

Semantic Signatures for Example-Based Linguistic Metaphor Detection

Michael Mohler and David Bracewell and David Hinote and Marc Tomlinson

Language Computer Corp.

Richardson, Texas, USA

{michael,david,dhinote,marc}@languagecomputer.com

Abstract

Metaphor is a pervasive feature of human language that enables us to conceptualize and communicate abstract concepts using more concrete terminology. Unfortunately, it is also a feature that serves to confound a computer’s ability to comprehend natural human language. We present a method to detect linguistic metaphors by inducing a domain-aware semantic signature for a given text and compare this signature against a large index of known metaphors. By training a suite of binary classifiers using the results of several semantic signature-based rankings of the index, we are able to detect linguistic metaphors in unstructured text at a significantly higher precision as compared to several baseline approaches.

1 Introduction

Metaphor is a widely-used literary mechanism which allows for the comparison of seemingly unrelated concepts. It has been thoroughly studied in both the linguistics literature (Ahrens et al., 2003; Lakoff and Johnson, 1980; Tourangeau and Sternberg, 1982; Wilks, 1978) and more recently within the field of computational linguistics.¹ Although there have been many influential theories regarding the cognitive basis of metaphor, the most prominent among them is Lakoff’s Contemporary Theory of Metaphor (Lakoff and Johnson, 1980; Lakoff, 1993), which popularized the idea of a *conceptual*

metaphor mapping. Within the cognitive framework of a given conceptual mapping, terms pertaining to one concept or domain (the *source*) can be used figuratively to express some aspect of another concept or domain (the *target*). For example, the conceptual metaphor “Life is a Journey” indicates a medium within which the target concept “life” may be more easily discussed and understood. This particular mapping allows us to speak of one being stuck in a “dead-end” job, a crucial decision as being a “fork in the road”, or someone’s life “taking a wrong turn”.

By allowing us to discuss an abstract target concept using the vocabulary and world knowledge associated with a more familiar source concept, metaphor serves as a vehicle for human communication and understanding, and as such, has been found to be extremely prevalent in natural language, occurring as often as every third sentence (Shutova et al., 2010). As a consequence of this ubiquity, it is crucial that any system tasked with the understanding of natural language be capable of detecting the presence of metaphor in text and of modeling the intended semantic content of the metaphoric expression. In this work, we first induce a domain-sensitive semantic signature which we define as a set of highly related and interlinked WordNet (Fellbaum, 1998) senses drawn and augmented from a text that may be used to place the text within the semantic space of a metaphoric concept. We then employ a suite of binary classifiers to detect metaphoricity within a text by comparing its semantic signature to a set of known metaphors. If the semantic signature of the text closely matches the signature of a known metaphor, we propose that it is likely to represent

¹For a broad survey of the relevant literature, see Shutova (2010).

Example	Metaphor
Obama heard a bomb ticking in his left ear.	No
Obama heard another political bomb ticking, this time in his left ear.	Yes

Table 1: The top sentence describes a literal bomb ticking, while the bottom sentence uses metaphoric language to describe an impending political disaster.

an instance of the same conceptual metaphor. To facilitate this work, we have built an index of known metaphors within a particular target domain. We have selected the domain of *Governance* which we define broadly to include electoral politics, the setting and enactment of economic policy, and the creation, application, and enforcement of rules and laws.

The problem of metaphor as it relates to computer understanding is illustrated in the example sentences of Table 1. A strictly literal reading suggests that the two sentences are describing something very similar. At the very least, the semantics of the phrases “bomb ticking” and “in his left ear” are indistinguishable without the added knowledge that the second sentence is using metaphor to convey information about something altogether different from explosives and body parts. From the context of the full sentences, it is clear that while the first sentence is straightforwardly describing Obama and his perception of a literal bomb, the second is describing an impending political crisis as though it were a bomb. Rather than a literal “ear” this sentence uses the phrase “in his left ear” to suggest that the source of the crisis is on the political “left”. In order for an automated system to correctly understand the intended meaning of these sentences, it must first be aware that the text under consideration is not to be taken literally, and given this knowledge, it must employ all available knowledge of the underlying conceptual mapping to appropriately interpret the text in context.

The remainder of this work is organized as follows. In Section 2, we survey related work in semantic representation and linguistic metaphor identification. Section 3 describes in detail our approach to metaphor identification through the use of semantic signatures. In Section 4, we discuss the setup of our experiment which includes the creation of our metaphor index as well as the extraction and annotation of our training and testing data sets. Finally,

we show the results of our experiments in Section 5 and share our conclusions in Section 6.

2 Related Work

The phenomenon of metaphor has been studied by researchers across multiple disciplines, including psychology, linguistics, sociology, anthropology, and computational linguistics. A number of theories of metaphor have been proposed, including the Contemporary Theory of Metaphor (Lakoff, 1993), the Conceptual Mapping Model (Ahrens et al., 2003), the Structure Mapping Model (Wolff and Gentner, 2000), and the Attribute Categorization Hypothesis (McGlone, 1996). Based on these theories, large collections of metaphors have been assembled and published for use by researchers. The Master Metaphor List (MML) (Lakoff, 1994) groups linguistic metaphors together according to their conceptual mapping, and the Hamburg Metaphor Database (HMD) (Eilts and Lönneker, 2002) for French and German fuses EuroWordNet synsets with the MML source and target domains for a robust source of metaphoric semantics in those languages.

In recent years, the computational linguistics community has seen substantial activity in the detection of figurative language (Bogdanova, 2010; Li and Sporleder, 2010; Peters and Wilks, 2003; Shutova, 2011) one aspect of which is the identification of metaphoric expressions in text (Fass, 1991; Shutova et al., 2010; Mason, 2004). Much of the early work on the identification of metaphor relied upon hand-crafted world knowledge. The met* (Fass, 1991) system sought to determine whether an expression was literal or figurative by detecting the violation of selectional preferences. Figurative expressions were then classified as either metonymic, using hand-crafted patterns, or metaphoric, using a manually constructed database of analogies. The CorMet (Mason, 2004) system determined the

source and target concepts of a metaphoric expression using domain-specific selectional preferences mined from Internet resources. More recent work has examined noun-verb clustering (Shutova et al., 2010) which starts from a small seed set of one-word metaphors and results in clusters that represent source and target concepts connected via a metaphoric relation. These clusters are then used to annotate the metaphoricity of text.

Similar to our work, the Metaphor Interpretation, Denotation, and Acquisition System (MIDAS) (Martin, 1990) employed a database of conventional metaphors that could be searched to find a match for a metaphor discovered in text. If no match was found, the metaphoric text was replaced with a more abstract equivalent (e.g. a hypernym) and the database was searched again. If a match was found, an interpretation mapping was activated, and the novel metaphor would be added to the database for use in future encounters. Unfortunately, this technique was limited to interpreting known metaphors (and descendants of known metaphors) and was unable to detect truly novel usages. By expanding the metaphors using a more robust semantic signature, we attempt to transcend this limitation thereby producing a more durable system for metaphoric example linking.

An additional vein of metaphor research has sought to model the human processing of metaphor as a semantic space within which source and target concepts can be placed such that the similarity between their representations within this space (i.e. semantic vectors) can be sensibly quantified (Katz, 1992; Utsumi, 2011). One computational example of this approach (Kintsch, 2000) has employed latent semantic analysis (LSA) (Landauer and Dumais, 1997) to represent the semantic space of the metaphors in a reduced dimensionality (i.e. using singular value decomposition). In their approach, metaphors were represented as a set of terms found using a spreading activation algorithm informed by the terms' independent vector relatedness to the source and target concepts within some LSA space. By contrast, we have chosen to represent the metaphoric space using WordNet senses which have been shown in previous work (Lönneker, 2003) to represent a viable representation language for metaphor. We believe that the ontological knowl-

edge encoded in the semantic relationships of WordNet represents an improvement over the distributional relatedness encoded within an LSA vector.

Also of relevance to the construction and use of semantic signatures is current research on the induction of topic signatures. A topic signature is a set of related words with associated weights which define and indicate the distinct topics within a text. In their work on automated summarization, Lin and Hovy (2000) developed a method for the construction of topic signatures which were mined from a large corpus. Similarly, Harabagiu and Lacatusu (2005) explored the use of topic signatures and enhanced topic signatures for their work on multi-document summarization. By contrast, we explore the use of semantic signatures which serve to enrich the semantics of the source and target frame concepts being expressed in a text for the purpose of detecting the presence of metaphor.

3 Methodology

In this work, we approach the task of linguistic metaphor detection as a classification problem. Starting from a known target domain (i.e. *Governance*), we first produce a target domain signature which represents the target-specific dimensions of the full conceptual space. Using this domain signature, we are able to separate the individual terms of a sentence into source frame elements and target frame elements and to independently perform a semantic expansion for each set of elements using WordNet and Wikipedia as described in our earlier work (Bracewell et al., 2013). Taken together, the semantic expansions of a text's source frame elements and target frame elements make up the full semantic signature of the text which can then be compared to an index of semantic signatures generated for a collection of manually detected metaphors. We use as features for our classifiers a set of metrics that are able to quantify the similarity between the given semantic signature and the signatures of metaphors found within the index.

3.1 Constructing a Target Domain Signature

In order to produce a semantic representation of the text, we first build a target domain signature, which we define as a set of highly related and interlinked

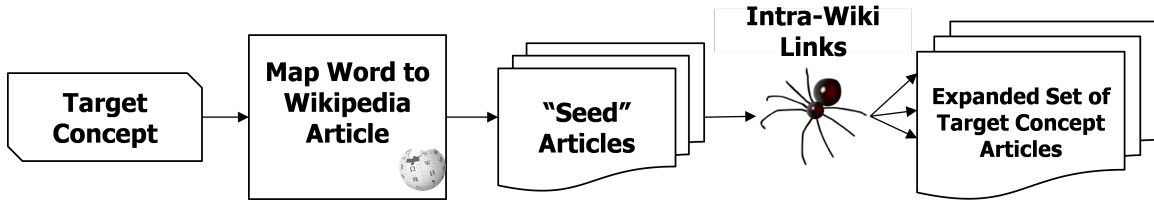


Figure 1: Focused crawling of Wikipedia articles pertaining to the target concept using intra-wiki links

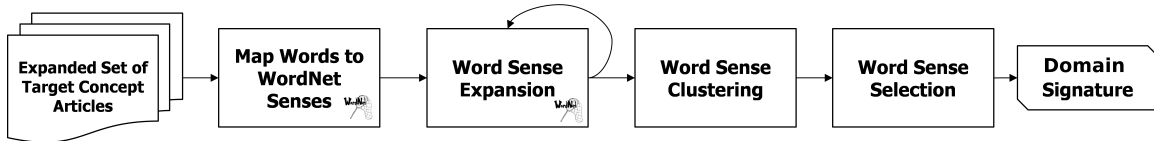


Figure 2: Constructing the domain signature of the target concept from Wikipedia articles pertaining to the target concept

WordNet senses that correspond to our particular target domain with statistical reliability. For example, in the domain of *Governance* the concepts of “law”, “government”, and “administrator”, along with their associated senses in WordNet, are present in the domain signature. We generate this signature using semantic knowledge encoded in the following resources: (1) the semantic network encoded in WordNet; (2) the semantic structure implicit in Wikipedia; and (3) collocation statistics taken from the statistical analysis of a large corpora. In particular, we use Wikipedia as an important source of world knowledge which is capable of providing information about concepts, such as named entities, that are not found in WordNet as shown in several recent studies (Toral et al., 2009; Niemann and Gurevych, 2011). For example, the organization “Bilderberg Group” is not present in WordNet, but can easily be found in Wikipedia where it is listed under such categories as “Global trade and professional organizations”, “International business”, and “International non-governmental organizations”. From these categories we can determine that the “Bilderberg Group” is highly related to WordNet senses such as “professional organization”, “business”, “international”, and “nongovernmental organization”.

We begin our construction of the domain signature by utilizing the semantic markup in Wikipedia to collect articles that are highly related to the target concept by searching for the target concept (and

optionally content words making up the definition of the target concept) in the Wikipedia article titles and redirects. These articles then serve as a “seed set” for a Wikipedia crawl over the intra-wiki links present in the articles. By initiating the crawl on these links, it becomes focused on the particular domain expressed in the seed articles. The crawling process continues until either no new articles are found or a predefined crawl depth (from the set of seed articles) has been reached. The process is illustrated in Figure 1. The result of the crawl is a set of Wikipedia articles whose domain is related to the target concept. From this set of articles, the domain signature can be built by exploiting the semantic information provided by WordNet.

The process of going from a set of target concept articles to a domain signature is illustrated in Figure 2 and begins by associating the terms contained in the gathered Wikipedia articles with all of their possible WordNet senses (i.e. no word sense disambiguation is performed). The word senses are then expanded using the lexical (e.g. derivationally related forms) and semantic relations (e.g. hypernym and hyponym) available in WordNet. These senses are then clustered to eliminate irrelevant senses using the graph-based Chinese Whispers algorithm (Biemann, 2006). We transform our collection of word senses into a graph by treating each word sense as a vertex of an undirected, fully-connected graph where edge weights are taken to be the product of the Hirst and St-Onge (1998) WordNet similarity be-

tween the two word senses and the first-order corpus cooccurrence of the two terms. In particular, we use the normalized pointwise mutual information as computed using a web-scale corpus.

The clusters resulting from the Chinese Whispers algorithm contain semantically and topically similar word senses such that the size of a cluster is directly proportional to the centrality of the concepts within the cluster as they pertain to the target domain. After removing stopwords from the clusters, any clusters below a predefined size are removed. Any cluster with a low² average normalized pointwise mutual information (npmi) score between the word senses in the cluster and the word senses in the set of terms related to the target are likewise removed. This set of target-related terms used in calculating the npmis are constructed from the gathered Wikipedia articles using TF-IDF (term frequency inverse document frequency), where TF is calculated within the gathered articles and IDF is calculated using the entire textual content of Wikipedia. After pruning clusters based on size and score, the set of word senses that remain are taken to be the set of concepts that make up the target domain signature.

3.2 Building Semantic Signatures for Unstructured Text

After constructing a signature that defines the domain of the target concept, it is possible to use this signature to map a given text (e.g. a sentence) into a multidimensional conceptual space which allows us to compare two texts directly based on their conceptual similarity. This process begins by mapping the words of the text into WordNet and extracting the four most frequent senses for each term. In order to improve coverage and to capture entities and terms not found in WordNet, we also map terms to Wikipedia articles based on a statistical measure which considers both the text of the article and the intra-wiki links. The Wikipedia articles are then mapped back to WordNet senses using the text of the categories associated with the article.

In the next step, source and target frame elements of a given text are separated using the WordNet senses contained in the target domain signature.

²We define low as being below an empirically defined threshold, τ .

Terms in the text which have some WordNet sense that is included in the domain signature are classified as target frame elements while those that do not are considered source frame elements. Figure 3 shows an overview of the process for determining the source and target concepts within a text. The remainder of the signature induction process is performed separately for the source and target frame elements. In both cases, the senses are expanded using the lexical and semantic relations encoded in WordNet, including hypernymy, domain categories, and pertainymy. Additionally, source frame elements are expanded using the content words found in the glosses associated with each of the noun and verb senses. Taken together, these concepts represent the dimensions of a full conceptual space which can be separately expressed as the source concept dimensions and target concept dimensions of the space.

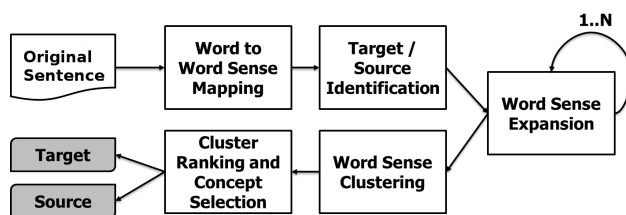


Figure 3: Example of a generated conceptual space for a given text. In this work, only one iteration of the sense expansion is performed.

In order to determine the correct senses for inclusion in the semantic signature of a text, clustering is performed using the same methodology as in the construction of the domain signature. First, a graph is built from the senses with edge weights assigned based on WordNet similarity and cooccurrence. Then, the Chinese Whispers algorithm is used to cluster the graph which serves to disambiguate the senses and to prioritize which senses are examined and incorporated into the source concept dimensions of the conceptual space. Word senses are prioritized by ranking the clusters based on their size and on the highest scoring word sense contained in the cluster using:

$$rank(c) = size(c) \cdot \left(\frac{\sum_s score(s)}{|c|} \right) \quad (1)$$

where c is the cluster, s is a word sense in the clus-

ter, and $|c|$ is the total number of word senses in the cluster. The senses are scored using: (1) the degree distribution of the sense in the graph (more central word senses are given a higher weight); and (2) the length of the shortest path to the terms appearing in the given text with concepts closer to the surface form given a higher weight. Formally, $score(s)$ is calculated as:

$$score(s) = \frac{degree(s) + dijkstra(s, R)}{2} \quad (2)$$

where $degree(s)$ is degree distribution of s and $dijkstra(s, R)$ is the length of the shortest path in the graph between s and some term in the original text, R .

Clusters containing only one word sense or with a score less than the average cluster score (μ_c) are ignored. The remaining clusters and senses are then examined for incorporation into the conceptual space with senses contained in higher ranked clusters examined first. Senses are added as concepts within the conceptual space when their score is greater than the average word sense score (μ_s). To decrease redundancy in the dimensions of the conceptual space, neighbors of the added word sense in the graph are excluded from future processing.

3.3 Classification

Given a semantic signature representing the placement of a text within our conceptual space, it is possible to measure the conceptual distance to other signatures within the same space. By mapping a set of known metaphors into this space (using the process described in Section 3.2), we can estimate the likelihood that a given text contains some metaphor (within the same target domain) by using the semantic signature of the text to find the metaphors with the most similar signatures and to measure their similarity with the original signature.

We quantify this similarity using five related measures which are described in Table 2. Each of these features involves producing a score that ranks every metaphor in the index based upon the semantic signature of the given text in a process similar to that of traditional information retrieval. In particular, we use the signature of the text to build a query against which the metaphors can be scored. For each

word sense included in the semantic signature, we add a clause to the query which combines the vector space model with the Boolean model so as to prefer a high overlap of senses without requiring an identical match between the signatures.³

Three of the features simply take the score of the highest ranked metaphor as returned by a query. Most simply, the feature labeled *Max Score (naïve)* uses the full semantic signature for the text which should serve to detect matches that are very similar in both the source concept dimensions and the target concept dimensions. The features *Max Score (source)* and *Max Score (target)* produce the query using only the source concept dimensions of the signature and the target concept dimensions respectively.

The remaining two features score the metaphors within the source dimensions and the target dimensions separately before combining the results into a joint score. The feature *Max Score (joint)* calculates the product of the scores for each metaphor using the source- and target-specific queries described above and selects the maximum value among these products. The final feature, *Joint Count*, represents the total number of metaphors with a score for both the source and the target dimensions above some threshold (μ_j). Unlike the more naïve features for which a very good score in one set of dimensions may incorrectly lead to a high overall score, these joint similarity features explicitly require metaphors to match the semantic signature of the text within both the source and target dimensions simultaneously.

Altogether, these five features are used to train a suite of binary classifiers to make a decision on whether a given text is or is not a metaphor.

4 Experimental Setup

One crucial component of our linguistic metaphor detection system is the index of metaphors (in the domain of *Governance*) against which we compare our candidate texts. As a part of this project, we have produced an ever-growing, metaphor-rich dataset taken from political speeches, political websites (e.g. Communist Party USA, Tea Party sites,

³This functionality comes standard with the search functionality of Apache Lucene which we employ for the production of our index.

Measure	Description
Max Score (naïve)	Find the score of the metaphor that best matches the full semantic signature
Max Score (source)	Find the score of the metaphor that best matches the source side of the semantic signature
Max Score (target)	Find the score of the metaphor that best matches the target side of the semantic signature
Max Score (joint)	Independently score the metaphors by the target side and by the source side. Find the metaphor with the highest product of the scores.
Joint Count	Independently score the metaphors by the target side and by the source side. Count the number of metaphors that receive a positive score for both.

Table 2: The five features used by our metaphoricity classifiers.

etc.), and political commentary in web-zines and on-line newspapers. Three annotators have analyzed the raw texts and manually selected snippets of text (with context) whenever some element in the text seemed to have been used figuratively to describe or stand in for another element not represented in the text.⁴ Each of these metaphors is projected into a conceptual space using the process described in Section 3.2 and assembled into a searchable index.

For evaluation purposes, we have selected a subset of our overall repository which consists of 500 raw documents that have been inspected for metaphoricity by our annotators. We allocate 80% of these documents for the training of our classifiers and evaluate using the remaining 20%. In total, our training data consists of 400 documents containing 1,028 positive examples of metaphor and around 16,000 negative examples. Our test set consists of 100 documents containing 4,041 sentences with 241 positive examples of metaphor and 3,800 negative examples. For each sentence in each document, our system attempts to determine whether the sentence does or does not contain a metaphor within the domain of *Governance*.

We have experimented with several flavors of machine learning classification. In addition to an in-house implementation of a binary maximum entropy (MaxEnt) classifier, we have evaluated our results using four separate classifiers from the popular Weka machine learning toolkit.⁵ These include an unpruned decision tree classifier (J48), a support vector machine (SMO) approach using a quadratic

kernel with parameters tuned via grid search, a rule-based approach (JRIP), and a random forest classifier (RF). In addition, we have combined all five classifiers into an ensemble classifier which uses a uniformly-weighted voting methodology to arrive at a final decision.

5 Results

We have evaluated our methodology in two ways. First, we have performed an evaluation which highlights the discriminatory capabilities of our features by testing on a balanced subset of our test data. Next, we performed an evaluation which shows the utility of each of our classifiers as they are applied to real world data with a natural skew towards literal usages.⁶ In both cases, we train on a balanced subset of our training data using all 1,028 positive examples and a set of negative examples selected randomly such that each document under consideration contains the same number of positive and negative examples. In an initial experiment, we trained our classifiers on the full (skewed) training data, but the results suggested that an error-minimizing strategy would lead to all sentences being classified as “literal”.

As shown in Table 3, the choice of classifier appears significant. Several of the classifiers (J48, JRIP, and MaxEnt) maintain a high recall suggesting the ability of the tree- and rule-based classifiers to reliably “filter out” non-metaphors. On the other hand, other classifiers (SMO and ENSEMBLE) operate in a mode of high precision suggesting that a high confidence can be associated with their positive classifications. In all cases, performance is signifi-

⁴Generally speaking, each annotator operated within a region of high precision and low recall, and the overlap between individual annotators was low. As such, we have selected the union of all metaphors detected by the annotators.

⁵<http://www.cs.waikato.ac.nz/ml/weka/>

⁶Note that metaphors that are not related to the domain of *Governance* are classified as “literal”.

Classifier	Precision	Recall	F-Measure
J48	56.1%	93.0%	70.0%
JRIP	57.7%	79.3%	66.8%
MaxEnt	59.9%	72.6%	65.7%
ENSEMBLE	72.0%	42.7%	53.7%
RF	55.8%	47.7%	51.5%
SMO	75.0%	33.6%	46.4%
All metaphor	50.0%	100.0%	66.7%
Random baseline	50.0%	50.0%	50.0%

Table 3: The results of our experiments using several machine learning classifiers while evaluating on a dataset with 241 positive examples and 241 negative examples.

cantly better than chance as reported by our random baseline.⁷

Table 4 shows the result of evaluating the same models on an unbalanced dataset with a natural skew towards “literal” sentences which reflects a more realistic use case in the context of linguistic metaphor detection. The results suggest that, once again, the decision tree classification accepts the vast majority of all metaphors (93%), but also produces a significant number of false positives making it difficult to usefully employ this classifier as a complete metaphor detection system despite its top-performing F-measure on the balanced dataset. More useful is the SMO approach, which shows a precision over twice that of the random baseline. Put another way, a positive result from this classifier is more than 110% more likely to be correct than a random classification. From the standpoint of utility, joining these classifiers in an ensemble configuration seems to combine the high precision of the SMO classifier with the improved recall of the other classifiers making the ensemble configuration a viable choice in a real world scenario.

6 Conclusions

We have shown in this work the potential utility of our example-based approach to detect metaphor within a domain by comparing the semantic signature of a text with a set of known metaphors. Although this technique is necessarily limited by the coverage of the metaphors in the index, we believe that it is a viable technique for metaphor detection

⁷According to Fisher’s exact test (one-tailed): RF ($p < 0.02$); all others ($p < 0.0001$).

Classifier	Precision	Recall	F-Measure
SMO	12.7%	33.6%	18.4%
ENSEMBLE	11.2%	42.7%	17.8%
MaxEnt	8.7%	72.6%	15.6%
JRIP	8.1%	79.3%	14.8%
J48	7.6%	93.0%	14.0%
RF	7.4%	47.7%	12.7%
All metaphor	6.0%	100.0%	11.3%
Random baseline	6.0%	50.0%	10.7%

Table 4: The results of our experiments using several machine learning classifiers while evaluating on naturally skewed dataset with 241 positive examples and 3,800 negative examples.

as more and more examples become available. In future work, we hope to supplement our existing features with such information as term imageability, the transmission of affect, and selectional preference violation we believe will result in a robust system for linguistic metaphor detection to further aid in the computer understanding of natural language.

Acknowledgments

This research is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0025. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government. We would also like to thank our annotators whose efforts have made this work possible.

References

- K. Ahrens, S.F. Chung, and C. Huang. 2003. Conceptual metaphors: Ontology-based representation and corpora driven mapping principles. In *Proceedings of the ACL 2003 workshop on Lexicon and figurative language-Volume 14*, pages 36–42. Association for Computational Linguistics.
- C. Biemann. 2006. Chinese whispers: an efficient graph clustering algorithm and its application to natural lan-

- guage processing problems. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, pages 73–80. Association for Computational Linguistics.
- D. Bogdanova. 2010. A framework for figurative language detection based on sense differentiation. In *Proceedings of the ACL 2010 Student Research Workshop*, pages 67–72. Association for Computational Linguistics.
- D. Bracewell, M. Tomlinson, and M. Mohler. 2013. Determining the conceptual space of metaphoric expressions. In *Computational Linguistics and Intelligent Text Processing*, pages 487–500. Springer.
- C. Eilts and B. Lönneker. 2002. The Hamburg Metaphor Database.
- D. Fass. 1991. met*: A method for discriminating metonymy and metaphor by computer. *Computational Linguistics*, 17(1):49–90.
- C. Fellbaum. 1998. *WordNet, An Electronic Lexical Database*. The MIT Press.
- S. Harabagiu and F. Lacatusu. 2005. Topic themes for multi-document summarization. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 202–209. ACM.
- G. Hirst and D. St-Onge. 1998. Lexical chains as representations of context for the detection and correction of malapropism. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 305–332. MIT Press.
- A.N. Katz. 1992. Psychological studies in metaphor processing: extensions to the placement of terms in semantic space. *Poetics Today*, pages 607–632.
- W. Kintsch. 2000. Metaphor comprehension: A computational theory. *Psychonomic Bulletin & Review*, 7(2):257–266.
- G. Lakoff and M. Johnson. 1980. *Metaphors we live by*, volume 111. Chicago London.
- G. Lakoff. 1993. The contemporary theory of metaphor. *Metaphor and thought*, 2:202–251.
- G. Lakoff. 1994. *Master metaphor list*. University of California.
- T.K. Landauer and S.T. Dumais. 1997. A solution to Plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review; Psychological Review*, 104(2):211.
- L. Li and C. Sporleder. 2010. Using gaussian mixture models to detect figurative language in context. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 297–300. Association for Computational Linguistics.
- C. Lin and E. Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 495–501. Association for Computational Linguistics.
- B. Lönneker. 2003. Is there a way to represent metaphors in WordNets?: insights from the Hamburg Metaphor Database. In *Proceedings of the ACL 2003 workshop on Lexicon and figurative language-Volume 14*, pages 18–27. Association for Computational Linguistics.
- J.H. Martin. 1990. *A computational model of metaphor interpretation*. Academic Press Professional, Inc.
- Z.J. Mason. 2004. CorMet: A computational, corpus-based conventional metaphor extraction system. *Computational Linguistics*, 30(1):23–44.
- M.S. McGlone. 1996. Conceptual metaphors and figurative language interpretation: Food for thought? *Journal of memory and language*, 35(4):544–565.
- E. Niemann and I. Gurevych. 2011. The people’s web meets linguistic knowledge: Automatic sense alignment of Wikipedia and WordNet. In *Proceedings of the 9th International Conference on Computational Semantics (IWCS)*, pages 205–214. Citeseer.
- W. Peters and Y. Wilks. 2003. Data-driven detection of figurative language use in electronic language resources. *Metaphor and Symbol*, 18(3):161–173.
- E. Shutova, L. Sun, and A. Korhonen. 2010. Metaphor identification using verb and noun clustering. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1002–1010. Association for Computational Linguistics.
- E. Shutova. 2010. Models of metaphor in NLP. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 688–697. Association for Computational Linguistics.
- E.V. Shutova. 2011. *Computational approaches to figurative language*. Ph.D. thesis, University of Cambridge.
- S.L. Toral, M.R. Martínez-Torres, F. Barrero, and F. Cortés. 2009. An empirical study of the driving forces behind online communities. *Internet Research*, 19(4):378–392.
- R. Tourangeau and R.J. Sternberg. 1982. Understanding and appreciating metaphors. *Cognition*, 11(3):203–244.
- A. Utsumi. 2011. Computational exploration of metaphor comprehension processes using a semantic space model. *Cognitive science*, 35(2):251–296.
- Y. Wilks. 1978. Making preferences more active. *Artificial Intelligence*, 11(3):197–223.
- P. Wolff and D. Gentner. 2000. Evidence for role-neutral initial processing of metaphors. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 26(2):529.