces. With less information for each target it is more difficult to smooth out noise resulting in lower accuracy. It is also possible to produce contradictory sentimental evaluations for objects with multiple different references.

Sentiment analysis can be approached at varying levels of granularity. Common approaches start at the sentence and document level. Each approach has its advantages. Sentiment analysis at the document level has a greater amount of data to work with for each decision, and does a better job with context than sentence-level analysis. At the document level, sarcasm should be noticeably different than the rest of the explicitly expressed sentiment. If the document is entirely sarcastic than it will stick out at the collection-of-documents level. At the sentence level there is nothing to compare the sarcastic sentence with, but it could be identified as too far oriented toward an extreme. Sentence level analysis can more accurately

determine what the target of the sentiment is. The real challenge here is how to combine the two approaches to get the best of both worlds. For example, sentences expressing similar sentiments tend to clump together in documents.

3.3 Sentiment Analysis

Early attempts to overcome these challenges avoided the problem of domain dependence by working in only one domain. Using statistical models instead of complex grammars and parsers reduced the problems caused by noisy data to a more manageable level. Most researchers using statistical methods chose to work at the document level, where the greater volume of text would give a more reliable result. This choice was also made under the hope that sarcasm would prove less detrimental. As a simplifying assumption, most researchers assumed that the sentiment in a document pertained to the document's main topic. Most researchers operating under these assumptions adopted the vector space model to work in.

3.3.1 Vector Space Analysis

The vector space model comes out of traditional Information Retrieval (IR). In a vector space model, data points are represented as a vector with a set of predefined dimensions. Data points in a vector space model are normally considered similar if they have a high cosine similarity score, which measures the cosine of the angle between the two vectors. There are many different ways to define the dimensions in a vector space model.

The most common kind of vector space is the Bag-of-Words (BoW) feature set. This feature set is a vector space in which each dimension in the vector holds the number of occurrences for that specific word. Bigram word pairs are commonly used in place of or in addition to the unigram word counts. A less popular alternative is to use character-level n-grams instead of words.

Support Vector Machines using BoW feature sets provide a strong baseline accuracy of 82.7% for movie reviews [39]. SVMs are an appropriate tool because they are resistant to blog noise, can handle large BoW feature sets, and are traditionally good at similar tasks like topic based classification [16]. Techniques such as augmenting the training sets with human-supplied annotator rationales [51], and using appraisal groups [49] have improved results to around 90% accuracy. These results are good enough for many applications, but higher accuracy is always a plus.

Zaidan et al. [51] used rational annotation to augment the training set with near duplicate documents by copying the raw document and removing from it the best features (as supplied by human annotators). These new support vectors should cut down the margin size when compared to models that did not train on them. Therefore, they added an extra constraint requiring an SVM to place the original documents further from the margin than their near duplicate documents by some constant. This improved classification accuracy.

Appraisal groups are phrase-level text snippets centered on adjectives containing markup that indicating the type and strength of the appraisal. Whitelaw et al. [49] used a semantic taxonomy of manually verified appraisal groups automatically generated from a seed set using WordNet and other similar resources in conjunction

with a standard BoW to achieve 90.1% accuracy on movie reviews. The total count of terms that matched a given appraisal group were added to the BoW as extra dimensions to form a new extended vector space. These new dimensions are necessarily a linear combination of the original input dimensions; consequently, they are derivable by an SVM with a linear kernel. However, they showed that these additions improve classification accuracy over the standard BoW, indicating that in the original BoW, these new dimensions were not valued as highly as they should be. Simply put, these appraisal groups provided useful semantic information about the relationship between the words in the BoW for the SVM to use. Clearly, there are some shortcomings with the vector space representation.

3.3.2 Vector Space Analysis Shortfalls

Casting documents into a vector space involves a loss of information. The vector space representation drops information about the semantics of each dimension and about the relationship between the things the different dimensions represent. The vector space representation drops information about word and sentence order.

Part of the problem with the vector space approach is that there is no way to specify to the learner that any two features are related to or similar to each other. Note how the following text segments: “The gift”, “The present”, “The toy”, “The toy car”, and “The brand new toy” all discuss the same things but, when they are represented in a BoW they appear quite different. Once they are transformed into a sequence of unique identifiers with their associated counts they appear to

be just as close to each other as these largely unrelated strings: “The bath ”, “The computer”, “The food”, “The food fell”, and “The poor need food”. Using character level representations fixes some of these problems while introducing others. Strings like “I like eating” and “meals are good” do not appear similar while strings like “I like eating” and “I am boring” do appear similar.

Some researchers have attempted to rectify this shortfall using matrix decompositions. One such example, Latent Semantic Analysis (LSA) involves representing all the documents as a term by document matrix. The count of the terms in these documents are divided by their entropy, to discount uninformative terms. “Automatic information retrieval methods are greatly improved by transformations of this general form.” [11] After the documents have been transformed, the Singular Value Decomposition (SVD) of the matrix is calculated and used to put the documents into a new space.

$$\mathbf{A}_{nxp} = \mathbf{U}_{nxn}\,\mathbf{S}_{nxp}\,\mathbf{V}^{\mathbf{T}}{}_{pxp}$$

New points can be represented in the space by multiplying the transformation matrix by the new point. The transformation matrix is computed by multiplying the inverse sigma matrix by the transpose of the left hand singular matrix U. This space can be further reduced by selecting the top k singular values. Since the sigma matrix is a diagonal matrix with the top singular values occurring in order along the diagonal it is easy to remove excess dimensions and their corresponding rows in the U transpose matrix. While these techniques have improved classification accuracy,

the dimensions they produce do not correlate to concepts that humans can easily understand.

An alternative approach attempts to improve classifier generalization with vector space using linguistic concepts like: synonym, antonymy, similar functions, and subclass relationships. Vector space representations fail to capture these relationships. BoW feature sets assign a pencil and a pen two different unique identifiers and ignore that they both have the same basic function. This poor grasp of semantics negatively impacts generalized classification accuracy. Machine readable dictionaries like WordNet describe these relationships. However, to use such a resource for classification the sense of the word must be identified automatically. This is still an open problem with future work branching into several areas: the sense granularity problem, the knowledge acquisition bottleneck problem, and domain specific word sense disambiguation. [30] The granularity problem for sense inventories is a well-known problem, especially for WordNet where the senses are too fine grained for most NLP tasks [27] [30]. It is possible to use SVD to generate term vectors of a large corpus to replace WordNet Domains. [45]

Vector spaces representations also ignore sentence and word order. However, both sentence and word order are very valuable information for sentiment analysis.

Pang and Lee took advantage of the observation that sentimental sentences tend to clump together in their minimum cuts paper. [36] They trimmed out objective content from movie reviews and used an SVM BoW classifier to determine review polarity. To determine subjective sentences, they cast the task as a graph problem, and used the minimum cut between the subjective node and the objective

node to form a classifier. First, they created an SVM subjectivity classifier and trained it with objective and subjective sentences from a different set of movie reviews. Then they broke reviews into sentences and inserted them into the graph as nodes. They also inserted a positive node and a negative node. Next, they weighted the edges between sentence nodes and the positive and negative pole nodes using the distance of those sentences from the margin of their subjectivity classifier. Working under the observation that sentimental text segments tend to occur next to each other in a review, they assigned scores to edges between sentences by their proximity within the review. Second, they used the minimum cut on this graph to remove the objective content from their reviews. Third, they trained and tested another SVM BoW classifier on their trimmed reviews. They found that using these trimmed reviews increased classification accuracy.

Word order is also very valuable. Calculating sentiment scores for words based upon their pointwise mutual information to known positive and negative sentiment words can provide their semantic orientation. [47] This approach takes advantage of the observation that the sentiment expressions tend to be close to their targets. This information is lost when the document is converted to a vector space.

The most important problem with using the vector space representation for sentiment classification is the assumption that the training and test sets are fundamentally similar to each other. In practice this assumption is usually violated because there is not enough similar labeled data for the test genre and domain. The accuracy of machine learning techniques drops with the degree to which this assumption is violated.

3.4 Domain Adaptation

The growth of commercial interest in sentiment analysis coupled with the limitless array of possible domains emphasizes the need for a solution that can span multiple domains with varying levels of similarity to our available labeled data. Domain adaptation, also known as transductive transfer learning, is the machine learning task of building a model for a target domain given labeled data for the problem in a different domain with the possibility of some amount of unlabeled data in the target domain.

Cross domain training was an early attempt to accomplish this. Cross domain training is a common technique used to compensate for a lack of existing labeled training data. This technique involves training a classifier for the target domain on labeled data from another domain. Results are generally much lower than training on a data set from a similar sized labeled data set from the target domain. Experiments with SVMs show that similar domains show a decrease in accuracy of about ten percent, but dissimilar domains suffer even greater penalties dropping accuracy to just above random chance. Using a mixture of multiple domains seems to provide mixed results based upon how well each individual domain does for cross domain training [1].

There are two primary lines of research in the domain adaptation community: transferring instances, and exploiting relationships between features. I do not believe either approach to be the right high-level strategy.

Transferring instances can be accomplished using an EM framework [7]. First,

label the target domain documents using Naive Bayes. Then for each iteration recalculate the probability values for terms based upon the target document labels in the prior iteration. Repeat for a fixed number of times. This is more accurate than either Naive Bayes or an SVM used in a cross domain training approach.

Some researchers view domain adaptation as a sample selection bias problem. Consider the source domain as a sample of the data in the problem space, and the target domain as a different data sample. SVMs with soft margins have been shown to degrade under a sample selection bias [50]. Eliminating this sample selection bias by resampling using knowledge of the selection bias, or by weighting instances to create a better SVM model is a viable domain adaptation approach.

It is very tempting to transfer instances because they are self contained labeled items. However, these kinds of instance weighting approaches do not account for utterances like "Read the Book" which is very positive in book reviews and very negative in movie reviews. It is quite possible that the probability of the class label for the same set of features will change between domains for sentiment classification tasks. When transferring instances it is imperative that the probability of the class label not change much between the two domains given the features [15].

The other major line of research is in exploiting the relationships between features. Structural correspondence learning (SCL) attempts to find and use features in the target domain that function like features in the source domain where labeled training data exists. SCL works by finding features that co-occur in the target domain with a set of known discriminative pivots (found using mutual information in the source domain) that exist in both the source and target domain. These

pivots are assumed to have the same bias in both the source and target domains. Features that occur with the pivots in the source domain are assumed to correspond to features in the target domain that also co-occur with the pivots. This information helps to create a mapping between the source and the target domains that allows for classification on the target dataset. SCL is highly dependent on the selection of good pivot terms, with the most successful criteria for sentiment analysis being the mutual information of the term with the class label [4].

Approaches that work on the relationships between features, like SCL and Spectral Feature Alignment (SFA) [34] frequently use matrix decompositions like the eigenvector decomposition and the singular value decomposition. SFA first identifies domain-independent features, then it establishes a bipartite graph between domain-independent features and all other features, this graph is then used for spectral clustering. The K largest eigenvectors are then selected and used as clusters. From this SFA determines a mapping between the domain-independent and the domain-specific features. This gives away their sentimental orientation since the orientation of the domain-independent features can be determined by their bias in the source domain.

These feature-relationship based approaches are intriguing, but my results indicate that they are not the right direction for document-level classification approaches. Features that correlate well to known discriminative domain-independent features are probably also discriminative, however at the document-level they are also quite redundant. Neither approach ever evaluated how well using just domain-independent features works. SFA's slightly better results than SCL could well be

explained by their better selection of domain-independent features. It seems likely that further advances in discovering and weighting domain-independent features is a better research avenue to pursue. Results presented later in this thesis show that using only well-weighted domain-independent features exceeds either approach's accuracy, and that the gains show using correlated features were almost negligible.

A recent transfer learning survey asked three questions: "What to Transfer?", "When to Transfer?", and "How to Transfer?" [35] These are good questions to ask for the general case of transfer learning, where the classification problem is allowed to change, or where the source labels may not be present. However, researchers working on domain adaptation have source labels and are working on the same classification problem.

Instead the following questions become more important:

1. What is the bias in the source domain(s) of the item to transfer?

2. From this bias what can be concluded about its bias in the target domain?

By posing the problem in this way domain adaptation researchers can profitably interact with machine learning researchers, since the field of machine learning provides numerous techniques to answer the first question. The second question is the defining problem for transfer learning. A simple baseline approach would assume that the item's bias will not change between the source and target domains. This assumption invariably leads to lackluster accuracy as the cross domain training approach described above illustrates.

Given that feature bias will change between domains and that the relative

frequency of the feature between the source and target domains is observable what can be concluded about the feature bias in the target domain? If the feature was biased in the source domain will it still be as biased in the target domain if it becomes less/more common? A simple examination of feature bias versus feature frequency on any sentiment analysis domain is sufficient to cast doubt upon this approach. TF and IDF are not very good predictors for feature bias strength. [26] [25] This point will be covered in greater detail when Delta IDF is described. Determining the bias of a feature in the target domain using the bias in the source domain is hard; it is much easier, and still quite useful, to determine if the bias for the feature is likely to change between the two domains.

Chapter 4

Delta IDF: Identifying In-domain Problem Specific Information

My theoretical foundation introduces three types of information in written documents: genre, domain, and problem specific information. This chapter describes Delta IDF, an efficient technique to weigh word-based features in binary classification tasks to identify problem-specific information. This technique is broadly applicable, as it significantly improves classification accuracy on a range of text problems including subjectivity detection, sentiment polarity detection, and spam detection. Most common text classification approaches use a document's n-grams (words and short phrases) as its features. They assign each feature a value equal to that feature's frequency or TFIDF score relative to the training corpus. My approach assigns each feature a value computed with the difference of that feature's inverse document frequency in the positive and negative training sets. While this technique is remarkably easy to implement, it gives a sizable, statistically significant improvement over the standard bag-of-words approaches using support vector machines on a range of classification tasks.

4.1 Introduction

Most textual vector space representations value every feature occurrence equally even though not all features are equal. IDF weighting approaches attempt to cor-

Sentiment Feature	X^{th} **Lowest IDF Score**	**Subjectivity Feature**	X^{th} **Lowest IDF Score**
like	41	you	62
good	83	love	65
best	149	like	67
bad	171	good	99
great	198	very	129
better	203	i	137
love	221	us	179
funny	315	we	185
interesting	331	funny	197
the best	343	your	210
more than	348	bad	216
a good	357	better	246
original	390	fun	324
fun	394	beautiful	328
pretty	424	the best	369
unfortunately	506	our	374
a great	512	entertaining	392
humor	624	want	394
entertaining	710	humor	404
obvious	728	interesting	410
perfect	730	evil	416
beautiful	839	feels	417
boring	862	feel	428
obviously	926	emotional	446
worse	928	you're	495
interest	930	sense of	498
enjoy	956	seem	522
stupid	992	pretty	543
		compelling	569
		entertainment	607
		seems to	675
		dramatic	697
		powerful	699
		a great	783
		hilarious	879
		gay	899
		funny ,	919
		engaging	922
		enjoy	924
		great	1000

Table 4.1: Problem to Correct: Surprising examples of low scoring IDF features from over 300,000 features in two movie review datasets.

rect this, however IDF is a poor choice for datasets that are focused upon the classification problem. Examining the movie review dataset reveals that obviously sentimental words, like those shown in Table 4.1, rank in the bottom 0.3% of 300,000 features. Words that express clear value judgments are some of the lowest-scoring IDF features in the movie review sentiment polarity detection data set because IDF is a measure of rarity and the document collection was selected based on the fact that the documents express sentiment about movies. Similarly, personal pronouns signal an opinion, but have very low IDF scores in the sentence level movie review subjectivity detection data set. I show how Delta IDF [25] fixes these problems and works on a broad range of classification tasks.

Many researchers have chosen to focus less upon textual feature weighting and more upon exploiting the particulars of the genre and domain. For example, determining how a congressional representative will vote on a bill given their congressional floor speeches can be accomplished by exploiting the argument structure found in speaker reference links [46]. In that dataset the speakers' votes were used to determine ground truth class labels. Manual annotations provide links between the various speakers.

Another example, focuses on the peculiarities of the review genre noting that movie reviews often start with predominantly objective plot summary before expressing opinions [36]. Pang and Lee trimmed out such objective content from movie reviews and used an SVM bag-of-words classifier to determine the sentiment polarity of the remainder of the review. In determining objective sentences, they cast the task as a graph problem and used the minimum cut between the subjective

node and the objective node to form a classifier. To do this they constructed a graph of review sentences cast as nodes and inserted two nodes, one representing a positive pole and one representing a negative pole. The distance between sentence nodes and the poles was calculated using their distance from the margin of an additional SVM subjectivity classifier trained on a different set of movie reviews. Then they assigned scores to edges between sentences by their proximity within the review. Next, they found the minimum cut between the positive and negative poles, and discarded the sentences on the objective side. Finally, they trained and tested another SVM bag-of-words classifier on their trimmed reviews.

A third example of researchers focusing on the unique properties of their genre occurs in the product review genre. In the product review genre it is important to identify features about the product, like a camera's picture quality or size, and to determine what the reviewer feels about those product features. Data has been labeled for this task [9]. This data allowed me to test how well my technique picks out features that humans would choose. In addition to this I created document level labels for this dataset's reviews by summing the sentiment about a product's features.

4.2 Approach

In a bag-of-words feature space, each term (i.e., word or phrase) is assigned a numeric value. Choosing the best way to compute this value can be crucial to obtaining good performance. A term's value is often just its frequency in the docu-

Sentiment Classification Movie Review Data

IDF Baseline	**Positive**	**Negative**
1700s	is excellent	this mess
seems muted	. cameron	this turkey
viewer's imagination	harris)	terrible .
metal at	most powerful	worst movie
only semi-serious	, great	a stupid
and astrophysicist	very effective	dull ,
paid big	lovingly	is terrible
fiction spectaculars	characters with	lame ,
, 133	fargo ,	falls flat
latifah as	melancholy	bland and
real alien	. spielberg	anyway ?
compelling performances	is terrific	degenerates
crichton science	gattaca	(scott
; has	ideals	not funny
norman's less	one which	4 on

Subjectivity Classification Movie Review Data

IDF Baseline	**Subjective**	**Objective**
apartments and	but it's	discovers
unfolding hidden	. it's	decide
personal belongings	. but	he finds
grifter	me	his father
hidden secrets	movie's	where he
and rummages	laughs	falls in
thus unfolding	the movie's	year old
breaks into	the screen	his mother
rummages their	it doesn't	) who
their apartments	. the	boyfriend
rummages	if you're	help of
grifter breaks	flick	the help
a grifter	it does	government
the all	entertaining	her to
ultimately provides	. this	discovers that

Table 4.2: Top 15 highest-scoring features for movie review datasets for both sentiment detection and sentiment polarity classification. Top IDF scoring words versus the top class-specific features found using the Delta IDF technique.

ment. Sometimes these values are further weighted by metrics measuring how rare the terms are in the documents in the corpus. My approach treats the positive and negative training points as two different corpora. Term counts are weighted by how biased the terms are to one corpus using the difference of that term's IDF scores in the two corpora. Simple add-one smoothing for document frequencies prevents potential errors caused by dividing by zero when a feature exists in only one corpus.

Properly smoothing low frequency terms is important for Delta IDF's feature level accuracy because there are a large number of low document frequency terms. Simple add one smoothing is sufficient to improve Delta IDF's accuracy when compared to non-smoothed versions. Simple add one smoothing assumes that the next document to occur will have the feature in question. This provides a great boost in feature accuracy because it improves the IDF estimates for the common situation when strong discriminative features occur in only one of the class labels.

However, add-one smoothing is inappropriate when the number of positive and negative documents are no longer roughly equal. In a setting where the corpus has been broken into two halves, each of which undergoes add one smoothing, it is implicitly assumed that the next document to occur is equally likely to be positive or negative. When the known labels distribution violates this assumption it is more appropriate to add a probabilistically weighted count to each of the two sub-corpora. Since the training dataset is balanced, simple add one smoothing is sufficient.

Delta IDF values are calculated as follows, given that:

- $|P_t|$ is the number of positively labeled training documents with term t.

- $|P|$ is the number of positively labeled training documents.
- $|N_t|$ is the number of negatively labeled training documents with term t.
- $|N|$ is the number of negatively labeled training documents.
- V_t is the feature value for term t.

The term weight can be simplified under the assumption that the training set is balanced, i.e. has the same number of positive and negative examples.

$$\begin{aligned} V_t &= \log_2\left(\frac{|N|+1}{|N_t|+1}\right) - \log_2\left(\frac{|P|+1}{|P_t|+1}\right) \\ &= \log_2\left(\frac{(|N|+1)\,(|P_t|+1)}{(|N_t|+1)\,(|P|+1)}\right) \\ &= \log_2\left(\frac{|P_t|+1}{|N_t|+1}\right) \end{aligned}$$

This equation assigns evenly divided features zero weight, but prefers words that are increasingly unevenly distributed between the positive and negative classes using inverse document frequency (IDF) values. High IDF scoring features in a given class are rarer in that class, their presence in a document indicates that the document does not belong to that class. In the balanced case the equation simplifies into the log of the ratio of positive and negative occurrences. The logarithm helps to discount extremely biased cases in addition to determining the correct sign to denote the bias of the feature.

Features that are more common in the negative training set than the positive one receive negative scores, perfectly balanced features receive a zero, and predominantly positive features receive positive scores. Regular IDF lacks this capability, but these feature biases enable visualization of the top positive and negative features

Congressional Debates Transcripts

IDF Baseline	Support Bill	Oppose Bill
one program	look forward	no child
their optimum	competition	less likely
between mentoring	. hastings	is supposed
developed .	order against	proponents
after school	representation act	struggling
15 hours	in representation	african
preschoolers from	july	proponents of
start graduates	property rights	votes for
nor would	not oppose	to recruit
optimum	elections to	recruit
offer mentoring	divided and	not discriminate
even become	to working	to amend
themselves some	general debate	rights protections
family services	him to	separation
a qualified	sponsor of	separation of

Enron Email Spam Classification

IDF Baseline	Spam	Not Spam
milind	meds	enron
name)	viagra	hpl
use vacation	paliourg	daren
x 39247	pain	=- forwarded
mountains	php	forwarded by
of bummed	. php	/ ect
like july	cialis	hou /
have transcended	drugs	/ hou
the keyboard	in compliance	/ enron
patil	spam	@ enron
as dave	biz	@ ect
they miss	xp	: subject
plateau	sex	ect @
39247	. biz	meter
transcended	dealer	ect cc

Sentiment Analysis on Products

IDF Baseline	Positive	Negative
company does	is easy	Symantec
fried	very easy	busy
they asked	solid	you pay
repaing	a camera	n't play
stopped supporting	camera for	neither
they stopped	is really	refund
os .	I like	to contact
happy man	great camera	mistake
recharged ,	very pleased	to avoid
sold for	her	of junk
mac .	i like	not buy
the damned	are easy	a refund
within months	beautiful	freezing
soundblaster	megapixel	of Norton
damned	6610	Security 2004

Table 4.3: Examining the 15 highest scoring features for three different binary classification datasets shows that the Delta IDF technique is better at identifying useful features when compared to the IDF baseline.

to verify Delta IDF's effectiveness for a domain. As Tables 4.2 and 4.3 show, the best IDF scoring words for each domain are much less useful than the class specific features determined by Delta IDF.

Delta IDF very accurately identifies positive and negative sentiment words in movie reviews. Not only are the top scoring positive and negative features clearly more sentimental than the features valued by IDF, they are also correctly oriented. Most top features are either obvious complements, insults, sentimentally expressive words, or sentimental phrases. Mentions of very popular films, such as seven-time academy award winner *Fargo*, correlate with positive sentiment, while mentions of unpopular films are, not surprisingly, just rare. The rest of the top 1000 positive and negative features using Delta IDF are just as intuitive and powerful.

Delta IDF is also very effective for subjectivity detection. Many objective features identified are story related because the reviewer must summarize the plot, this involves talking about how the main character discovers something about his past, or falls in love with some other character, or where the main character receives the help of other characters and defeats a villain. Subjective features such as *"entertaining"* and *"laughs"* express a clear value judgment. Other top subjective features indicate a change of expectation such as *"but"*, or prime the reader for a value judgment with references to the author, the reader, and generic mentions like *"the movie's"*.

Expressing political support is complex. While obvious features like *"looking forward"* and *"not oppose"* exist, many features are more complicated. For example, *"as amended"* indicates support because it shows that both sides have had a chance

to compromise, or at least buy votes with pork, and come to an agreement. In politics mentioning inflammatory issues pertaining to race, religion, discrimination, sex, and party affiliation is a quick way to close down real debate and compromise. Discussing party affiliation is a sure sign that partisan politics are in play. Even mentions of partisanship and bipartisanship are toxic, features such as *"party-line vote"*, and *"bipartisan spirit"* were predominantly used by opponents of the bill. If you have to talk about a bipartisan spirit you certainly don't have it. These types of features show up in the top 1000 out of over 300000 features ranked by Delta IDF. Furthermore, these features are ranked much higher by Delta IDF than by raw IDF.

Spam classification features as shown in Table 4.3 are easy to understand. Spammers advertise medications and products. The top 1000 spam features include a long list of investment related terms, pain relief related terms, and terms relating to deals or free stuff. Top not-spam features in the Enron email dataset [20] include terms related to Enron and HPL, which was acquired by Enron. Real business communications frequently involve forwarding messages, communicating with shared coworkers, and attaching documents, many of which are spreadsheets. The 27^{th} highest not-spam feature is *".xls"*. Popular first names also feature prominently in emails that are not spam.

In the Liu product review data set, Delta IDF correctly determines the strength and bias of features that assess Liu's manually annotated product specific attributes. Liu's annotations asserted that small size is a positive attribute for cameras [9]. Delta IDF deemed camera based features like *"so small"*, *"is small"*, *"very small"*,

"small", *"small ,"*, *"small and"*, and *"a small"* to be among the top 1000 most positive terms in their data set. As expected, none of the top 1000 negative features scored by Delta IDF for cameras contained the word small. Top positive features for the camera picture attribute include *"quality pictures"* and *"great pictures"* while the top negative features include *"picture after"* and *"no picture"*. Furthermore, Delta IDF's top 1000 strongest features include more terms like *"very easy"*, *"mistake"*, *"to avoid"*, and *"a refund"* than the top 1000 strongest regular IDF terms.

However, many of the top scoring Delta IDF features are dominated by product sentiment imbalance in the training set. People love their digital cameras, but hate their anti-virus software: camera reviews are positive by at least a nine to one ratio while reviews for anti-virus software are three to one negative. Although the documents in this dataset all come from the product review genre they can come from different domains. The camera domain is not like the anti-virus software domain; features that are discriminative in the camera domain may have the same sentimental orientation, a different orientation, or they may be absent in the ant-virus software domain. To examine these important differences I introduce another product review dataset with many more product categories. Blitzer's domain adaptation dataset is an additional product review dataset with 25 different types of products as categorized by Amazon [4]. Figure 4.1 is a visualization of the 50 most positive and negative features according to Delta IDF for the apparel product review category. The relative size of terms in that word cloud are proportional to their relative Delta IDF strength score. Terms in blue indicate positive sentiment while terms in red indicate negative sentiment. In every product category except

Figure 4.1: Top Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red.

the tools domain the terms appear to be both strongly sentimental and correctly oriented. The tools domain is quite aberrant due to the extremely small number of labeled data points causing Delta IDF to preform poorly. For a complete set of all 25 different in-domain model visualizations please see Appendix B.

4.3 Evaluation

4.3.1 Delta IDF weights for SVMs

Delta IDF weights can be used to increase or decrease the relative weights of terms in documents before training or testing an SVM. To do this, multiply the number of occurrences of each term by its Delta IDF score. This mirrors how TFIDF multiples the TF component by the IDF component. The movie review used in Table 4.4 shows that the document's top scoring features are clearly more sentimental than

Delta TFIDF	TFIDF	Term Frequency
, city	angels	,
cage is	angels is	the
mediocrity	, city	.
criticized	of angels	to
exhilarating	maggie ,	of
well worth	city of	a
out well	maggie	and
should know	angel who	is
really enjoyed	movie goers	that
maggie ,	cage is	it
it's nice	seth ,	who
is beautifully	goers	in
wonderfully	angels ,	more
of angels	us with	you
underneath the	city	but

Table 4.4: The three feature-value metrics (Delta TFIDF, TFIDF and raw frequency) emphasize different features. Compare the 15 highest ranked features for a positive review of the film *City of Angels*. Delta TFIDF has promoted features that evidence positive sentiment.

either TFIDF or plain term frequencies. TFIDF's top scoring features appear to be the topics of the review, while the top raw term frequency terms are dominated by stop words. Delta IDF places a much greater weight on sentimental words than either of the alternatives.

This evaluation uses several datasets including Pang and Lee's movie review, subjectivity, and congressional debates transcripts data-sets, along with the Enron email spam corpus, and Liu's product review dataset. The variety of datasets, labeled for multiple different classification tasks, and with data points ranging from sentences to full documents shows that Delta IDF is robust and versatile. In the following experiments Delta IDF is compared against a baseline bag of unigram and bigram words using 10-fold cross validation and paired two tailed t-tests to prove statistical significance.

To ensure experimental uniformity and validity the baseline and Delta IDF

experiments used the same code base. They had a common feature space with both single words, and bigrams (i.e., ordered word pairs). Words that occurred in only one document were removed, but stop words were retained. All tests used svm_perf with a linear kernel [18]. The linear kernel was chosen because it is fast, to allow direct comparison with other researchers who use the linear kernel, because linear kernels yield higher accuracy for most variations on the bag-of-words feature sets [21], and because I deem our problem to be linearly separable. Words were not stemmed or lemmatized because [21] shows that these expensive steps are detrimental to accuracy.

Movie Review Data	**10-fold Acc**	**Variance**
SVM Delta IDF	**88.1%**	17.88
SVM Term Count Baseline	84.65%	3.94
SVM TFIDF baseline	82.85	9.17
Mincuts + subj. detection	87.2%	Unknown
Subjectivity		
SVM Delta IDF	**91.26%**	.47
SVM Term Count Baseline	89.4%	.74
Congressional Debates		
SVM Delta IDF	**72.47%**	13.84
SVM Term Count Baseline	66.84%	7.36
Spam Detection		
SVM Delta IDF	**98.917%**	2.5×10^{-5}
SVM Term Count Baseline	96.617%	6.8×10^{-5}
Product Reviews		
SVM Delta IDF	81.41%	.00306
SVM Term Count Baseline	79.242%	.00205

Table 4.5: Accuracy of Delta IDF versus term frequency for binary classification on five datasets, showing the 10 fold cross validation accuracy and variance. Boldface results are significant to the 98% confidence level.

Table 4.5 shows that Delta IDF outperforms the raw bag-of-words baseline for each dataset. Movie review sentiment classification results are higher than the

dataset's creators using their more complex and computationally more expensive minimum cuts approach. They use an additional trained sentence level SVM subjectivity classifier, which requires an additional set of subjectivity labeled sentences. The subjectivity entry in Table 4.5 shows that when Delta IDF is used on their subjectivity dataset it outperforms the type of subjectivity classifier they used in [36] with a P value of .000106. Therefore, using Delta IDF will even further improve their movie review results. These two datasets prove that Delta IDF works on both subjectivity detection and documents of varying sizes.

SVM classification with Delta IDF produces significant improvements over the baseline for other kinds of binary classification problems as well, including spam detection and predicting a congress member's vote on a bill given their comments about it. The baseline congressional classifier matches the method described in [46] for SVM speech classification and produces equivalent results, but Delta IDF improves on the baseline with a P value of .000582. On the Enron email spam classifier Delta IDF out-performs the baseline with a P value of 1.7×10^{-5}.

Delta IDF can also be used on other kinds of sentimental datasets such as product reviews. In Liu's data for nine products [9] the reviews were not annotated for overall product opinion, so I labeled them for sentiment using the sum of product feature scores. While this method does not account for the importance of different product features to the reviewer, I believe it is a reasonable approximation for human labels. This could be one reason why the results in Table 4.5 fall just short of a 95% confidence interval. With a P value of .0509 I am still reasonably confident that these results represent a modest gain.

The imbalance of class labels for each product is a contributing factor to the weaker results on Liu's data. The dataset consists of 399 positive documents and 198 negative documents, but the actual imbalance for any given product type is much worse. Canon cameras had 75 positive but only eight negative reviews – not even enough negative Canon reviews to put one in each fold! Reviewers are also overwhelmingly positive about routers. Conversely, Norton AntiVirus has 32 negative reviews to nine positive ones. This imbalance distorts the distribution of product specific features: just mentioning anything about cameras is a good sign that the review will be positive because most people like their digital cameras. This observation confirms the general theory of three types of signals, and is one of the main drivers for the next chapter.

4.3.2 Dot Product Classification

To improve label quality, to account for the variety of products, and to control for class imbalance in the training set, my next set of tests used Blitzer's 25 category Amazon product review data set. The evaluation data set contains reviews for 25 diverse types of products ranging form clothing, to food, to outdoor sports, all the way to electronics. The document level labels for these reviews are much cleaner than Liu's dataset because they come directly from the reviewer's own star rating for the review. Reviews were labeled as positive or negative using a five star rating provided by the reviewer explicitly in the review form. Four and five star reviews were deemed positive documents and one and two star documents were deemed

negative documents. The data set was balanced for each type of product to ensure that the number of positive and negative reviews were roughly equal. These reviews were tokenized into unigram and bigram word counts and released for public use in conjunction with work on SCL-MI [4]. I used this dataset and the tokenization provided to enable direct comparisons with earlier work without anomalies created by different tokenization.

Tables 4.2 and 4.3 show evidence for the discriminative power of Delta IDF's feature weighting. Using these weighted features to represent data points provides a statistically significant improvement to state-of-the-art machine classification approaches that use other term weighting functions. In fact, the bias of these features is so good that they can be used to directly classify documents with no further learning. This is very beneficial because iterative parameter tuning and training on the development set is a time consuming task that is usually accomplished by relying upon human intuition.

Instead of using the Delta IDF vector as a prior weight on all documents before SVM classification [25], they can be used to directly perform judgments. Here I describe a new process called the "dot product classification rule". First take the dot product of the weight vector with the document's term frequency vector. If the dot product is positive the point is positive, if the dot product is negative the point is negative, if the dot product is zero make no decision.

To avoid building a biased classifier, which is generally considered less useful than a balanced classifier, and that will not transfer well to a new domain, individually normalize the positive and negative features to unit length. To accomplish this,

treat the positive features as a vector and normalize it to unit length (L^2 norm). Then do the same for negative features. Finally, concatenate them back together to form a single vector. This bias balanced vector will always be square root of two units long. For convenience, normalize this vector to unit length so that any dot product operations with other unit length vectors will produce cosine similarity scores.

The dot product classifier is a linear combination of feature weights. When used with Delta IDF feature weights this becomes a linear combination of non-linear (logarithmic) feature weights. This classifier will assign positive labels when the document vector's cosine similarity to the weight vector is positive and negative when the document vector is dissimilar to the weight vector.

The dot product classification rule works well with Delta IDF weight vectors for English language documents. Most weight vectors produced by Delta IDF, on a reasonable number of documents, will cover enough of the English language to produce a non-zero judgment in almost all cases.

Applying the dot product rule is simple, fast, and intuitively similar to most kernel machines. The popular SVM classification approach compares the dot product of a document with the normal vector of the maximum margin separating hyperplane against a bias feature. The two key differences are centered on the actual model learned from the data, and the presence of a bias term. Using zero for the dot product rule's decision point is similar to an SVM with a bias feature equal to zero. This is equivalent to setting a constraint on your SVM to produce hyperplanes that intersect the origin. The dot product rule classifies points by the total sum of

the bias of their features. Clearly, the dot product rule makes a lot of sense, and is a well grounded classification rule for machine learning approaches using vector spaces.

For the next set of experiments the dot product classifier, using Delta IDF weight vectors as models, was evaluated against published SVM baselines built by Blitzer. There are four different categories of review data Blitzer evaluated his thesis work on: Kitchen, Electronics, DVDs, and Books. He published the accuracy results of a baseline SVM for each of these four categories. Each baseline was built using 10-fold cross validation on the documents for their respective product review category.

For each product category I built a Delta IDF model in a 10-fold cross validation setting, and reported the total accuracy. I compared the accuracy of my Delta IDF classifier over the four categories against the published accuracy of Blitzer's SVMs. Since 10-fold cross validation makes exactly one judgment per document it is possible to pair up the accuracy results on each product category for a paired t-test because the same documents are being measured in both cases. Furthermore, additional confounding factors like document parsing and tokenization have been eliminated in this paired experiment by using the document tokenization provided by Blitzer.

Even though there are only four observation pairs, a paired t-test is still the correct statistical test to use. The test requires that each set of observations must follow a normal distribution and that the experiment has been setup with paired observations where the pairing is designed to eliminate confounding factors. My

paired experiment removes confounding factors such as classification on different documents, and different tokenization and parsing for documents. This leaves any difference in the accuracy of the two classification approaches to be explained by the difference in learned model quality alone.

The distribution of the average accuracies for each product category follows a normal distribution. I shall prove this claim using the central limits theory. The central limits theory states that multiple averages, of different samples drawn from the same underlying distribution, are normally distributed if the number of points in each sample is sufficiently large. Notice that, for every given product category the sample of points is drawn from the same general distribution of product reviews. Furthermore, each product category, expect for Tools is sufficiently large. Sufficiently large is usually defined as greater than 30. The smallest of the four categories in this test has 7760 data points.

For readers that are still not convinced, statistical tests on the distribution of averages for Delta IDF on the 24 sufficiently large product categories for skew and kurtosis found no evidence that the data was not normally distributed. The Lilliefors normality test [23] did not pass at the lowest confidence level published in the table of selected values. Therefore, I must reject the alternative hypothesis that the data is not normally distributed. The D'Agostino skewness test found no evidence to reject the normality assumption with a p-value of .3669. The Anscombe-Glynn kurtosis test found no evidence to reject the normality assumption with a p-value of .193. The Jarque-Bera test that examines both skewness and kurtosis at the same time found no evidence to reject the normality assumption with a p-value of .4198.

In theory, the distribution of average accuracy for each product category should be normal. There is no statistical evidence that these distributions are not normal. Thus, paired t-tests are a valid test to use. They do not require that the two distributions have the same variance. Instead they require that the samples be paired to reduce variance caused by confounding facts, as I have clearly done. Paired t-test are the correct test to use because the pairing process reduces variance for confounding factors giving a better confidence estimate.

Category	**SVM**	**ΔIDF Classifier**
Kitchen	87.7	**89.50**
Electronics	84.4	**88.00**
DVDs	82.4	**86.17**
Books	80.4	**87.45**

Table 4.6: In-domain classification accuracy for bigram BoW SVM vs. dot product rule classification with Delta IDF weight vector. Results are significant to the 95% confidence level.

In-domain Delta IDF with the dot product classification rule is more accurate than term frequency weighted SVM models built by Blitzer as shown in Table 4.6. These differences are statistically significant using paired two tailed t-tests to the 95% confidence level.

The average in-domain dot product classification rule's accuracy for all 25 domains in the dataset is 83.23%. Although most of the 25 domains have an average accuracy in the mid-to-upper 80s the average accuracy over all 25 domains drops due to poor performance on a few domains. These domains are characterized by

having an extremely limited number of labeled data points. The worst offender is the Tools category, which is poorly balanced by class label and has only 19 data points. The in-domain accuracy for the Tools category is 56.67% causing a substantial drop to the 25 category average. Needless to say, it is very hard to build an accurate classifier in a ten fold cross validation setting when each fold has between one and two data points. Delta IDF weights accurately capture the sentimental orientation of terms in product reviews. In the next chapter these results will serve as a baseline to illustrate the advantages of domain-independent classification.

4.4 Discussion and Conclusions

Many bag-of-words approaches weight features using only a function of their occurrence count in the document. TFIDF is a notable exception where the term frequency of a feature is multiplied by its pre-computed IDF score in the corpus. However, IDF weights are a bad choice for classification problem specific datasets because they prefer rare features. Since the documents are all from the same domain, the most descriptive features will be much more frequent than they would have been if the data had been randomly sampled from all recorded documents, thus resulting in some of the best features having the worst IDF scores.

This effect is especially noticeable for sentiment classification tasks. When detecting subjective versus objective speech for film reviews, IDF ranks the word *"funny"* and the bigram *"funny ,"* along with the word *"but"* among the lowest scoring features. Additionally, sentiment words tend to have very low frequency

counts in any given document because authors, often add linguistic variety to their reviews using synonyms, resulting in lower TF scores. In practice many sentiment words are generic and tend to have low TFIDF scores. Delta IDF not only ranks these example words and other similar words as some of the most useful features, but also correctly identifies the polarity of the sentiment or subjective orientation they indicate.

For support vector machines, using Delta IDF to weight features statistically outperforms raw term counts and TFIDF feature weights for documents of all sizes for subjectivity detection, sentiment polarity classification, detecting congressional support for bills, and spam classification. Delta IDF provides the orientation of a term to the class label and ranks these terms by their strength. Delta IDF weights are so accurate that using them with the dot product classification rule negates the need for further training. Delta IDF is the first feature weighting scheme to identify and boost the importance of discriminative terms using the observed uneven distribution of features between the two classes before classification. This transformation should improve performance with character level n-grams, on other domains, on other languages, and with any binary classification technique that uses a bag-of-words.

Chapter 5

DIVA: Domain Independence Verification Algorithm for Sentiment Analysis

In-domain textual sentiment classification using machine learning is very effective. However, when applied to out-of-domain data, the accuracy of these models drops noticeably. To solve this problem, I describe a method to determine the sentimental strength and polarity of words in a given domain, couple it with a new method to verify the domain independence of these bias scores across multiple domains, and use these bias scores with a simple classification rule. This classification rule allows me to classify out-of-domain documents with higher accuracy than both state-of-the-art domain adaptation techniques like SCL-MI and in-domain SVMs. My Domain Independence Verification Approach yields the highest known accuracy for product review categories using Blitzer's 25 category Amazon product review dataset.

5.1 Introduction

Sentiment analysis is the automatic process of determining the author's feelings about a specific topic of interest. Single domain sentiment analysis has been studied extensively. While Support Vector Machines (SVMs) [17] have been used to label documents as expressing a positive or negative opinion about the topic [39], or into a

five star rating about the topic [37] with respectable accuracy, getting these classifiers to work with reasonable accuracy on unlabeled data from a different domain is still very difficult.

This sentimental domain adaptation problem has many commercial applications [38]. Classifiers trained on one domain do poorly on other domains [1] [48]. One reason for this is that many of the most popular approaches use word level polarity associations determined from a training set or from a precomputed lexicon to determine topical sentiment. However, the polarity and strength of these words do not remain constant when transitioning from the training set or from a fixed lexicon to the domain of interest.

Most researchers dealing with a single domain recognize this factor and will provide an anecdotal example or two, but then present their research without any further discussion about the extent of the problem. A common example, "Read the book," [38] expresses one opinion in movie reviews and the opposite opinion in book reviews. Other common examples include "very long battery life" verses "a very long movie", and the "fuzzy teddy bear" verses the "fuzzy picture" of a camera or other electronic devices. In this study we provide data to numerically describe the degree of domain dependence exhibited for terms, observe a key pattern in this data, and describe an approach to exploit this pattern that allows us to build transferable sentiment lexicons.

This thesis research supports sentiment aware search and analysis engines that can adapt to individual topics at run-time. Building transferable domain-independent sentiment models, with accurate sentiment strength scores, is an im-

portant step to this goal. The simple and computationally cheap algorithms I describe allow for accurate real time sentimental document labeling and subsequent data mining.

5.2 Related Work

I used the exact product review data and sentence tokenization provided by Blitzer for the experiments on Structural Correspondence Learning with pivots selected using mutual information (SCL-MI) [4]. This is a high quality data set where labels have been provided by the reviewers in the form of a star rating. There are 25 domains included in this data set. Blitzer used four of these domains to evaluate SCL-MI.

Spectral Feature Alignment (SFA) [34] is similar to SCL in that both techniques attempt to find good domain-independent pivots. From these pivots they discover a mapping from features in the source domain to similar features in the target domain. Both have the same high level strategy, but with quite different approaches. SCL-MI selects pivots with the highest mutual information between the feature and the class label in the source domain. SFA chooses its pivots by selecting terms with the lowest mutual information between the feature and the domain. This process yields features that are domain-independent, but not necessarily (and probably not) very revealing about the class label. SFA then uses the co-occurrence of other features with these pivot features to create a bipartite graph. From this bipartite graph, they learn a mapping between features using spectral clustering.

SCL-MI uses pivot predictor weight vectors instead. These two approaches suggest that domain-independent features should be discriminative with respect to the classification problem, and occur with similar distributions throughout the various possible domains.

The type of approach used in SCL-MI and SFA is very different from my own. These approaches find and exploit a feature mapping between the source and the target domains. Instead, I build upon the previous chapter's results by using Delta IDF [25], in multiple domains, to find and score features that are informative in multiple domains. These features weights form a linear classifier for the target domain.

Delta IDF is a clean information retrieval based method to determine feature polarity and strength for many different kinds of textual binary classification problems. Delta IDF weights are computed, according to Equation 4.1, using the difference of the IDF scores for a term in the collection of all the positive labeled documents with the IDF score for all the negative documents. Consequently, the feature's sign indicates its semantic orientation. Mutual Information scores do not have this useful property. Delta IDF feature weights with simple smoothing have shown remarkable improvements for in-domain sentiment analysis tasks [33] ranging from blog data sets [32], to movie reviews [39], to product reviews [4]. Delta IDF feature weights have also been applied to other tasks, including subjectivity detection (opinionated vs. objective) and spam detection [26].

My general approach to combine knowledge about feature orientation extracted from multiple domains is similar to that of Drezde and Crammer [10], where

the authors combined multiple Confidence Weighted Linear Classifiers trained in separate domains into a single classifier. The single classifier is then applied to a new domain. They found that domain adaptation from multiple domains is more accurate than doing domain adaptation from a single domain. My results support this conclusion.

However, my approach is also quite different from Drezde and Crammer. Whereas creating a Confidence Weighted Linear Classifier is a minimization problem in the machine learning style, my single domain feature orientation scores are created using information retrieval metrics that are fast to calculate and easy to implement. As I show later on in the evaluation subsection, my approach is also more accurate. The method I use to combine multiple weight vectors into one weight vector was inspired by, and is similar to, the dot product. The mathematical and geometric interpretation to this process is clear and easy to understand. It also has a very low accuracy cost for transferring from multiple domains to a new target domain. In fact it is a bit more accurate than the in-domain version when there is a plethora of in-domain data, and quite a bit more accurate than in-domain classifiers when there is a scarcity of in-domain data.

Given my feature level focus, it might appear to be a bad sign that work on sentiment analysis with fusion training shows better accuracy with classifier level fusion than feature level fusion [22]. However, our focus on domain-independent features allows us to avoid some of the undesirable side effects of their algorithm. For their feature level classifier, they combined all of the training data into a single training set and then created a classifier on it. This completely ignores information

about which domain the data point comes from. To overcome this problem, they build a classifier for each of the product review categories and then combined them with meta-learning. In this way my feature level approach is actually more like their classifier level fusion than their feature level fusion. My results are still more accurate than their's, even though they used labeled data from the target domain while I did not.

The Domain Independence Verification Algorithm can be integrated into many other techniques by improving the quality and augmenting the size of the starting set of domain-independent terms. Not only could this improve SCL and SFA, it could also improve Semantic Orientation (SO) [47] calculations. The SO of a term is calculated using the Pointwise Mutual Information (PMI) of that term to a small list of known sentimental seed words. This is a popular approach that many techniques [28] [13] have built upon. However, even the most cited early works in sentiment analysis [39] recognize that manual lists frequently miss unintuitive but very revealing terms, and can be quite erroneous. Consequently, researchers have tried to mine sentiment terms from text [8]. To increase the size of these lists some researchers have turned to iterative bootstrapping methods [40]. There are many different ways to create dictionaries, several different types of dictionary creation algorithms, and some simple domain adaptation algorithms for dictionaries [14].

5.3 Dataset Description

5.3.1 Collection and Tokenization

This study uses the processed and balanced sentiment data-set extracted from Amazon product reviews for SCL-MI [4]. It is available for download at `http://www.cs.jhu.edu/~mdredze/datasets/sentiment/`. These Amazon reviews are organized by a two level hierarchy. They were segregated into categories by their class membership at the lowest level of the hierarchy. Amazon reviews come with an explicit five star rating provided by the writer of the review. These ratings were used by the authors cited above to label the review as positive or negative. Reviews with greater than three stars were deemed positive, while those with less than three stars were deemed negative. Reviews with a three star rating were discarded as ambiguous. Each review was transformed into a vector counting unigram and bigram words that occurred in the review text and the title of the review. Numbers that occurred in reviews were replaced with a special token "<num>" to indicate that some number was used.

While the data set was balanced to have a roughly even split between positive and negative reviews within each category, some categories are much more popular to review for than others. The second column of Table 5.4 shows the number of reviews for each target category.

5.3.2 Sentimental Domain Dependence Statistics

Definition 1 *Sentimental domain independence is the degree to which a term's sentimental bias remains unchanged in multiple domains.*

Sentimental domain independence is not a boolean flag: terms have varying degrees of sentimental domain independence. *Even some of the most generic strongly oriented sentimental words are left out in at least one of our product categories, or used with the opposite sentiment.* Figure 5.1 shows the count of how many features have each specific combination of positive or negative sentimental orientation for the 25 domains used in our dataset on a logarithmic scale. Going along the upper left edge of the figure features appear in more domains in a primarily negative context. Likewise, going perpendicular to this direction along the the upper right edge of the figure shows features that appear in more domains in a primarily positive context. The number of cells along the upper left direction indicate the number of domains in which a feature occurred in primarily negative documents, while the number of cells going perpendicular to it indicate the number of domains in which the feature occurred in primarily positive documents.

The topmost corner has no feature counts because it is the count of features that were positive in zero domains and negative in zero domains. Since there are only 25 domains and a feature can't take both orientations in a given domain, the counts in the lower triangle of the graph are all zero.

Notice how features that have the same orientation in all of the domains are quite rare, while features that occur in only one domain are incredibly common.

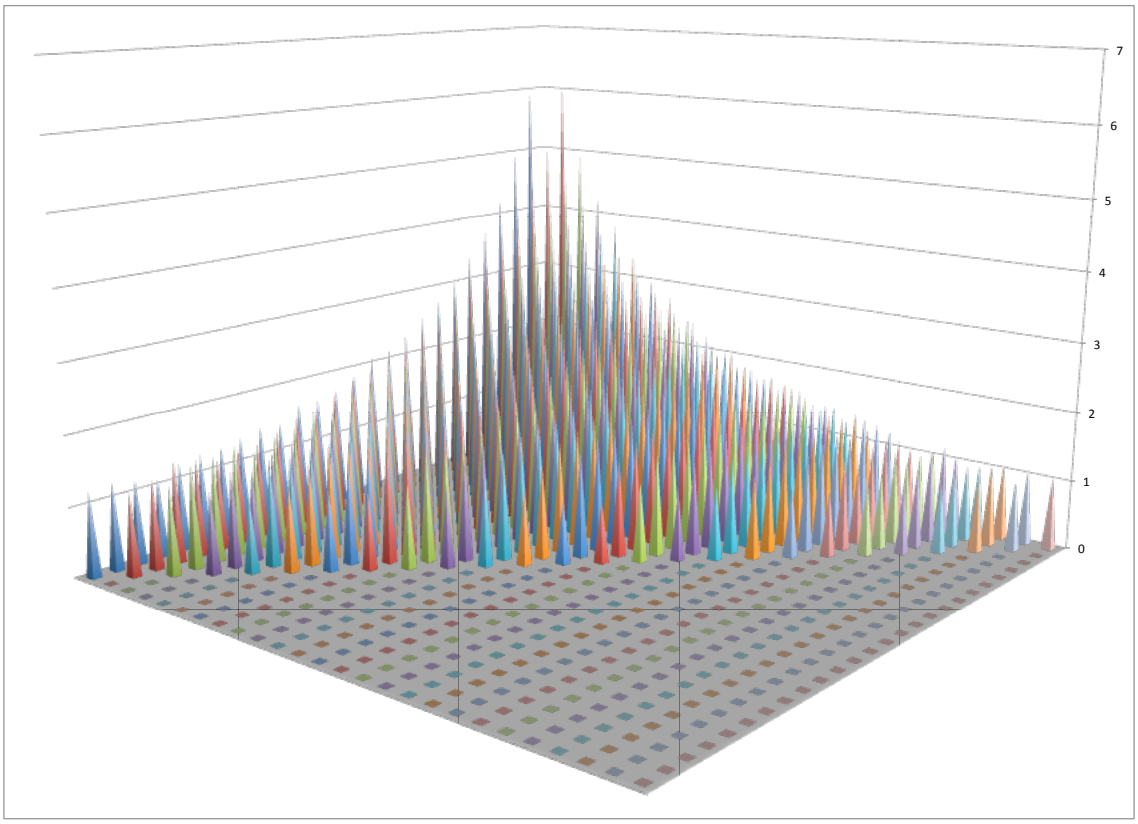

Figure 5.1: Log count of the number of features that are primarily negative in X domains and primarily positive in Y domains. Features are increasingly negative going to the right corner along the X axis, increasingly positive going to the left corner along the Y axis, and more widely used among the domains going down.

Progressing from the top corner to any corner of the graph shows that the values appear to drop off with a log-linear scale. In fact, moving away from the top right hand corner on any straight line the values appear to fall off in log-linear fashion. Therefore, the number of domain-independent features falls off logarithmically as the threshold for domain independence grows linearly.

5.4 Approach

Answer two key questions:

1. What is the bias in the source domain(s) of the term to transfer?

2. From this bias what can be concluded about its bias in the target domain?

Since it is very difficult to answer the second question directly, I shall substitute in a simple question to answer. Instead, determine if the bias for the feature is likely to remain the same in the training and the test data points. Assume that a term's bias is just as likely to change between two randomly selected domains as it is between two other randomly selected domains. In this case, the target domain can be ignored, and terms may be selected on the condition that they pass a confidence test for sentiment domain independence using labeled data from the training domains. Since Figure 5.1 shows that number of terms rapidly drops when greater confidence is required, it is important to balance confidence with the number of terms that will pass the test.

Definition 2 *Domain-independent sentiment terms are words or word pairs that pass a confidence test for sentimental domain independence.*

According to Definition 2, domain-independent sentiment terms are words or word pairs that pass a confidence test for sentimental domain independence. It follows that verifying term level sentimental orientation scores produced by Delta IDF in one domain with those produced by Delta IDF in multiple domains will produce domain-independent sentiment terms. The simplest way to do this is to eliminate any features that do not exist or do not have the same sign in all the domains selected for training. If a feature passes these restrictions, then multiply its weights in all of the selected domains to create its final weight. These rules are quite strict, so the number of training domains has been set to three to ensure

Figure 5.2: Top Delta IDF features for Books. Font size indicates relative strength. Positive features are in blue. Negative features are in red.

Figure 5.3: Top Delta IDF features found in the union of Books, DVDs, and Electronics with the same sentimental orientation. Font size indicates relative strength. Positive features are in blue. Negative features are in red.

that enough features would be available to use for classification. Once the features have been selected and weighted, then the final weight vector can be produced by balancing the positive and negative features and re-normalizing the vector to unit length. This process is the Domain Independence Verification Algorithm (DIVA). Figure 5.3 is a visualization of the top most positive and negative features found by DIVA using the books, DVDs, and electronics source domains. Please note how the domain-specific terms for books seen in Figure 5.2 are no longer present after running DIVA.

Feature level DIVA scores can be seen as bias scores that pass a simple confidence test. Pretend that a feature has an equal probability in any given domain of being either positive, negative, or absent. By observing only one domain their is no way to evaluate the confidence that the feature bias will be the same in the target domain. However, with a second domain their is a one in three chance that any given feature will have the orientation it exhibited in the first domain. Adding a second domain lowers this chance to one in nine. This threshold is slightly less demanding that the standard 95% confidence interval that is considered acceptable for most published research. Three domains for DIVA is an appropriate number.

Consider using all but the target domain as training domains for DIVA. This produces a very small list of features, or a classifier that ignores most of the data point. Domain independence is a relative thing: *there is almost always a topic where a normally positive word is used primarily in negative expressions or a topic where using that positive term would be out of place and is consequently missing.* Figure 5.1 illustrates this point.

The choice of how many domains to use for verification should be made on a per-genre basis. Domain independence statistics, data volume, and number of labeled domains varies with genre choice. DIVA may require a different number of input domains to reach the same confidence level for sentimental domain independence in the blog genre than it did for the product review genre, because the signal is less clear in the blog genre. Researchers should measure sentimental domain independence, plot it as shown in Figure 5.1, and decide the correct number of domains to use with DIVA just as I have done for product reviews.

5.5 Evaluation

The evaluation data set [4] is the 25 category product review dataset. Once again, I used the tokenization provided to enable direct comparisons with earlier work without anomalies created by different tokenization.

5.5.1 Single Domain Adaption

Although generic discriminative features found in multiple domains are the primary concern, measuring the discriminative power of domain-specific features on other domains is the first step. To measure this, Delta IDF weights created in one domain can be blindly applied to arbitrary target domains using the dot product classification rule as shown in Table 5.2. These results compare favorably to other domain adaptation results, like SCL-MI, on this data set. Table 5.1 shows the accuracy for each source/target category pair for SCL-MI. On average SCL-

Source	Target			
	Kitchen	Elect	DVDs	Books
Kitchen	*	**86.8**	**76.9**	68.6
Electronics	**85.9**	*	76.2	75.4
DVDs	81.4	74.1	*	**79.7**
Books	78.9	75.9	75.8	*

Table 5.1: Blitzer's SCL-MI domain adaptation accuracy going from the source domain to the target domain. Best results for each target domain are in bold.

Source	Target			
	Kitchen	Elect	DVDs	Books
Kitchen	*	**84.0**	79.1	77.0
Electronics	**87.8**	*	77.1	72.8
DVDs	78.5	75.1	*	**84.4**
Books	75.3	72.1	**82.9**	*

Table 5.2: Delta IDF dot product rule domain adaptation accuracy going from a single source domain to the target domain. Best results for each target domain are in bold.

MI is 77.97% accurate compared to the slightly higher average of 78.85% for blind Delta IDF transfer. This difference is not statistically significant with a p value of aproximately .4822 using a paired two tailed t-test.

It is surprising that just transferring the weights from one domain to another works as well as state of the art domain adaptation techniques. I expected this approach to be less accurate given how poorly SVMs (that use a similar feature space and classification rule) trained in one domain perform in a new domain and

given how computationally expensive SCL-MI is to run. One possible explanation for the unexpectedly good results is due to the algorithm's simplicity: This approach is less capable of overfitting the data than an SVM. Depending on the kernel function used, SVMs models can learn weights for a variety of more expressive, potentially non-linear, combinations of existing dimensions. Delta IDF can only learn a weight for each input dimension that is independent of all the other dimensions.

5.5.2 Domain-Independent Classification

Category	**DIVA**	In-Dom ΔIDF	SCL-MI + 50 Tar	SVM
Kitchen	88.9	89.5	85.9	87.7
Electronics	87.6	88.0	77.9	84.4
DVDs	85.1	86.2	78.5	82.4
Books	84.5	87.5	76.0	80.4

Table 5.3: Domain-independent Delta IDF weights vs alternative techniques and in-domain models.

Knowledge about the four domains examined in the preliminary single domain experiments is sufficient to build a strong domain-independent classifier. A DIVA vector was built, for each of the four target domains examined above, from the remaining three training domains. These vectors were each applied to their respective target domains as shown in Table 5.3.

SCL-MI is capable of using labeled data in the target domain to improve classification accuracy. For comparison purposes, the best source domains for SCL-MI

with 50 labeled instances are included in Table 5.3. Having outperformed those results, the in-domain SVM model Blitzer trained was also included. DIVA is significantly better than both. The confidence level for both experiments exceeds 98%. In fact, DIVA's results are comparable to the in-domain Delta IDF baselines. The p-value associated with a two tailed t-test between these two distributions is .1228. There is not enough evidence to reject the null hypothesis that DIVA is as good as in-domain Delta IDF classification. In practice, verifying a feature's sentimental orientation with two other domains then applying the dot product classification rule is as accurate as in-domain classification on similarly sized labeled datasets.

Domain-independent weight vectors are very accurate over a wide span of different domains. The product review dataset has 25 domains, leaving the 21 other domains to test on. I built in-domain Delta IDF weight vectors for these other domains and used them with the dot product classification rule to produce 21 additional in-domain baselines. These in-domain classifiers averaged 82.4% accuracy. From the four out-of-domain classifiers already built the multi-domain Delta IDF weight vector tested on the kitchen category was selected to be the general purpose model because it was the most accurate model built on the development data.This model used the in-domain Delta IDF weight vectors that were already built in the books, DVDs, and electronics domains. Henceforth, this model shall be called the BDE model. Using the dot product rule, the BDE classifier is on average 88.4% accurate over the other 21 domains, beating the in-domain baseline to the 99.5% confidence interval.

These results do not counteract the long held machine learning assumption

Target Category	Dom Size	In-Dom Model	General Model	Best Acc	Best Combination of 3 Source Domains
Apparel	2603	89.17	88.59	89.63	Books, Elect, Kitchen
Automotive	314	80.92	88.54	91.08	DVD, Elect, Sports
Baby	1756	89.41	87.70	89.12	Elect, Kitchen, Toys
Beauty	821	85.38	88.06	88.31	DVD, Kitchen, Music
Camera	1718	86.54	90.97	91.73	Elect, Health, Kitchen
Cell Phone	692	83.66	88.73	91.04	Apparel, Camera, Kitchen
Comp Games	485	72.77	83.71	86.60	Health, Music, Toys
Food	377	76.41	88.06	90.19	Apparel, DVD, Grocery
Grocery	654	84.25	86.54	88.38	Apparel, DVD, Music
Health	2713	87.36	88.24	89.24	DVD, Elect, Kitchen
Instruments	93	84.28	90.32	94.62	Elect, Kitchen, Mags
Jewelry	606	85.32	88.12	91.42	Apparel, Elect, Health
Magazines	1124	85.40	87.54	87.90	Books, Elect, Music
Music	20751	85.06	84.68	85.77	Books, DVD, Video
Office	109	76.32	90.83	92.66	Music, Sports, Video
Outdoor	593	84.13	89.88	92.24	Books, Elect, Kitchen
Software	1551	79.44	88.46	89.74	Elect, Kitchen, Music
Sports	2029	87.09	87.33	90.09	Baby, Camera, Kitchen
Tools	19	56.67	94.74	1	Software, Sports, Toys
Toys	4929	86.88	88.21	88.35	DVD, Elect, Kitchen
Video	4726	84.19	87.13	89.44	Books, DVD, Music

Table 5.4: General DIVA "BDE" model built on Books, DVDs, and Electronics vs. in-domain Delta IDF weights vs. the best DIVA combination of three domains on the target domain using brute force.

that models trained on in-domain data will do better than models built on a similar amount of out of domain. If the total amount of data points the BDE model is built on is reduced to the number of data points in the target domain, then the DIVA model's accuracy will be reduced to 74.23%. This is less accurate than the in-domain model to the 99.5% confidence interval. The point is that DIVA allows machine learning practitioners to leverage large amounts of out of domain data by providing them with a method to verify that the information learned from other

domains is still valid. The tools entry in Table 5.4 is a clear example of why this approach is so important: this is the entire point of domain adaptation.

While more training data is an advantage, DIVA works well for more than just this reason. DIVA preserves and intelligently uses the information captured by splitting the document pool into different domains. Consider building a Delta IDF dot product classifier using the union of all the data that the BDE model was trained on using DIVA. Table 5.5 shows how the DIVA model is more accurate than the Delta IDF classifier created from the union of the same set of documents at an accuracy of 88.4% to 84.24%. This difference is significant to the 99.95% confidence level.

A popular alternative technique to leverage more out of domain data is to use multiple classifiers under a weighted voting approach. Delta IDF dot product classification is particularly well suited to this approach because, when both the documents and the weight vectors are normalized to unit length, the magnitude of the dot product can serve as the vote's weight. Weighted voting using the books, DVDs, and electronics domain over the test domains is 84.28% accurate. The difference between weighted voting and the DIVA method using the same training and test points is significant to the 99.98% confidence level. This weighted voting approach is statistically no different than the union model as indicated by a p-value of .8515 . These results are also included in Table 5.5.

Target Category	Dom Size	In-Dom Model	DIVA Model	Union Model	Weighted Voting
Tools	19	56.67	94.74	78.95	78.95
Instruments	93	84.28	90.32	84.95	84.95
Office	109	76.32	90.83	84.4	85.32
Automotive	314	80.92	88.54	79.3	80.25
Food	377	76.41	88.06	84.35	83.82
Comp Games	485	72.77	83.71	83.92	80.21
Outdoor	593	84.13	89.88	86	86.34
Jewelry	606	85.32	88.12	86.14	86.8
Grocery	654	84.25	86.54	85.02	84.71
Cell Phone	692	83.66	88.73	76.3	77.6
Beauty	821	85.38	88.06	83.56	85.02
Magazines	1124	85.4	87.54	83.99	83.81
Software	1551	79.44	88.46	82.27	81.95
Camera	1718	86.54	90.97	86.26	87.37
Baby	1756	89.41	87.7	87.47	87.07
Sports	2029	87.09	87.33	86.1	85.9
Apparel	2603	89.16	88.59	86.52	87.28
Health	2713	87.36	88.24	81.87	82.53
Video	4726	84.19	87.13	88.17	87.24
Toys	4929	86.87	88.21	87.66	87.38
Music	20751	85.06	84.68	85.8	85.44

Table 5.5: General DIVA "BDE" model built on Books, DVDs, and Electronics vs. the Delta IDF model built on the same documents vs. weighted voting of books, DVDs, and electronics Delta IDF models built on the same set of documents. Results have been sorted by size and the in-domain accuracies have been left as a reference.

5.6 Future Work

Even though DIVA outperforms current domain adaptation methods, and is at least as accurate as the most advanced in domain techniques, there is still work to be done. Many of the design choices made raise tough questions. DIVA works by selecting features that pass a certain confidence level for domain independence as implemented by a constraint satisfaction problem. It is possible to implement this idea by explicitly evaluating this confidence level. What is the optimal confidence level that balances feature confidence versus the total number of features to achieve the highest out-of-domain classification accuracy? Is this a convex or a non-convex optimization problem?

Domain specific adaptation can produce more accurate classifiers for any given target domain than a general model. To prove this point I decided to take advantage of how fast DIVA is combined with the fact that it requires no parameter tuning, by using brute force. For every domain, a model built from every possible combination of three other domains was evaluated. Table 5.4 shows, every domain has some combination of three domains that performs better than the original general purpose books, DVDs, and electronics model. These differences are statistically significant to the 99.998% confidence level.

Brute force accuracy results for DIVA combinations of two domains, tested on the remaining domains, show a maximum accuracy of 88.5% for the DVDs and kitchen combination. This classifier is no better the BDE combination; It fails to pass a 40% confidence interval with BDE. On average, selecting any two domains is

about 82.0% accurate. Only selecting two domains for DIVA in the product review genre is not as good as selecting three domains.

It is not feasible right now to run a brute force test for DIVA combinations of four, but current samples using a greedy heuristic indicate that combinations of four are significantly worse than combinations of three. Since BDE is a strong classifier with a fairly large number of features it might be possible to filter some of the less sentimentally domain independent features out by adding another domain. If this process leaves a substantial number of features behind, their improved quality should help improve accuracy. The most likely domain that can be added to the model to accomplish this is the music domain. The music domain is very similar to the DVD domain, still quite similar to the electronics domain, and has some slight similarity to the books domain as they are both still works of art. Furthermore, the music domain is the largest domain with the greatest number of features. However, adding the music domain did not improve accuracy. The music, books, DVDs, and electronics model is only 87.65% accurate. This model is less accurate than the BDE model on every test domain. This result is statistically significant to 99.9985% confidence interval.

Early experiments have shown multiple factors including domain size, domain similarity, and Delta IDF classifier agreement on the target domain to be important factors that determine how well any particular DIVA combination will work on a target domain. Instead of using brute force on the accuracy statistic I found the brute force combination of three domains whose individual judgments on the target domain agree the most often. The average accuracy for this DIVA domain selection

technique on the 21 test domains is 87.93%. This model performs no differently than the BDE model; It fails to pass a 35% confidence interval with BDE. Future research should determine the best way to take advantage of domain level similarity to enhance classification accuracy by customizing the classifier to the target domain.

Due to the difficulty of satisfying the hard constraints posed by DIVA the majority of the domains in the data set contributed nothing to the creation of the general purpose model evaluated in Table 5.4. An approach implementing the high level idea of confidence testing domain independence for terms that can use an unlimited number of domains should be able to improve accuracy further.

The distribution of positive and negative documents varies between domains. How does this affect domain adaptation and how can it be used to improve accuracy?

5.7 Conclusion

I presented a technique to determine the sentimental polarity and relative strength of in-domain word and phrase level features that is incredibly accurate, easy to code, and computationally cheap. These feature weights can be directly applied using the dot product classification rule to accurately classify document sentiment. Furthermore, these in-domain feature values can be transferred to a new domain with no loss due to domain adaption simply by applying the Domain Independence Verification Algorithm. DIVA allows machine learning practitioners to bring to bear large amounts of out of domain data more effectively by leveraging the fact that these points came from different domains. DIVA produced the highest

domain adaption accuracy results to date on the gold standard multi-domain 25 category product review dataset published by Blitzer et al.

Chapter 6

Novel Feature Discovery in Unfamiliar Domains

Current domain adaptation approaches generally involve first transferring known features from the source domain to help determine the bias of novel features in the target domain. DIVA only determines which features to transfer to the target domain. The second step, identifying the bias of new features, usually uses the assumption that novel features in the target domain have the same sentimental orientation as the weighted combination of the features they are correlated to. However, since the novel feature's bias is determined by the already known bias of correlated features, when there are enough domain-independent features, this process does little to improve classification accuracy past the level that domain-independent features already allow. To improve domain adaptation accuracy future research should take a different approach.

6.1 Introduction and Related Work

Spectral Feature Alignment (SFA) [34] selects "domain-independent" terms by measuring the mutual information between the feature and the domain. Working on the standard assumption that similarly biased features will co-occur with each other, SFA uses the co-occurrence statistics to create a bipartite graph of domain-independent and domain-specific terms. These terms are spectrally clustered to pro-

duce a feature mapping that groups together sentiment features of similar strength. This feature space combines domain-specific terms of similar strength and sentimental orientation from both domains into a single unified dimension. When a classifier is trained on source data represented in this feature space, the domain-specific terms from the source domain provide useful information about similar domain-specific terms in the target domain.

SFA's bipartite graph is built solely upon co-occurrence statistics with the domain-independent terms. Every part of the algorithm after that is dependent upon the assumption that similarly biased features have similar co-occurrence statistics. However, since these features co-occur so highly with the domain-independent features, they almost never occur in situations where the domain-independent terms would be insufficient to classify the document. Structural Correspondence Learning with pivots selected by Mutual Information (SCL-MI) [4] shares the same fundamental assumption and consequently shares the same problem.

6.2 Experimental Design

Hypothesis : *Novel features in the target domain share their bias score with the domain-independent features they are correlated to. If these novel features are included in the domain-independent model they will yield a statistically significant improvement to domain adaptation accuracy.*

Assigning the bias and weight of domain-specific features by their correlation to known domain-independent terms is an effective way to determine their bias,

but these new terms are not very informative. The easiest way to test this fact is to compare the judgments of a classifier built with the domain-independent terms against a classifier built with only the correlated domain-specific terms.

This kind of test is easy to setup using the algorithms I have already described. Simply label documents in a new domain with a domain-independent classifier and then mine terms from those documents using Delta IDF. The terms mined with Delta IDF can then be filtered to remove the terms from the domain-independent model. Finally, compare the judgments both classifiers make to see if the new domain-specific terms disagree with the original labels. If they do not then the extra step to mine domain-specific terms is not making judgments any more accurate, and thus research effort should be directed to more fruitful areas. Although this test does not evaluate the actual implementation of algorithms like SCL-MI and SFA it does test the shared assumption that in the aggregate, features correlated at the document level will share the same biases and that the domain-independent features provide additional information that will make document level judgments more accurate.

First, build a domain-independent model (call it the BDE model) using DIVA from the books, DVDs, and electronics domains. In this test, Delta IDF used probabilistic smoothing, as described in Chapter 5. The differences between Delta IDF with simple add-one smoothing and probabilistic smoothing are not statistically significant. The results reported in Table 6.1 show the accuracy of this BDE model.

Second, classify documents in the target domain with the BDE model. Then, train an in-domain Delta IDF model for each domain using the new labels generated by the domain-independent BDE model. I used probabilistic smoothing with Delta

IDF for consistency. Third, remove from the new set of models all features in the BDE model. This ensures that the domain-independent words have no bearing in the domain-specific classifier. For most domains these features appeared to be oriented to the correct sentiment with emotionally strong terms appearing in the top 50 word cloud visualizations. Figure 6.1 is a representative example of these models. The terms for each domain appear to be more domain specific than those shown in the DIVA visualizations. The only exception to these promising and indicative weight vectors is the Tools weight vector that was built on only 19 data points. The terms shown in Figure B.23 frequently indicate the wrong polarity and there are too many terms that appear to be sentiment neutral. The original in-domain Delta IDF weight vector for tools shares the same problems. Please see Appendix B for the complete set of model visualizations.

Use the novel feature models to classify documents. Finally, calculate the Pearson's correlation coefficient for document level judgments between each pair of domain-independent and domain-specific classifiers. If the labels these two models produce are highly correlated then novel features only serve to reinforce the decisions made by the domain-independent model. This would indicate that novel feature mining by their correlation to domain-independent terms is not by itself an effective way to improve domain-independent classification.

Figure 6.1: Top domain-specific Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red.

6.3 Results and Discussion

Table 6.1 shows that using only derived domain-specific terms to classify documents is just as accurate as the domain-independent classifier they were created from. This positive news implies that the domain-specific terms are properly oriented and weighted. However, in depth analysis at the document level reveals a key finding: these extra domain-specific terms contribute no useful new information on top of the domain-independent terms for document level classification purposes.

The Pearson's correlation coefficient calculated for each pair of domain-independent and domain-specific classifiers show that they clearly agree on document labels for the vast majority of all documents. In fact, for half of the test domains these classifiers make exactly the same set of judgments. Many EM approaches would be forced

Target Category	Dom Size	Only Domain Specific Model	General Model	Pearson Correlation
Apparel	2603	88.74	88.59	0.9954
Automotive	314	88.54	88.54	1
Baby	1756	87.59	87.70	0.9977
Beauty	821	88.06	88.06	1
Camera	1718	90.8	90.97	0.9916
Cell Phone	692	88.73	88.73	1
Comp Games	485	83.51	83.71	0.9958
Food	377	88.06	88.06	1
Grocery	654	86.54	86.54	1
Health	2713	88.43	88.24	0.9960
Instruments	93	90.32	90.32	1
Jewelry	606	87.95	88.12	0.9967
Magazines	1124	87.37	87.54	0.9929
Music	20751	84.67	84.68	0.9876
Office	109	90.83	90.83	1
Outdoor	593	89.88	89.88	1
Software	1551	88.27	88.46	0.9884
Sports	2029	87.19	87.33	0.9951
Tools	19	94.74	94.74	1
Toys	4929	88.5	88.21	0.9870
Video	4726	87.03	87.13	0.9979

Table 6.1: General DIVA model built on Books, DVDs, Electronics with probabilistic smoothing (Difference not statistically significant compare to simple add one smoothing) vs. domain-specific Delta IDF weights built using the labels produced by the general model. Terms in the general model were removed from the domain-specific model. Note that both classifiers when run on the same documents make nearly identical judgements as exhibited by the Pearson correlation coefficient.

to terminate after the first iteration.

It should not come as much of a surprise to readers that weighting new features in proportion to the features that they are correlated with will produce features that serve only to reinforce the weight of the original features. If the initial set of features are accurate for the domain and large enough to coverage the test, then they should be sufficient to ensure that any newly generated features will not be needed to classify documents.

6.4 Future Work

Although these novel features do not improve document-level classification there are still uses for them. These very accurate domain-specific features could be very helpful for sentence-level sentiment analysis. While it is very likely that an entire document will use enough domain-independent terms to facilitate accurate domain-independent classification, it is much less likely that this will remain true for sentences. Good authors avoid boring their readers by using a variety of synonyms. Authors also attempt to use more specific words when possible, especially when describing things in detail. In these circumstances, derived domain-specific terms mined as described in the approach could be very helpful.

Domain-specific features extracted from derived labels in unfamiliar domains are more revealing about the specific features of the domain. It should be possible to leverage domain-specific features to improve sentimental information extraction about specific aspects of the topic.

Sentiment search could benefit from identifying terms like the ones produced by this approach. First, sentimental query expansion could benefit from a weighted list of mined query specific bias terms. Better query expansion should improve recall. However, even if these terms do not boost recall because they always occur with the pre-expansion terms, their presence will allow for a better estimate of the document's relevance. This should boost document relevance scores and improve early precision. Given that most users do not scroll past the first page of results, this could result in a large functional improvement. Similarly, document re-weighting

approaches could also benefit.

6.5 Conclusion

Although domain-specific terms have almost no impact on document level classification in the product review corpus, these features are still accurate and can be used for other important natural language processing and data mining tasks. At the document level domain-independent sentiment terms are common enough to accurately estimate a document's polarity, but at the sentence level domain-specific terms could help fill in the gaps between domain-independent sentiment terms. In an information retrieval setting these features may be able to increase recall and should boost document relevance scores and improve early precision. Document summarization algorithms might also want to prefer these terms.

This experiment strongly suggests that finding domain-specific features with a similar bias to known domain-independent features by their document level correlation is not a very promising approach in spite of the amount of research effort that has gone into it. Finding more terms to reinforce existing judgments will not produce an improvement over existing judgements. Instead, a more appropriate approach to investigate may be to search for domain-specific terms that have the opposite sentimental orientation when compared to any seeds they are correlated to. These terms could mitigate the effects of the domain-independent terms, possibly causing a different judgment call on documents based upon better contextual knowledge.

Chapter 7

Genre Adaptation

The genre signal is primarily defined by the data source or medium the author uses to communicate. Writing styles change with the publishing medium. Books, news articles, blog posts, surveys, product reviews, status updates, and tweets have very different styles. Length, stop-word and near stop-word usage, personal pronoun usage, abbreviations, formatting, degree of editing, and the tendency to stay on topic are affected by the publishing medium. This creates a unique signal that I call the genre signal. The genre signal is only important to the degree that it affects the problem specific signal.

Genre adaptation is the process of building a classifier in one genre to be used on a different genre. This task is even more challenging than domain adaptation. While the exact term frequencies and their biases may vary between two different domains, if these domains are within the same genre, then most other things stay the same. It is immediately obvious that some genres are more formal (News Articles) than others (Twitter). This results in problems caused by: varying sentence length, decreasing grammatical fidelity, more prevalent or unfamiliar abbreviations, and shorter documents.

7.1 Introduction and Related Work

While these problems are easy to foresee, and receive a lot of research attention, they are not as problematic as the community has generally feared them to be. Instead, larger documents have proven to be a greater challenge. An interesting multi-genre sentiment analysis study [3] covering twitter posts, the TREC 06 corpus [32], micro reviews from Blippr, and the classic movie reviews corpus [39] concluded that the brevity and unusual artifacts found in informal domains like twitter to be advantageous.

The problem with larger documents from a genre adaptation perspective is rooted in topic shift. *It is very hard for human beings to restrain themselves from wandering off topic.* My own observations on the TREC dataset indicate that the majority of all written content for any given post is not relevant to the query topic. Better document segmentation would alleviate this problem greatly. Other sentiment analysis researchers working on financial blogs for stock related opinions found that most relevant documents shifted topics from one stock to another in the course of a document [31]. They found that segmenting documents using a fixed window of words around the topic was more effective than competing sentence and paragraph level document segmentation approaches, but that all three approaches were better than document level classification.

7.2 Genre Adaptation Experiments

Definition 3 *Sentimental genre independence is the degree to which a term's sentimental bias remains unchanged in multiple genres.*

Definition 4 *Genre-independent sentiment terms are words or word pairs that pass a confidence test for sentimental genre independence.*

This set of experiments shows the level of sentimental genre independence exhibited by domain-independent terms. The domain-independent DIVA models built on topically focused product reviews are less genre independent than they are domain independent. To support this conclusion the books, DVDs, and electronics model ("BDE model") was applied without any changes to directly classify documents in two new genres from two different datasets. The first data set is the TREC 06 sentiment search blog track corpus. The second dataset is the Irish news dataset collected between the late summer and early fall of 2009. It was annotated for sentiment about the state of the irish economy by a set of 33 users [5]. It is available for download at `http://mlg.ucd.ie/files/datasets/sentiment.zip`.

7.2.1 TREC Topic Sentiment Prediction

To control for some degree of noise documents that were longer than 250 words were sent through a filter process. Terms in these TREC documents that were farther from a query term than 25 words were removed. Documents were represented as a bag of unigrams and bigrams. There were no further processing steps: there was no stop word removal, and no part of speech tagging.

The BDE model over all 50 queries is on average 64.53% accurate. This is a noticeable drop from the average in-domain Delta IDF baseline of 76.67%. The BDE model results compare favorably with in-domain 5-fold cross validation results using Maximum Entropy classifiers with either unigrams (60.6%) or bigrams and trigrams (60.35%) [24]. However, other in-domain approaches which add part of speech information, stop wording, in conjunction with an SVM can achieve 69% accuracy [3]. The BDE model is about as accurate as other popular published in-domain approaches even if it is not as accurate as my own in-domain approach.

Although the in-domain Delta IDF approach is more accurate than just classifying with the BDE model, the scripts to build a model failed in two domains because there were less than 10 data points to divide among the 10 folds.

More importantly, this test still shows that DIVA is an appropriate way to transfer domain information because the BDE model is more accurate than either the books model at 54.42% accuracy, the DVDs model at 57.15%, or the electronics model at 50.32%. This test illustrates that transferring a domain-specific model from a different genre is only a little better than tossing a coin.

One of the main applications for sentiment classification on web documents is to determine if people like or dislike the topic in general. This task is less restrictive than document level classification. If we frame this question as "Are the documents on this topic primarily positive or primarily negative?" then the BDE model can predict the right majority class for 40 out the 50 TREC queries.

7.2.2 Irish News Financial Prediction

The dataset of positive or negative news about the Irish economy is not available as raw text documents. Instead the documents have been tokenized into a bag of unigrams. Posts were labeled by multiple annotators by the dataset providers. The posts were labeled as having a positive, negative, or irrelevant impact on the Irish economy at the time of publication. To produce a single gold standard label for each document I used majority voting between the positive and negative labels with ties being discarded. Irrelevant labels were ignored.

The BDE model over is 63.02% accurate. This is a noticeable drop from the in-domain Delta IDF baseline of 72.15%. The dataset's creators report a 77% accuracy with an SVM [6]. This is the first time in-domain Delta IDF dot product classification has failed to outperform an SVM baseline. I speculate that Delta IDF is suffering in this case because a bag of unigrams is not as expressive as a bag of bigrams. Only having a bag of unigrams would be less detrimental to SVMs, which are much more capable of both fitting the data and overfitting the data. An SVM has the ability to create new dimensions that are a weighted combination of other dimensions, which in effect creates a form of bigram information.

Keep in mind that the BDE model built on product reviews is a much better model of the irish economic dataset than either the books model at 54.46% accuracy, or the DVDs model at 45.08% accuracy, and is still a better model than the grossly imbalanced electronics model at 61.66% accuracy. Please note that the actual bias in the labels is about 39.25% positive. 4.37% of all labels generate by the electronics

model are positive as compared to 27.23% of all labels being positive for the BDE model. Of the four models examined the BDE model is not only the most accurate model, it is also the model that most closely mirrors the actual class bias of the dataset. This last property is especially important for the kinds of economic tasks that such a model would be used for. A model that just predicts the majority class for this kind of task is useless: Investors are particularly interested in finding positive news during a negative period.

7.3 Discussion and Future Work

After working on positive or negative sentiment classification in two new genres it is increasingly clear that off topic information and differences between the meaning of sentimental annotations are substantial research challenges. These observations are the result of numerous hours examining queries and documents where things went wrong. Designing a set of experiments to isolate each of these components is an important future research project for genre independent classification.

Off topic information can be very misleading. Sometimes off topic sentiment is just noise with no bearing on the sentiment of the actual query target. However, sometimes the off topic sentiment is inversely related to sentiment of the topic. The TV show "Arrested Development" was widely popular, with a 9.6 out of 10 rating on IMDB with over 62 thousand user ratings. In the TREC blog craw there were 149 positive documents about the show and only two negative documents. However, the BDE model did no better than guessing. After eyeballing the reviews the reasons for

this were obvious. The vast majority of the text was not about how good Arrested Development was, but rather, it was about how much they hated Fox for canceling the show. Logically this implies a love for the show, which was not lost on NIST assessors when they labeled the dataset. In-domain classifiers had no trouble at all determining that the kind of negative words used to deride Fox for canceling the show indicated positive sentiment about the show.

This kind of phenomenon is not limited to just a few queries. The query about disgraced lobbyist Jack Abramoff's ties with then President George Bush is another obvious example. During this time period neither George Bush nor Jack Abramoff were well liked to say the least. People hated them individually, and any tie between them would certainly be a negative thing, yet somehow you got the feeling that the authors were quite pleased when further evidence linked them together. To detect this a system would need to know that people like when bad things happen to their enemies.

In these kind of situations domain independent sentiment analysis is going to be very ineffective without either a reasoning engine, some kind of greater world knowledge, or precise target of sentiment information. Even with precise target of sentiment information it still requires something extra to realize that negative sentiment about a domain-specific item "X" implies positive sentiment about another domain-specific item "Y".

These observations beg the question, "What do the sentimental annotations in each domain mean?" In the product review genre positive annotations mean that the product was worth the price. In a movie review genre price seems to be

less important, instead a positive review means that the entertainment value of the movie was worth the person's time. In the Irish news genre a positive annotation meant that the document had a positive impact on Irish economic transactions. In the TREC corpus a positive annotation meant that the author approved of the query topic even if the topic was objectively negative such as being connected to a "disgraced lobbyist", or making or having made insulting controversial "sexist or racist" remarks . While there is a great deal of commonality between topical sentimental associations in different genres there are still substantial differences.

The most obvious way to improve accuracy would be to apply a domain independence verification style approach to multiple genres to create a Genre Independent Vocabulary Extraction System (GIVES). A first cut of this algorithm would use DIVA vectors built in multiple domains as a new set of vectors for DIVA, or a similarly motivated algorithm, to do feature selection and weighting upon.

7.4 Conclusion

Although DIVA did not outperform baselines created on the target genre and domain pair, the technique still beats baseline genre adaptation approaches. DIVA has proven resistant to erroneously transferring the overall class bias of the source domain and resistant to excessive classifier bias in the target domain. For many real world applications this is sufficient to determine the general bias about a subject of interest. Additionally, DIVA weights are compatible with and can be integrated into other approaches and other problems easily.

Chapter 8

Conclusion

8.1 Contributions

This dissertation makes several discoveries about text analysis and domain adaptation that lead to concrete measures and algorithms, which are supported by classification results on the de facto standard evaluation datasets.

The first discovery in this dissertation describes the three kinds of information and types of noise that exist in all text classification problems. These signals are the genre signal, the domain signal, and the classification task signal. The three signals are not orthogonal: they interact with each other. Feature presence and meaning are affected differently by each of these three signals. Controlling for the effects of the genre and the domain signals will improve sentiment classification accuracy.

The second discovery is that all supervised machine learning based textual domain adaptation approaches answer two questions: First, "What is the bias in the source domain(s) of the term to transfer?" and second, "From this bias what can be concluded about its bias in the target domain?" Correctly answering these two questions is sufficient to achieve high accuracy. Finding classification problem specific terms that are unlikely to change their bias when transfered to the target domain is an acceptable, albeit suboptimal, answer to the second question.

Text analysis researchers have been informally using the terms domain, genre,

and domain independence but this is the first work to provide a formal definition for them. The definitions for sentimental domain independence and domain-independent sentiment terms indicate what criteria to optimize that will yield an accurate domain-independent model.

The general strategy used by the leading approaches in the field (SFA, SCL-MI) spend a considerable amount of research attention and add a similar amount of complexity to answer other questions that turn out to be largely irrelevant. These approaches use the fact that domain-specific features in the target domain share approximately the same bias with the domain-independent features they are correlated to. These approaches assume that if these novel domain-specific features are included in the domain-independent model, then they will yield a statistically significant improvement to domain adaptation accuracy. I demonstrated that the domain-specific term biases, produced from their co-occurrence to domain-independent features of the same bias, could be used to accurately classify documents in the target domain with no better nor no worse accuracy than what the domain-independent features could already do. These domain-specific terms only reinforce the same judgments the existing domain-independent baseline makes. This assumption has been relied upon inside most domain adaptation algorithms implying that other parts of these multifaceted approaches are producing most of their accuracy and that these algorithms should be evaluated piecewise or step by step. This contribution should help guide future research into more profitable directions.

These discoveries have yielded several practical contributions including Delta IDF and DIVA.

Delta IDF is an intuitive, easy to implement, fast, and effective feature weighting algorithm. These feature weights can be used to pre-weight documents before training in a classifier, or they can be used with the dot product rule to directly classify documents. In both cases they outperform state of the art techniques like an SVM. While Delta IDF weights were designed for textual sentiment analysis they have been tested and found to be quite effective on dozens of different domains, in multiple different genres, and on several different classification problems. In particular, the Delta IDF dot product classification results form the first complete in-domain baseline for the popular 25 category Amazon product review dataset. Providing the first complete in-domain baseline for this dataset, that is also the most accurate in-domain baseline for the more popular product categories, is a valuable service to the domain adaptation community, and a minor contribution in its own right. Most importantly, this contribution supports the more important algorithmic contribution of Delta IDF.

DIVA is a fast, simple, and effective constraint satisfaction technique to identify domain-independent terms and then derive their classification task specific bias. This algorithm flowed directly from the discovery of the two sufficient conditions for a domain-independent classifier. Like Delta IDF the terms and term weights produced by DIVA can be used for other downstream NLP tasks.

On the gold standard evaluation data set for domain adaptation DIVA substantially raises the bar. DIVA does more than just beat existing domain adaptation approaches and cutting edge semi-supervised learning approaches, even the best in-domain approaches are inferior. Models built with DIVA demonstrate an incredible

level of accuracy over dozens of different domains.

8.2 Future Work

Genre adaptation accuracy still has room for improvement. The definitions for domain and genre independence and the example set forth by DIVA pave the way for future improvements. A DIVA like approach for genre adaptation seems to be an obvious first step. However, this requires a multi-genre dataset that is possibly delimited by domain within the genres. Given the differences of opinion on the meaning of positive and negative sentiment judgments produced by various labeling tasks, I do not believe that bundling multiple dataset together will achieve a high quality multi-genre sentiment analysis dataset. Producing a sufficiently large multi-domain dataset with a uniform standard for sentiment labels is a very big task.

DIVA is very accurate using only a small subset of the domains for training. The nature of domain independence makes it prohibitive to use too many domains with DIVA. A statistical confidence test based implementation of the domain independence confidence constraints would allow DIVA to use all of the available domains for training. This should improve accuracy by having better confidence estimates and simply by having more training data.

Sarcasm has always been outside the scope of this thesis. However, this research has brought to my attention other more common linguistic constructions that could have an even bigger impact on genre adaptation. Sarcasm is not the only case

where the total weight of a person's expressed sentiment on the topic runs counter to their held sentiment. It is good that bad things happen to bad people. Explicit and or gratuitous gloating is uncouth, consequently positive sentiment terms will be considerably rarer than they should be in these situations. This phenomenon is quite common, but current domain adaptation approaches will not get the correct sentiment.

Classifiers built to work on a specific target domain of unlabeled data should be more accurate than a domain-independent model built without any knowledge of the target domain. The experiment using the best brute force combination of 3 domains for any given target domain shows that there is still room to improve over a domain-independent model using the same number of training domains. However, selecting these domains is an open problem. In a more general sense, it is still unclear how to leverage knowledge about domain-independent terms into a better classifier for any given domain with a pool of unlabeled data.

8.3 Summary

This dissertation characterized and categorized textual expressions into three fundamental types of information: genre, domain, and task specific knowledge; it described algorithms that identified, isolated, and transferred the problem specific information found in the source domains to improve textual sentiment analysis problems with little or no labeled data. The thesis specified exactly what high-level ideas an algorithm must implement to do domain adaptation accurately. My DIVA imple-

mentation of these high-level ideas is the most accurate approach on the standard domain adaptation dataset.

Appendix A

In-Domain Delta IDF Model Visualizations

This is a compilation of model visualizations for all 25 product review categories. The models were produced using Delta IDF on the set of labeled documents for the specified category.

Figure A.1: Top Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red.

Figure A.2: Top Delta IDF features found in the **automotive** domain. Positive Features are in blue. Negative Features are in red.

Figure A.3: Top Delta IDF features found in the **baby products** domain. Positive Features are in blue. Negative Features are in red.

Figure A.4: Top Delta IDF features found in the **beauty products** domain. Positive Features are in blue. Negative Features are in red.

Figure A.5: Top Delta IDF features found in the **books** domain. Positive Features are in blue. Negative Features are in red.

Figure A.6: Top Delta IDF features found in the **cameras** domain. Positive Features are in blue. Negative Features are in red.

Figure A.7: Top Delta IDF features found in the **cell phones and cellular service** domain. Positive Features are in blue. Negative Features are in red.

Figure A.8: Top Delta IDF features found in the **computer and video games** domain. Positive Features are in blue. Negative Features are in red.

Figure A.9: Top Delta IDF features found in the **DVDs** domain. Positive Features are in blue. Negative Features are in red.

Figure A.10: Top Delta IDF features found in the **electronics** domain. Positive Features are in blue. Negative Features are in red.

Figure A.11: Top Delta IDF features found in the **gourmet food** domain. Positive Features are in blue. Negative Features are in red.

Figure A.12: Top Delta IDF features found in the **groceries** domain. Positive Features are in blue. Negative Features are in red.

Figure A.13: Top Delta IDF features found in the **health and personal fitness** domain. Positive Features are in blue. Negative Features are in red.

Figure A.14: Top Delta IDF features found in the **jewelry and watches** domain. Positive Features are in blue. Negative Features are in red.

Figure A.15: Top Delta IDF features found in the **kitchen and housewares** domain. Positive Features are in blue. Negative Features are in red.

Figure A.16: Top Delta IDF features found in the **magazines** domain. Positive Features are in blue. Negative Features are in red.

Figure A.17: Top Delta IDF features found in the **music** domain. Positive Features are in blue. Negative Features are in red.

Figure A.18: Top Delta IDF features found in the **musical instruments** domain. Positive Features are in blue. Negative Features are in red.

Figure A.19: Top Delta IDF features found in the **office products** domain. Positive Features are in blue. Negative Features are in red.

Figure A.20: Top Delta IDF features found in the **outdoor living** domain. Positive Features are in blue. Negative Features are in red.

Figure A.21: Top Delta IDF features found in the **software** domain. Positive Features are in blue. Negative Features are in red.

Figure A.22: Top Delta IDF features found in the **sports and outdoors** domain. Positive Features are in blue. Negative Features are in red.

Figure A.23: Top Delta IDF features found in the **tools and hardware** domain. Positive Features are in blue. Negative Features are in red.

Figure A.24: Top Delta IDF features found in the **toys and games** domain. Positive Features are in blue. Negative Features are in red.

Figure A.25: Top Delta IDF features found in the **video** domain. Positive Features are in blue. Negative Features are in red.

Appendix B

Novel Features Visualizations

This is a compilation of model visualizations for all 25 product review categories. The models were produced by labeling all the product reviews with the DIVA model built on books, DVDs, and electronics. These new labels were used to train a Delta IDF model on each category. Terms in the DIVA model were removed to show only new features discovered by this process.

Figure B.1: Top domain specific Delta IDF features found in the **apparel** domain. Positive Features are in blue. Negative Features are in red.

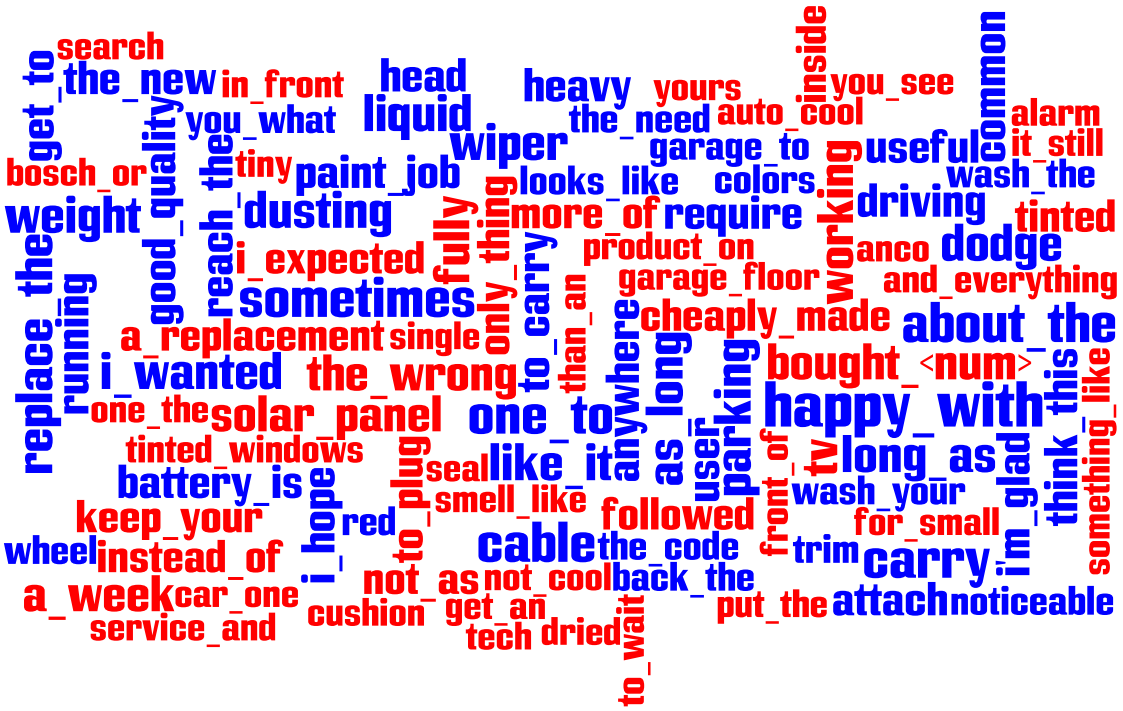

Figure B.2: Top domain specific Delta IDF features found in the **automotive** domain. Positive Features are in blue. Negative Features are in red.

Figure B.3: Top domain specific Delta IDF features found in the **baby products** domain. Positive Features are in blue. Negative Features are in red.

Figure B.4: Top domain specific Delta IDF features found in the **beauty products** domain. Positive Features are in blue. Negative Features are in red.

Figure B.5: Top domain specific Delta IDF features found in the **books** domain. Positive Features are in blue. Negative Features are in red.

Figure B.6: Top domain specific Delta IDF features found in the **cameras** domain. Positive Features are in blue. Negative Features are in red.

Figure B.7: Top domain specific Delta IDF features found in the **cell phones and cellular service** domain. Positive Features are in blue. Negative Features are in red.

Figure B.8: Top domain specific Delta IDF features found in the **computer and video games** domain. Positive Features are in blue. Negative Features are in red.

Figure B.9: Top domain specific Delta IDF features found in the **DVDs** domain. Positive Features are in blue. Negative Features are in red.

Figure B.10: Top domain specific Delta IDF features found in the **electronics** domain. Positive Features are in blue. Negative Features are in red.

Figure B.11: Top domain specific Delta IDF features found in the **gourmet food** domain. Positive Features are in blue. Negative Features are in red.

Figure B.12: Top domain specific Delta IDF features found in the **groceries** domain. Positive Features are in blue. Negative Features are in red.

Figure B.13: Top domain specific Delta IDF features found in the **health and personal fitness** domain. Positive Features are in blue. Negative Features are in red.

Figure B.14: Top domain specific Delta IDF features found in the **jewelry and watches** domain. Positive Features are in blue. Negative Features are in red.

Figure B.15: Top domain specific Delta IDF features found in the **kitchen and housewares** domain. Positive Features are in blue. Negative Features are in red.

Figure B.16: Top domain specific Delta IDF features found in the **magazines** domain. Positive Features are in blue. Negative Features are in red.

Figure B.17: Top domain specific Delta IDF features found in the **music** domain. Positive Features are in blue. Negative Features are in red.

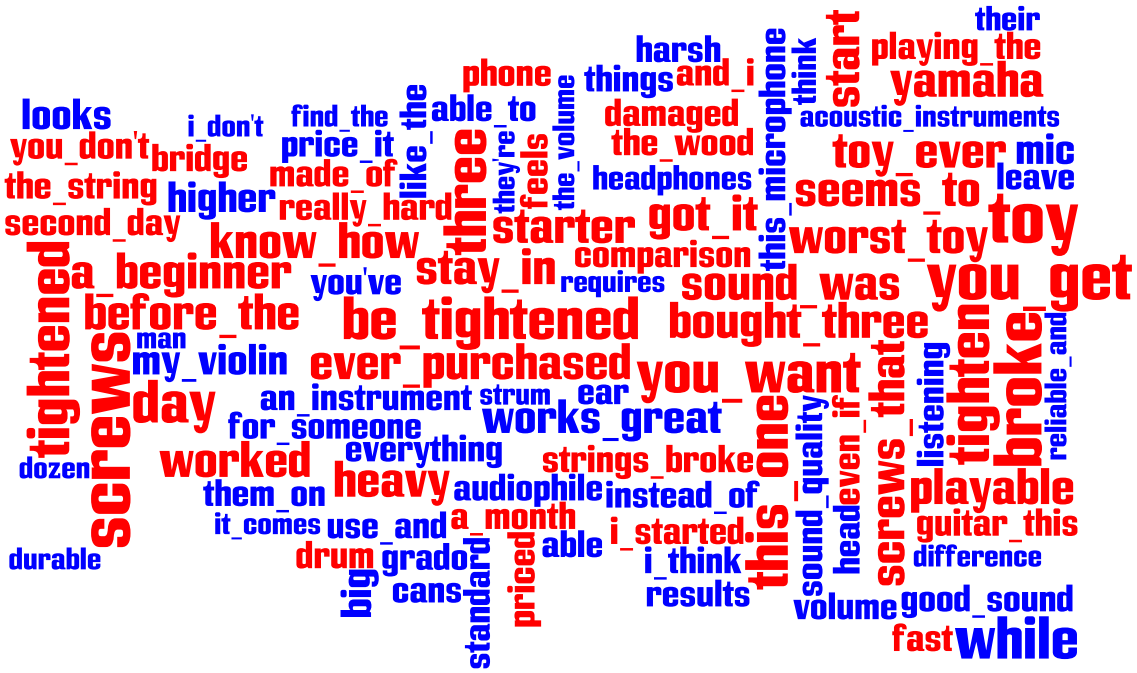

Figure B.18: Top domain specific Delta IDF features found in the **musical instruments** domain. Positive Features are in blue. Negative Features are in red.

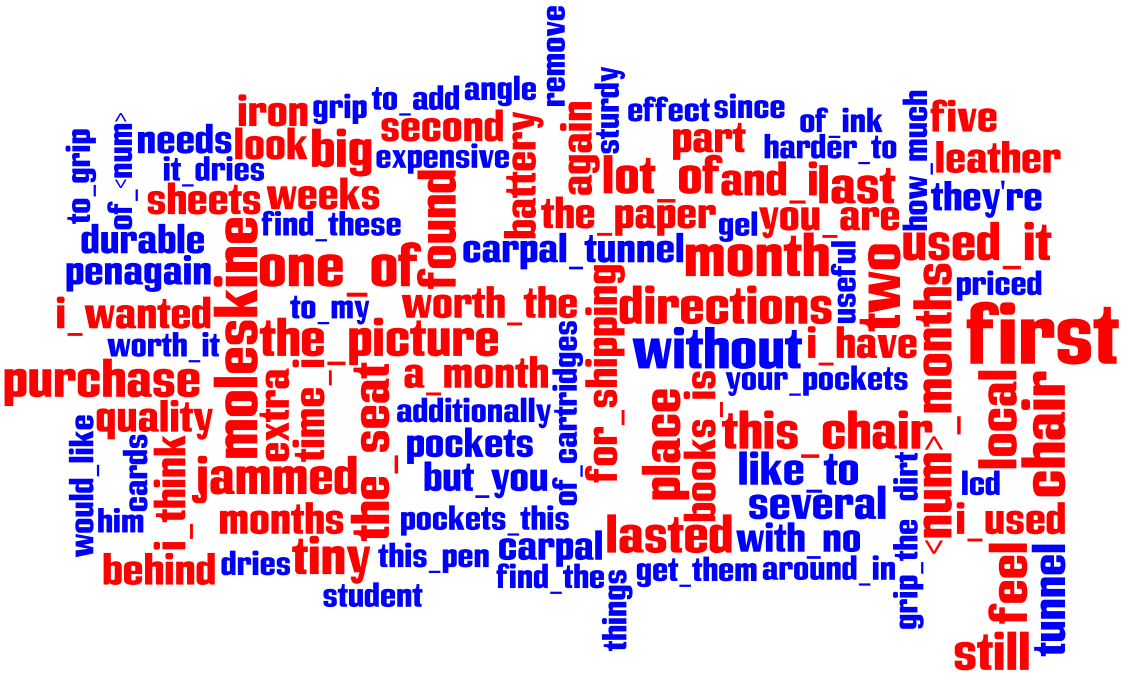

Figure B.19: Top domain specific Delta IDF features found in the **office products** domain. Positive Features are in blue. Negative Features are in red.

Figure B.20: Top domain specific Delta IDF features found in the **outdoor living** domain. Positive Features are in blue. Negative Features are in red.

Figure B.21: Top domain specific Delta IDF features found in the **software** domain. Positive Features are in blue. Negative Features are in red.

Figure B.22: Top domain specific Delta IDF features found in the **sports and outdoors** domain. Positive Features are in blue. Negative Features are in red.

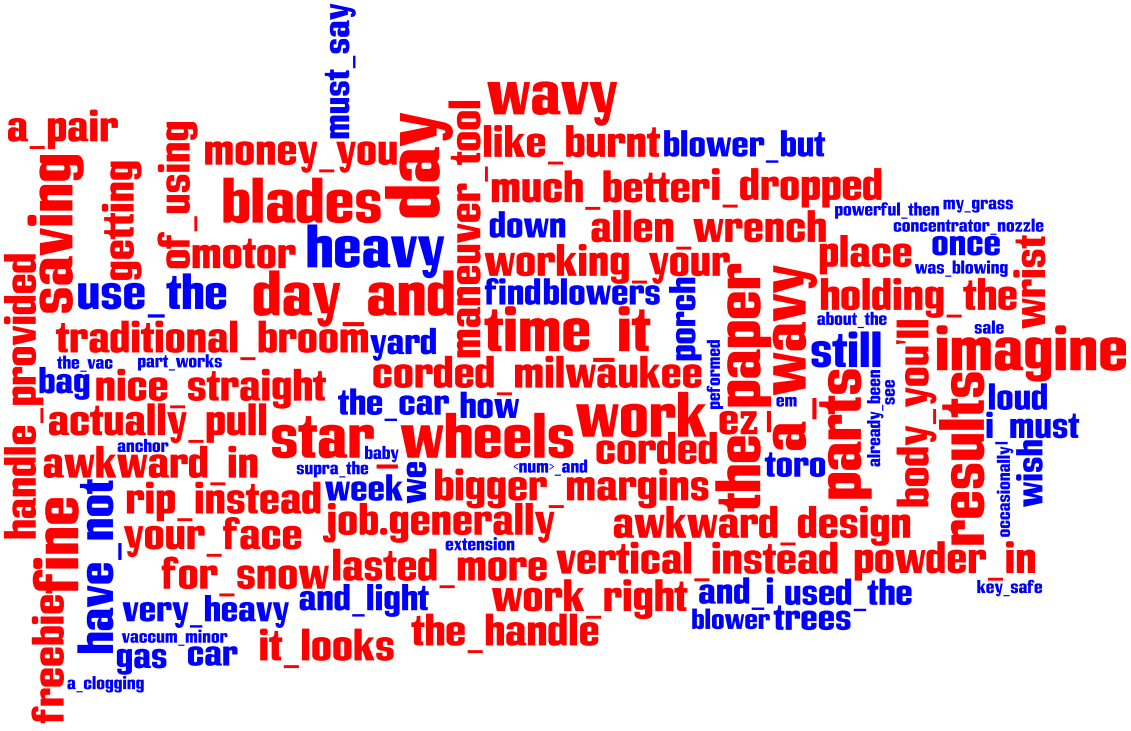

Figure B.23: Top domain specific Delta IDF features found in the **tools and hardware** domain. Positive Features are in blue. Negative Features are in red.

Figure B.24: Top domain specific Delta IDF features found in the **toys and games** domain. Positive Features are in blue. Negative Features are in red.

Figure B.25: Top domain specific Delta IDF features found in the **video** domain. Positive Features are in blue. Negative Features are in red.

Bibliography

[1] A. Aue and M. Gamon. Customizing sentiment classifiers to new domains: a case study. In *Proceedings of RANLP*, 2005.

[2] M. Baker and J. Wurgler. Investor Sentiment and the Cross-Section of Stock Returns. *The Journal of Finance*, 61(4):1645–1680, 2006.

[3] A. Bermingham and A.F. Smeaton. Classifying sentiment in microblogs: is brevity an advantage? In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1833–1836. ACM, 2010.

[4] J. Blitzer, M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Annual Meeting-Association for Computational Linguistics*, volume 45, pages 440–447, 2007.

[5] A. Brew, D. Greene, and P. Cunningham. A machine learning system for tracking sentiment in irish economic news. *Artificial Intelligence and Cognitive Science*, pages 285–286, 2010.

[6] A. Brew, D. Greene, and P. Cunningham. Using crowdsourcing and active learning to track sentiment in online media. In *Proceeding of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pages 145–150. IOS Press, 2010.

[7] W. Dai, G.R. Xue, Q. Yang, and Y. Yu. Transferring naive bayes classifiers for text classification. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22, page 540. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[8] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *WWW*, pages 519–528, 2003.

[9] X. Ding, B. Liu, and P.S. Yu. A holistic lexicon-based approach to opinion mining. In *Proceedings of the International Conference on Web Search and Web Data Mining*, pages 231–240. ACM New York, NY, USA, 2008.

[10] M. Dredze and K. Crammer. Online methods for multi-domain learning and adaptation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 689–697. Association for Computational Linguistics, 2008.

[11] ST Dumais and TK Landauer. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological review*, 104:211–240, 1997.

[12] C.J. FILLMOR. Types of lexical information.

[13] M. Gamon and A. Aue. Automatic identification of sentiment vocabulary: exploiting low association with known sentiment terms. In *Proceedings of the ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing*, pages 57–64. Association for Computational Linguistics, 2005.

[14] S. Godbole, I. Bhattacharya, A. Gupta, and A. Verma. Building re-usable dictionary repositories for real-world text mining. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1189–1198. ACM, 2010.

[15] J. Jiang and C.X. Zhai. Instance weighting for domain adaptation in nlp. In *Annual Meeting-Association for Computational Linguistics*, volume 45, page 264, 2007.

[16] T. Joachims. *Text Categorization with Support Vector Machines: Learning with Many Relevant Features*. Springer, 1997.

[17] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. *Machine Learning: ECML-98*, pages 137–142, 1998.

[18] T. Joachims. *Making large-scale support vector machine learning practical*. MIT Press Cambridge, MA, USA, 1999.

[19] J. Klein, Y. Moon, and RW Picard. This computer responds to user frustration: Theory, design, and results. *Interacting with Computers*, 14(2):119–140, 2002.

[20] B. Klimt and Y. Yang. Introducing the enron corpus. In *First Conference on Email and Anti-spam (CEAS)*, 2004.

[21] E. Leopold and J. Kindermann. Text Categorization with Support Vector Machines. How to Represent Texts in Input Space? *Machine Learning*, 46(1):423–444, 2002.

[22] S. Li and C. Zong. Multi-domain sentiment classification. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 257–260. Association for Computational Linguistics, 2008.

[23] H.W. Lilliefors. On the kolmogorov-smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association*, pages 399–402, 1967.

[24] F. Liu, B. Li, and Y. Liu. Finding opinionated blogs using statistical classifiers and lexical features. In *Third International AAAI Conference on Weblogs and Social Media*, 2009.

[25] J. Martineau and T. Finin. Delta TFIDF: An Improved Feature Space for Sentiment Analysis. In *Proceedings of the Third AAAI International Conference on Weblogs and Social Media*. AAAI Press, 2009.

[26] J. Martineau, T. Finin, A. Joshi, and S. Patel. Improving binary classification on text problems using differential word features. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, pages 2019–2024. ACM, 2009.

[27] D. McCarthy. Relating wordnet senses for word sense disambiguation. In *Proceedings of ACL Workshop on Making Sense of Sense, 2006*, 2006.

[28] T. Mullen and N. Collier. Sentiment analysis using support vector machines with diverse information sources. *Proceedings of EMNLP*, pages 412–418, 2004.

[29] T. Mullen and R. Malouf. A preliminary investigation into sentiment analysis of informal political discourse. In *Proceedings of the AAAI Symposium on Computational Approaches to Analyzing Weblogs*, pages 159–162, 2006.

[30] R. Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10, 2009.

[31] N. O'Hare, M. Davy, A. Bermingham, P. Ferguson, P. Sheridan, C. Gurrin, and A.F. Smeaton. Topic-dependent sentiment analysis of financial blogs. In *Proceeding of the 1st International CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 9–16. ACM, 2009.

[32] I. Ounis, M. Rijke, C. Macdonald, GA Mishne, and I. Soboroff. Overview of the TREC-2006 Blog Track. *TREC 2006 Working Notes*, pages 15–27.

[33] G. Paltoglou and M. Thelwall. A study of Information Retrieval weighting schemes for sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1386–1395. Association for Computational Linguistics, 2010.

[34] S.J. Pan, X. Ni, J.T. Sun, Q. Yang, and Z. Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th International Conference on World Wide Web*, pages 751–760. ACM, 2010.

[35] S.J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, pages 1345–1359, 2009.

[36] B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *Proceedings of the ACL*, pages 271–278, 2004.

[37] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 115–124. Association for Computational Linguistics, 2005.

[38] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, 2008.

[39] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP 2002*, 2002.

[40] E. Riloff and R. Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the National Conference on Artificial Intelligence*, pages 474–479. JOHN WILEY & SONS LTD, 1999.

[41] M.D. Rudd. Suicide warning signs in clinical practice. *Current Psychiatry Reports*, 10(1):87–90, 2008.

[42] M.D. Rudd, A.L. Berman, T.E. Joiner Jr, M.K. Nock, M.M. Silverman, M. Mandrusiak, K. Van Orden, and T. Witte. Warning Signs for Suicide: Theory, Research, and Clinical Applications. *Suicide and Life-Threatening Behavior*, 36(3):255–262, 2006.

[43] M.D. Silber, A. Bhatt, Police Dept, N. York, and Intelligence Division. *Radicalization in the West: The Homegrown Threat*. Police Dept., 2007.

[44] A. Sloman. Beyond shallow models of emotion. *Cognitive Processing*, 2(1):177–198, 2001.

[45] C. Strapparava, A. Gliozzo, and C. Giuliano. Pattern abstraction and term similarity for word sense disambiguation: Irst at senseval-3. In *Proceedings of SENSEVAL-3 Third International Workshop on Evaluation of Systems for the Semantic Analysis of Text*, pages 229–234, 2004.

[46] Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of EMNLP*, pages 327–335, 2006.

[47] Peter D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424, 2002.

[48] M. Whitehead and L. Yaeger. Building a general purpose cross-domain sentiment mining model. In *Computer Science and Information Engineering, 2009 WRI World Congress on*, volume 4, pages 472–476. IEEE, 2009.

[49] C. Whitelaw, N. Garg, and S. Argamon. Using appraisal groups for sentiment analysis. *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pages 625–631, 2005.

[50] B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21st International Conference on Machine learning*, page 114. ACM, 2004.

[51] O.F. Zaidan, J. Eisner, and C.D. Piatko. Using "Annotator Rationales" to Improve Machine Learning for Text Categorization. *Proceedings of NAACL HLT*, pages 260–267, 2007.