Detecting novel metaphor using selectional preference information

Hessel Haagsma and **Johannes Bjerva**
University of Groningen
The Netherlands

Abstract

Recent work on metaphor processing often employs selectional preference information. We present a comparison of different approaches to the modelling of selectional preferences, based on various ways of generalizing over corpus frequencies. We evaluate on the VU Amsterdam Metaphor corpus, a broad corpus of metaphor. We find that using only selectional preference information is enough to outperform an all-metaphor baseline classification, but that generalization through prediction or clustering is not beneficial. A possible explanation for this lies in the nature of the evaluation data, and lack of power of selectional preference information on its own for non-novel metaphor detection. To better investigate the role of metaphor type in metaphor detection, we suggest a resource with annotation of novel metaphor should be created.

1 Introduction

Within natural language processing (NLP), there has been an increasing interest in the processing of figurative language in recent years. This increasing interest is exemplified by the (NA)ACL workshops on metaphor in NLP, and shared tasks like SemEval-2015, Task 11, about sentiment analysis of figurative language in Twitter.

The main benefits of improved treatment of figurative language within NLP lie with high-level tasks dependent on semantics, such as semantic parsing. For example, in multilingual semantic parsing, one would like to see both the English 'This textbook costs an arm and a leg.' and the Dutch 'Dit lesboek kost een rib uit het lijf.' (lit. 'This textbook costs a rib from the body.') to be mapped to the same meaning representation, which should not include references to any of the body parts used in the idiomatic expressions.

In this paper, we focus on one area of figurative language processing, namely the detection of novel metaphor, and especially the utility of selectional preference features for this task. By doing this, we hope to answer a two-fold question: can selectional preference information be used to successfully detect novel metaphor? And how do generalization methods influence the effectiveness of selectional preference information?

2 Background

2.1 Types of metaphor in language

The first part of any work on metaphor processing is is to arrive at a definition for 'metaphor'.

A good starting point for this is the MIP framework (Pragglejaz Group, 2007), which has this as the most important criterion for metaphor: 'a lexical unit is metaphorical if it has a more basic contemporary meaning in other contexts than in the current context' (paraphrased). Where 'more basic' is defined as more concrete, related to bodily action, more precise, and historically older. This is a very broad definition, causing many meanings to be classified as metaphorical.

Examples of this are found in the VU Amsterdam Metaphor corpus, which is annotated using the MIPVU procedure (Steen et al., 2010), an extension of MIP. In the corpus, we find a wide range of

metaphor from highly conventionalized metaphor, such as *have* in Example 1, to more clear metaphorical cases, like *rolling* in Example 2.

(1) Do the Greeks **have** a word for it?

(2) [. . .] the hillsides ceased their upward **rolling** and curved together [. . .]

Three general categories of meaning can be distinguished. The first type of contextual meaning is literal meaning, which is the most basic meaning of a word. This kind of meaning is generally the least problematic, since the meaning can be deduced from the lexicon. The second type of meaning is conventional metaphorical meaning, i.e. a non-basic meaning of a word, which is also in the lexicon or sense inventory.

Novel metaphors form a third type of contextual meaning, one which is non-basic, but is not accounted for by the lexicon either. These are the most problematic, since their meaning cannot be deduced by selecting the correct sense from a sense inventory.

Aligned with the distinction between conventional metaphor and novel metaphor is the distinction between word sense disambiguation and metaphor detection. That is, word sense disambiguation can be expected to cover literal and conventionalized metaphorical meanings, whereas metaphor processing is necessary to deal with novel metaphorical meanings.

We argue here that the most important application of metaphor processing approaches lies not with conventional metaphors, but with novel metaphors. This type of metaphor is the most problematic for other NLP applications, since it requires a deduction of meaning from outside the sense lexicon. Therefore, it requires a dedicated metaphor processing system, whereas more conventional metaphor can, in principle, be handled by existing WSD systems.

This is not to say that metaphor processing research should exclusively focus on novel metaphor, since insights about metaphor and metaphoricity can definitely be useful for improving the handling of conventional metaphor within a word sense disambiguation framework. Nevertheless, the approach proposed in this paper aims only to deal with novel metaphor. In addition, separating the treatment of novel and conventional metaphor avoids the possible pitfall of trying to solve two essentially different problems using the same method.

2.2 Selectional preference violation as a feature for metaphor detection

Because, by definition, we cannot use a pre-existing sense or meaning definition for novel metaphor, novel metaphor processing provides a new kind of challenge. On the other hand, with the ultimate goal of integrating figurative language processing in a semantic parsing framework, novel metaphor is the most problematic for deducing meaning, and is thus the most relevant and interesting.

Another important aspect of novel metaphors is that they tend to be less frequent than conventional metaphor (and literals). This has the advantage of making one feature commonly used for metaphor detection, selectional preference violation, more effective.

The idea of using selectional preference violation as an indicator of metaphoricity goes far back (Wilks, 1975; Wilks, 1978), and has been widely used in previous work on metaphor detection (Fass, 1991; Mason, 2004; Jia and Yu, 2008; Shutova et al., 2010; Neuman et al., 2013; Wilks et al., 2013).

Using selectional preference violation as a heuristic works well, but has the fundamental shortcoming that selectional preferences are based on frequency; selectional preference data are obtained from a corpus by counting how often words occur together in a certain relation. Metaphor, on the other hand, is defined by basicness of meaning, and not frequency of meaning. Although these two are correlated, there are also many cases where the figurative sense of a word has become more frequent than its original, literal sense.

The result of this is that metaphor detection systems based on selectional preferences erroneously classify low-frequent literals as metaphorical and high-frequent metaphors as literal. However, if we intend only to capture novel metaphors, the selectional preference violation approach is less vulnerable to these errors, since novel metaphors are less likely to be mistakenly classified as literal due to their lower frequency.

As such, we hypothesize that this feature, when used correctly, can be very useful for metaphor detection. To this purpose, we examine new ways of

generalizing over selectional preferences obtained from a corpus.

3 Methods

Selectional preference information can be automatically gathered from a large, parsed corpus. This allows for counting the occurrences of specific pairs, such as 'evoke-excitement'. Based on this, we can then calculate conditional probabilities or a measure like selectional association (Resnik, 1993).

However, even a very large corpus will not yield full coverage of all possible pairs. This could cause the system to flag, for example 'evoke-exhilaration' as a selectional preference violation and thus as metaphoric, simply because this combination occurs very infrequently in the corpus.

The solution to this problem is to do some sort of generalization over these verb-noun pairs. One way of doing this is to perform clustering the verbs and nouns in semantically coherent classes. We test two clustering methods: Brown clustering (Brown et al., 1992), and a novel method, which relies on k-means clustering of word embeddings.

In addition to these clustering methods, we also look at using word embeddings directly in a classifier, in order to prevent the information loss inherent in a clustering approach. These approaches are compared to performance based on using the selectional preference information without any generalization and a most-frequent class baseline.

3.1 Corpus frequencies

Before we can apply generalization methods over selectional preferences, we need to gather selectional preference frequencies from a large corpus of English. We use a recent dump (13-01-2016)[1] of the English part of Wikipedia. The corpus was preprocessed in order to extract only the raw text using WikiExtractor[2], resulting in a raw text corpus of approximately 1.6 billion words.

We use Wikipedia because of its size, which should help reduce sparsity for word pair frequencies, its range of topics, which increases the coverage in terms of word types. The downside of using Wikipedia is that it contains only one genre, namely scientific or encyclopaedia-style text. In contrast, the data used for evaluation is taken from different parts of the BNC, which represent different genres. Nevertheless, we assume that the larger size and number of topics of the Wikipedia corpus compensate for this.

As a dependency parser, we use the spacy.io[3] Python library, because of its speed, high accuracy and ease-of-use. From the parsed corpus, all verb-noun pairs with the labels `nsubj` and `dobj` were extracted and counted. This yielded 101 million verb-subject triples of 14.0 million distinct types and 67 million verb-object triples of 7.8 million distinct types. The triples contained 175,630 verb lemma types and 1,982,512 noun lemma types.

3.2 Selectional preference metrics

Selectional preference information can be used in three ways: as a conditional probability, a selectional preference strength for the verb, and as selectional association between a verb and a noun. The conditional probability is defined as follows:

$$P(n|v) = \frac{P(n,v)}{P(v)} \approx \frac{c(n,v)}{c(v)} \quad (1)$$

Where $P(n|v)$ is the probability of seeing noun n occurring with verb v. This can be approximated by using the counts c from the corpus.

For the other two metrics, we follow Shutova (2010) in using the metrics proposed by Resnik (1993), selectional preference strength (SPS) and selectional association (SA). Selectional preference strength is the information gain of a verb. This is an indicator of how selective the verb is in the choice of its arguments, and could thus be useful in filtering out weak-preference verbs.

$$SPS(v) = \sum_{n \in N} P(n|v) * \log \frac{P(n|v)}{P(n)} \quad (2)$$

More directly useful is the selectional association (SA) measure, which is an indication of how typical a noun is in its occurrence with a certain verb. It is defined as follows:

$$SA(v,n) = \frac{1}{SPS(v)} * P(n|v) * \log \frac{P(n|v)}{P(n)} \quad (3)$$

[1] https://dumps.wikimedia.org/enwiki/20160113/

[2] http://medialab.di.unipi.it/wiki/Wikipedia_Extractor

[3] https://spacy.io

3.3 Clustering

For the Brown clustering, we use the pre-trained clusters from Derczynski et al. (2015). They provide clusterings[4] with different numbers of clusters, which makes it easy to find an optimal number of clusters. We use the clusters trained on 64M words of newswire text.

For k-means clustering of word embeddings, we use the GloVe word embeddings (Pennington et al., 2014), with 300 dimensions, trained on a 840B word corpus and cluster the embeddings of the top 400,000 most frequent words. Clustering is done using the k-means clustering implementation from scikit-learn (Pedregosa et al., 2011), with k-means++ initialization, a maximum of 500 iterations, and the best clustering out of 10 runs with random initialization in terms of inertia.

3.4 Word embeddings-based probability predictions

In the word embeddings-based learner setting, rather than doing generalization through clustering, we use the word embeddings for generalization directly. This is done by modelling the conditional probabilities directly, based on the word embeddings of the noun and verb involved. We expect this to benefit generalization, since it avoids the information loss inherent in clustering when dealing with distributed representations – clustering (and thus discretizing) such representations effectively removes any notion of similarity between representations.

We use a neural network implemented in Theano (Bergstra et al., 2010; Bastien et al., 2012) using Keras. As input, we take the concatenation of the distributed representations of each verb and its subject/object. The output of the network is a continuous variable, namely the log probability of the construction at hand. We use a single hidden layer containing 600 hidden units, with a sigmoid activation function. Optimisation is done using the ADAM optimiser (Kingma and Ba, 2014). The loss function used is the mean squared error. Dropout is applied as regularisation after the hidden layer ($p = 0.5$) (Srivastava et al., 2014). We use a batch size of 2,000 and train over the course of 200 epochs. The resulting predictions are used as the Predicted Log-Probability (P-LP) feature. Metaphor detection results using this feature are presented in Table 4.

3.5 Evaluation

We evaluate the approaches on the VU Amsterdam Metaphor corpus (VUAMC)[5]. The corpus is preprocessed by extracting all raw text, and marking each word with the attribute `function="mrw"`, which indicates a metaphor-related word, as metaphor.

This results in a corpus of 40,622 verbs, of which 23,069 have at least one subject or object relation. We split the data three-way: verbs with only a subject (13,466 pairs), verbs with only an object (3,913 pairs), and verbs with both a subject and an object (5,539 triples).

We use this corpus since it is the largest corpus with metaphor annotations available. The downside is that it includes a large number of very conventionalized metaphors, and there is no annotation of novel metaphor or a degree of metaphoricity.

Evaluation is performed using a logistic regression classifier, which takes one or more of the selectional preference-based features. The features used are: conditional probability, log probability, selectional preference strength and selectional association, for subject-verb and object-verb pairs, and based on corpus frequencies, predictions or clusterings. We use only those features which are available, i.e. for the subject dataset we only use the four features for subject-verb pairs, while in the subject-and-object dataset we use eight features, four based on the subject-verb pair and four based on the object-verb pair.

Model fitting and evaluation is done using 10-fold cross validation, and l2-regularization with strength 1 is applied. In case of missing data points (e.g. no cluster available for the verb/noun), the majority class (non-metaphor) is assigned. As a baseline, we calculate the score when classifying all items as metaphor.

In addition, we evaluate the effect of re-weighting examples. Beigman Klebanov et al. (2015), who also evaluated their metaphor detection system on the VUAMC, showed that re-weighting training examples can have a large impact on performance, due to the large class imbalance in the VUAMC (mostly

[4] http://derczynski.com/sheffield/brown-tuning

[5] http://www.vismet.org/metcor/documentation/home.html

Data	BL	CP	LP	P-LP	SPS	SA	All
Subject	**23.0**	0.0	0.0	0.0	0.0	0.0	1.3
Object	**50.8**	0.0	3.2	1.4	0.0	0.0	2.4
Both	**53.4**	0.0	18.1	0.7	0.0	2.3	32.1

Table 1: F1-scores on metaphor identification using selectional preference information without any generalization, with varying features. Features: all-metaphor BaseLine (BL), Conditional Probability (CP), Log-Probability (LP), word embeddings-based Predicted Log-Probability (P-LP), Selectional Preference Strength (SPS), Selectional Association (SA), combination of CP, LP, SPS and SA (All).

Data	BL	CP	LP	P-LP	SPS	SA	All
Subject	23.0	24.5	24.5	23.2	20.9	26.4	**33.6**
Object	50.8	**53.4**	45.6	49.2	49.0	51.2	47.6
Both	53.4	54.2	44.3	50.0	50.5	53.8	**57.8**

Table 2: F1-scores on metaphor identification using selectional preference information without any generalization, with varying features. Features: all-metaphor BaseLine (BL), Conditional Probability (CP), Log-Probability (LP), word embeddings-based Predicted Log-Probability (P-LP), Selectional Preference Strength (SPS), Selectional Association (SA), combination of CP, LP, SPS and SA (All). Re-weighting of training examples was applied.

non-metaphor). They find that re-weighting examples increases F1-score by sacrificing precision for a large increase in recall.

4 Results

In this section, we present the results of evaluating each of the approaches to modelling selectional preference information on the VUAMC, as described previously. Table 4 shows the results for metaphor detection without any form of generalization. Table 4 shows the same, but with re-weighting of examples.

The most striking difference here is the difference between the two sets of results. Without re-weighting, performance is poor. The classifier classifies almost everything as non-metaphorical for the subject and object datasets, and for the subject-object dataset, even the best feature set is still far below the baseline.

In contrast, we see that performance with re-weighting is a lot better for all datasets. For the subject and subject-object datasets, the setting with all features performs best, out-performing the baseline by 10.6% and 4.4% respectively. For the object dataset, however, performance is highest with only the most basic feature, the conditional probabilities, out-performing the baseline by 2.6%. Clearly, the re-weighting of of examples enables the classifier to find a model that actually detects metaphor, rather than defaulting to a majority-class baseline.

Tables 5 shows the results for Brown clustering, Table 5 does the same for k-means clustering, and the P-LP column in Table 4 shows the results for word-embeddings based prediction. Since performance using all features and re-weighting of examples worked best in the no-clustering setting, we only report the results using these settings.

Results show that all generalization approaches fail to improve over the non-generalization setting. The Brown clustering seems to work better than the k-means clustering, which in turn yields slightly higher results than the predicted probabilities. The differences between the datasets are not affected by generalization, performance, relative to the baseline, is highest for the subject-only data, whereas it is lowest on the object-only data.

For both the Brown and k-means clustering, cluster size seems to only have a minor effect on performance, and no consistent pattern emerges from the results. That is, there is no cluster size which performs consistently better across data sets and/or clustering methods.

5 Discussion

The most prominent conclusion we can draw from the results is that, in the current set-up, generalization across selectional preference information using clustering or prediction does not work. Although the generalization approaches sometimes improve over the baseline, they never out-perform the best results from the no-generalization setting.

The idea behind generalization is that we sacrifice some information (about specific words) for a gain in robustness of the feature values, especially for less frequent words and word combinations. Here, clearly, the benefits do not outweigh the disadvantages. Since we explored a large range of cluster

Data	BL	80	160	320	640	1280	2560	5120
Subj	23.0	26.3	**28.8**	27.9	25.9	26.3	26.6	25.3
Obj	**50.8**	48.7	47.7	45.3	46.9	44.7	44.6	46.2
Both	53.4	52.7	52.8	53.7	54.3	53.5	54.3	**54.5**

Table 3: F1-scores on metaphor identification using Brown clustering for generalization, for varying numbers of clusters, using all available features. Re-weighting of training examples was applied.

Data	BL	80	160	320	640	1280	2560	5120
Subj	23.0	24.2	23.5	**30.7**	28.6	24.4	23.6	22.9
Obj	**50.8**	40.4	44.8	45.8	44.2	48.9	48.8	49.8
Both	**53.4**	49.8	48.2	50.4	49.2	47.6	50.4	49.5

Table 4: F1-scores on metaphor identification using word embedding-based k-means clustering for generalization, for varying numbers of clusters, using all available features. Re-weighting of training examples was applied.

sizes, which did not have a clear effect on detection performance, the root cause for this is likely to be the semantic coherence of the clusters. That is, either the clusters are not semantically coherent, or they are not coherent in such a way that they form a useful domain for metaphor detection.

As for the different datasets, we see a large performance difference in Table 4, where the model almost always defaults to a majority class classification in the object- and subject-only cases. Re-weighting the training data, however, removes this effect completely. Even for the subject-only data, which has the most skewed distribution (only 13% metaphor) the re-weighting model works well.

Looking at instances classified as metaphor by the best-performing system (57.8% F-score, Table 4) reveals three main things about the system performance and the nature of the evaluation data.

First, all datasets, but especially the subject- and object-only datasets contain a large proportion of pronoun subjects and objects, for which selectional preference information is not useful, unless pronoun resolution is applied. Since the metaphor annotation procedure does include resolving of referring expressions, this significantly hurts performance. An obvious improvement therefore would be to combine metaphor detection with coreference resolution.

Second, we find that a large part of the triples classified as metaphor contain a light verb, such as *have*, *make*, *take*, and *put*. Looking at the dataset as a whole reveals that 85-90% of the triples containing these verbs are classified as metaphor by the system. In the VUAMC, these verbs are often annotated as metaphor, since, due to their wide usage, they often occur in a sense that is not the most basic. It could be argued that these verbs, on the contrary, have such an eroded meaning that they are never metaphorical (Shutova and Teufel, 2010; Beigman Klebanov et al., 2014).

Third, we notice that the amount of novel metaphors in the VUAMC is minimal. Looking at instances correctly classified by the best-performing system yielded only one example of novel metaphor (Example 3). Here, the verb *escape* is used in a novel way; no matching sense is found in WordNet.

(3) [...] Adam might have **escaped** the file memories for years,

We hypothesized that selectional preference information can be used successfully for detection of novel metaphor, but the VUAMC contains all kinds of metaphor, including a large number of highly conventionalized metaphors. As such, even if the hypothesis is true, we cannot see this in the current evaluation setting. Unfortunately, this presents us from answering the research question posed in the introduction; it is still unclear whether selectional preference information is useful for detection of novel metaphor.

Due to the different kinds of metaphor in the corpus, the logistic regression model attempts to learn a model for all kinds of metaphor. This is not possible, and as such almost all instances get classified as the majority class, non-metaphorical. If novel metaphor could be successfully detected with the features used here, the regression learner might not produce a model that achieves this, since there is a large majority of conventional metaphor in the data.

In order to empirically evaluate how well these features work for novel metaphor, a corpus of novel metaphor, or a general corpus of metaphor with metaphoricity annotations is required. Considering that this is not available, an interesting course

for future work would be to create such a resource, and use that for the evaluation of systems for novel metaphor detection.

This could be done as a completely new resource, but also as an extension of existing resources' annotation. The VUAMC for example, could be extended by annotating whether metaphorically used words are used in a novel sense, i.e. an out-of-vocabulary sense. This would fit nicely with the MIPVU procedure, which is already strongly dependent on a fixed sense lexicon. An alternative would be to annotate metaphoricity or 'novelty of metaphor' as scalar, based on human judgements. This would also allow resources to distinguish more easily between types of metaphor.

Acknowledgements

We would like to thank the anonymous reviewers for their insightful and constructive comments, and Beata Beigman Klebanov for pointing out the utility of re-weighting the training data.

References

Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian J. Goodfellow, Arnaud Bergeron, Nicolas Bouchard, and Yoshua Bengio. 2012. Theano: new features and speed improvements. In *Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop*.

Beata Beigman Klebanov, Chee Wee Leong, Michael Heilman, and Michael Flor. 2014. Different texts, same metaphors: Unigrams and beyond. In *Proceedings of the Second Workshop on Metaphor in NLP*, pages 11–17.

Beata Beigman Klebanov, Chee Wee Leong, and Michael Flor. 2015. Supervised word-level metaphor detection: Experiments with concreteness and reweighting of examples. In *Proceedings of the Third Workshop on Metaphor in NLP*, pages 11–20.

James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. 2010. Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based *n*-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Leon Derczynski, Sean Chester, and Kenneth S. Bøgh. 2015. Tune your Brown clustering, please. In *Proceedings of RANLP 2015*, pages 126–133.

Dan Fass. 1991. met*: A method for discriminating metonymy and metaphor by computer. *Computational Linguistics*, 17(1):49–90.

Yuxiang Jia and Shiwen Yu. 2008. Unsupervised Chinese verb metaphor recognition based on selectional preferences. In *22nd Pacific Asia Conference on Language, Information and Computation*, pages 207–214.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Zachary J. Mason. 2004. CorMet: A computational, corpus-based conventional metaphor extraction system. *Computational Linguistics*, 30(1):23–44.

Yair Neuman, Dan Assaf, Yohai Cohen, Mark Last, Shlomo Argamon, Newton Howard, and Ophir Frieder. 2013. Metaphor identification in large texts corpora. *PLOS ONE*, 8(4).

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of EMNLP 2014*, pages 1532–1543.

Pragglejaz Group. 2007. MIP: A method for identifying metaphorically used words in discourse. *Discourse, Metaphor and Symbol*, 22(1):1–39.

Philip Stuart Resnik. 1993. *Selection and Information: A class-based approach to lexical relationships*. Ph.D. thesis, University of Pennsylvania.

Ekaterina Shutova and Simone Teufel. 2010. Metaphor corpus annotated for source-target domain mappings. In *Proceedings of LREC 2010*, pages 3255–3261.

Ekaterina Shutova, Lin Sun, and Anna Korhonen. 2010. Metaphor identification using verb and noun clustering. In *Proceedings of COLING 2010*, pages 1002–1010.

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Proceedings of ACL 2010*, pages 1029–1037.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Gerard J. Steen, Aletta G. Dorst, J. Berenike Herrmann, Anna Kaal, Tina Krennmayr, and Trijntje Pasma. 2010. *A method for linguistic Metaphor Identification: from MIP to MIPVU*. John Benjamins, Amsterdam.

Yorick Wilks, Lucian Galescu, James Allen, and Adam Dalton. 2013. Automatic metaphor detection using large-scale lexical resources and conventional metaphor extraction. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 36–44.

Yorick Wilks. 1975. A preferential, pattern-seeking, semantics for natural language inference. *Artificial Intelligence*, 6(1):53–74.

Yorick Wilks. 1978. Making preferences more active. *Artificial Intelligence*, 11(3):197–223.