Learning to Identify Metaphors from a Corpus of Proverbs

Gözde Özbal[†] and **Carlo Strapparava**[†] and **Serra Sinem Tekiroğlu**[†] and **Daniele Pighin**[‡]
[†]FBK-Irst, Trento, Italy, [‡]Google - Zürich, Switzerland
gozbalde@gmail.com, {strappa, tekiroglu}@fbk.eu, biondo@google.com

Abstract

In this paper, we experiment with a resource consisting of metaphorically annotated proverbs on the task of word-level metaphor recognition. We observe that existing feature sets do not perform well on this data. We design a novel set of features to better capture the peculiar nature of proverbs and we demonstrate that these new features are significantly more effective on the metaphorically dense proverb data.

1 Introduction

Recent years have seen a growing attention towards attempts to understand figurative language in text (Steen et al., 2010, Shutova and Teufel, 2010, Turney et al., 2011, Neuman et al., 2013, Klebanov et al., 2015). Recently, Özbal et al. (2016) published a resource consisting of 1,054 proverbs annotated with metaphors at the word and sentence level, making it possible for the first time to test existing models for metaphor detection on such data. More than in other genres, such as news, fiction and essays, in proverbs metaphors can resolve a significant amount of the figurative meaning (Faycel, 2012). The richness of proverbs in terms of metaphors is very fascinating from a linguistic and cultural point of view. Due to this richness, proverbs constitute a challenging benchmark for existing computational models of metaphoricity.

In this paper, we devise novel feature sets especially tailored to cope with the peculiarities of proverbs, which are generally short and figuratively rich. To the best of our knowledge, this is the first attempt to design a word-level metaphor recognizer specifically tailored to such metaphorically rich data. Even though some of the resources that we use (e.g., imageability and concreteness) have been used for this task before, we propose new ways of encoding this information, especially with respect to the density of the feature space and the way that the context of each word is modeled. On the proverb data, the novel features result in compact models that significantly outperform existing features designed for word-level metaphor detection in other genres (Klebanov et al., 2014), such as news and essays. By also testing the new features on these other genres, we show that their generalization power is not limited to proverbs.

2 Background

In this section we provide a brief overview of the efforts of the NLP community to build metaphor datasets and utilize them to develop computational techniques for metaphor processing. Steen et al. (2010) construct the Amsterdam Metaphor Corpus (VUAMC) by annotating a subset of BNC Baby[1]. Linguistic metaphors in VUAMC are annotated by utilizing the Metaphor Annotation Procedure (MIP) proposed by Group (2007). VUAMC contains 200,000 words in sentences sampled from various genres (news, fiction, academic, and conversations) and 13.6% of the words are annotated as metaphoric (Shutova, 2010). Another metaphor annotation study following the MIP procedure is conducted by Shutova and Teufel (2010). A subset of

[1] http://www.natcorp.ox.ac.uk/corpus/babyinfo.html

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 2060–2065,
Austin, Texas, November 1-5, 2016.

the British National Corpus (BNC) (Burnard, 2000) is annotated to reveal word-level verb metaphors and to determine the conceptual mappings of the metaphorical verbs.

Turney et al. (2011) introduce an algorithm to classify word-level metaphors expressed by an adjective or a verb based on their concreteness levels in association with the nouns they collocate. Similarly, Neuman et al. (2013) extend the concreteness model with a selectional preference approach to detect metaphors formed of concrete concepts. They focus on three types of metaphors: i) IS-A, ii) verb-noun, iii) adjective-noun. Rather than restricting the identification task to a particular POS or metaphoric structure, Hovy et al. (2013) aim to recognize any word-level metaphors given an unrestricted text, and they create a corpus containing sentences where one target token for each sentence is annotated as metaphorical or literal. They use SVM and CRF models with dependency tree-kernels to capture the anomalies in semantic patterns. Klebanov et al. (2014) propose a supervised approach to predict the metaphoricity of all content words in a running text. Their model combines unigram, topic model, POS and concreteness features and it is evaluated on VUAMC and a set of essays written for a large-scale assessment of college graduates. Following this study, Klebanov et al. (2015) improve their model by re-weighting the training examples and redesigning the concreteness features.

The experiments in this paper are carried out on PROMETHEUS (Özbal et al., 2016), a dataset consisting of 1,054 English proverbs and their equivalents in Italian. Proverbs are annotated with word-level metaphors, overall metaphoricity, meaning and century of first appearance. For our experiments, we only use the word-level annotations on the English data.

3 Word-level metaphor detection

Similarly to Klebanov et al. (2014), we classify each content word (i.e., adjective, noun, verb or adverb) appearing in a proverb as being used metaphorically or not. Out of 1,054 proverbs in PROMETHEUS, we randomly sample 800 for training, 127 for development and 127 for testing. We carry out the development of new features on the development set; then we compare the performance of different feature sets using 10-fold cross validation on the combination of the development and training data. Finally, we test the most meaningful configurations on the held-out test data. As a baseline, we use a set of features very similar to the one proposed by Klebanov et al. (2014). To obtain results more easily comparable with Klebanov et al. (2014), we use the same classifier, i.e., logistic regression, in the implementation bundled with the *scikit-learn* package (Pedregosa et al., 2011). For all the experiments, we adjust the weight of the examples proportionally to the inverse of the class frequency.

3.1 Baseline features (B)

Unigrams (u_B): Klebanov et al. (2014) use all content word forms as features without stemming or lemmatization. To reduce sparsity, we consider lemmas along with their POS tag.

Part-of-speech (p_B): The coarse-grained part-of-speech (i.e., noun, adjective, verb or adverb) of content words[2].

Concreteness (c_B): We extract the concreteness features from the resource compiled by Brysbaert et al. (2014). Similarly to Klebanov et al. (2014), the mean concreteness ratings, ranging from 1 to 5, are binned in 0.25 increments. We also add a binary feature which encodes the information about whether the lemma is found in the resource.

Topic models (t_B): We use Latent Dirichlet Allocation (LDA) (Blei et al., 2003) using Gibbs sampling for parameter estimation and inference (Griffiths, 2002). We run LDA on the full British National Corpus (Consortium and others, 2001) to estimate 100 topics, using 2000 Gibbs sampling iterations, and keeping the first 1000 words for each topic. As topic model features for a lemma, we use the conditional probability of the topic given the lemma for each of the 100 topics generated by LDA. Besides, we use a binary feature that encodes whether the lemma exists in the LDA model.

3.2 Novel features (N)

We introduce five feature sets that capture other aspects of the data which we consider to be meaningful for the peculiar characteristics of proverbs.

[2]Klebanov et al. (2014) consider the Penn Treebank tagset generated by Stanford POS tagger.

Imageability (i) and Concreteness (c): Imageability and concreteness of the metaphor constituents were found to be highly effective in metaphor identification by several studies in the literature (Turney et al., 2011, Broadwell et al., 2013, Neuman et al., 2013, Tsvetkov et al., 2014). We obtain the imageability and concreteness scores of each lemma from the resource constructed by Tsvetkov et al. (2014), as it accounts for both dimensions. The imageability (concreteness) feature set contains the following four features:

- ***Has score*:** A binary feature that indicates whether the lemma exists in the relevant resource.
- ***Score value*:** The imageability (concreteness) score of the lemma.
- ***Average sentence score*:** The average imageability (concreteness) score of the other lemmas in the sentence.
- ***Score difference*:** The difference between *Average sentence score* and *Score value*.

The last two features take the context of the target lemma into account and encode the intuition that metaphorical lemmas often have higher imageability (concreteness) than the rest of the sentence (Broadwell et al., 2013).

Metaphor counts (m): This feature set consists of three features. The first two features encode the number of times a lemma-POS pair is used as a metaphor and a non-metaphor in the data. The third feature evaluates to the difference between these counts[3].

Standard domains (d_s) and normalized domains (d_n): These features reflect our intuition that there is a strong prior for some domains to be used as a source for metaphors. This notion is backed by the analysis of PROMETHEUS carried out by Özbal et al. (2016). We also expect that words which are clearly out of context with respect to the rest of the sentence are more likely to be used as metaphors. The correlation between word and sentence domains described below aims to model such phenomenon. For each lemma-POS pair, we collect the domain information from WordNet Domains[4] (Magnini et al., 2002, Bentivogli et al., 2004) for the standard domains feature set, which consists of 167 features (1 real valued, 166 binary). It includes a binary indicator set to 1 if the lemma is found in WordNet Domains. A domain vector consisting of 164 binary indicators mark the domains to which the lemma belongs. Then, we compute a sentence domain vector by summing the vectors for all the other lemmas in the sentence, and we encode the Pearson correlation coefficient between the two vectors (lemma and sentence) as a real valued feature. Finally, a binary feature accounts for the cases in which no other lemma in the sentence has associated domain information.

The same process is repeated for the normalized domains. For normalization, we use a reduced set of domains (43 distinct domains) by considering the middle level of the WordNet Domains hierarchy. For instance, VOLLEY or BASKETBALL domains are mapped to the SPORT domain. Normalization already proved to be beneficial in tasks such as word sense disambiguation (Gliozzo et al., 2004). It allows for a good level of abstraction without losing relevant information and it helps to overcome data sparsity. The set of normalized domain features (d_n) consists of 46 features (45 binary, 1 real valued).

Dense signals (s): This set includes three binary features which summarize the concreteness, imageability and metaphor count feature sets. The first (second) feature is set to 1 if the imageability (concreteness) of the lemma is higher than the average

Feature sets	*C*	*P*	*R*	*F*
$B^{\#}$	0.9	0.666	0.832	0.738
N^{*}	0.6	0.785	0.884	0.833
$B \cup N^{*}$	0.6	0.798	0.875	0.834
$N \setminus i^{*}$	0.6	0.788	0.886	0.833
$N \setminus c^{*}$	0.6	0.782	0.888	0.831
$N \setminus m^{*\#}$	0.6	0.780	0.824	0.799
$N \setminus d_s^{*}$	1.0	0.787	0.842	0.815
$N \setminus d_n^{*}$	1.0	0.789	0.884	0.832
$N \setminus (d_s \cup d_n)^{\#}$	1.0	0.746	0.704	0.724
$N \setminus s^{*}$	1.0	0.776	0.909	0.836
$(N \setminus (d_s \cup d_n)) \cup t_B^{\#}$	0.6	0.751	0.705	0.724

Table 1: Cross-validation performance on the proverb training and development data. The meta-parameter C is the inverse of the regularization strength. *: significantly different from B with $p < .001$; $^{\#}$: s.d. from N with $p < .001$.

[3] Counts are estimated on training folds. To reduce overfitting, lemmas are randomly sampled with a probability of 2/3.

[4] We always select the first sense of the lemma-POS.

Features	*P*	*R*	*F*
$B^{\#}$	0.75	0.70	0.73
N^{*}	0.86	0.83	0.85
$N \setminus s^{*}$	0.82	0.87	0.85
$B \cup N^{*}$	0.87	0.85	0.86

Table 2: Performance on the proverb test data. *: significantly different from B with $p < .001$. $^{\#}$: significantly different from N with $p < .001$.

Genre	*Features*	*C*	*P*	*R*	*F*
News	B	1.0	0.475	0.742	0.576
	N	1.0	0.576	0.479	0.522
	$B \cup N$	1.0	0.615	0.539	0.574
Academic	B	0.6	0.489	0.733	0.568
	N	0.6	0.572	0.494	0.511
	$B \cup N$	1.0	0.539	0.648	0.569
Conversation	B	0.6	0.292	0.799	0.416
	N	0.6	0.304	0.626	0.393
	$B \cup N$	1.0	0.299	0.731	0.406
Fiction	B	0.6	0.349	0.695	0.460
	N	0.6	0.430	0.418	0.421
	$B \cup N$	0.6	0.409	0.551	0.465

Table 3: Cross-validation performance on VUAMC. B is always significantly different from N ($p < .001$), and $B \cup N$ is always significantly different from both B and N ($p < .001$).

imageability (concreteness) of the rest of the sentence. The third feature is set to 1 if the lemma was observed more frequently as a metaphor than not, as estimated on training data.

3.3 Results

Table 1 shows the results of the 10-fold cross validation on the English proverb data. The value reported in the column labeled C is the optimal inverse of regularization strength, determined via grid-search in the interval $[0.1, 1.0]$ with a step of 0.1. Using only baseline features (B) we measure an average F1 score of 0.738. The performance goes up to 0.833 when the novel features are used in isolation (N) (statistically significant with $p < 0.001$). We believe that the difference in performance is at least in part due to the sparser B features requiring more data to be able to generalize. But most importantly, unlike B, N accounts for the context and the peculiarity of the target word with respect to the rest of the sentence. The combination of the two feature sets ($B \cup N$) very slightly improves over N (0.834), but the difference is not significant. The second block of rows in Table 1 presents a summary of the ablation tests that we conducted to assess the contribution of the different feature groups. Each lowercase letter indicates one of the feature sets introduced in the previous section. All configurations reported, except $N \setminus (d_s \cup d_n)$, significantly outperform B. In two cases, $N \setminus m$ and $N \setminus (d_s \cup d_n)$, there is a significant loss of performance with respect to N. The worst performance is observed when all the domain features are removed (i.e., $N \setminus (d_s \cup d_n)$). These results suggest that the prior knowledge about the domain of a word and the frequency of its metaphorical use are indeed strong predictors of a word metaphoricity in context. The fact that $N \setminus d_n$ and $N \setminus d_s$ do not result in the same loss of performance as $N \setminus (d_s \cup d_n)$ indicates that both d_n and d_s are adequately expressive to model the figuratively rich proverb data. In one case (i.e., $N \setminus s$), the F1 measure is slightly higher than N, even though the difference does not appear to be statistically significant. Our intuition is that each of the three binary indicators is a very good predictor of metaphoricity *per se*, and due to the relatively small size of the data the classifier may tend to over-fit on these features. As another configuration, the last row shows the results obtained by replacing our domain features d_s and d_n with the topic features t from B. With this experiment, we aim to understand the extent to which the two features are interchangeable. The results are significantly worse than N, which is a further confirmation of the suitability of the domain features to model the proverbs dataset.

We then evaluated the best configuration from the cross-fold validation ($N \setminus s$) and the three feature sets B, N and $B \cup N$ on the held-out test data. The results of this experiment reported in Table 2 are similar to the cross-fold evaluation, and in this case the contribution of N features is even more accentuated. Indeed, the absolute F_1 of N and $B \cup N$ is slightly higher on test data, while the f-measure of B decreases slightly. This might be explained by the low-dimensionality of N, which makes it less prone to overfitting the training data. On test data, $N \setminus s$ is not found to outperform N. Interestingly, $N \setminus s$ is the only configuration having higher recall than precision. As shown by the feature ablation experi-

ments, one of the main reasons for the performance difference between N and B is the ability of the former to model domain information. This finding can be further confirmed by inspecting the cases where B misclassifies metaphors that are correctly detected by N. Among these, we can find several examples including words that belong to domains often used as a metaphor source, such as "grist" (domain: "gastronomy") in "All is grist that comes to the mill", or "horse" (domain: "animals") in "You can take a horse to the water , but you can't make him drink".

Finally, Table 3 shows the effect of the different feature sets on VUAMC used by Klebanov et al. (2014). We use the same 12-fold data split as Klebanov et al. (2014), and also in this case we perform a grid-search to optimize the meta-parameter C of the logistic regression classifier. The best value of C identified for each genre and feature set is shown in the column labeled C. On this data, N features alone are significantly outperformed by B ($p < 0.01$). On the other hand, for the genres "academic" and "fiction", combining N and B features improves classification performance over B, and the difference is always statistically significant. Besides, the addition of N always leads to more balanced models, by compensating for the relatively lower precision of B. Due to the lack of a separate test set, as in the original setup by Klebanov et al. (2014), and to the high dimensionality of B's lexicalized features, we cannot rule out over-fitting as an explanation for the relatively good performance of B on this benchmark. It should also be noted that the results reported in (Klebanov et al., 2014) are not the same, due to the mentioned differences in the implementation of the features and possibly other differences in the experimental setup (e.g., data filtering, pre-processing and meta-parameter optimization). In particular, our implementation of the B features performs better than reported by Klebanov et al. (2014) on all four genres, namely: 0.52 vs. 0.51 for "news", 0.51 vs. 0.28 for "academic", 0.39 vs. 0.28 for "conversation" and 0.42 vs. 0.33 for "fiction".

Even though the evidence is not conclusive, these results suggest that the insights derived from the analysis of PROMETHEUS and captured by the feature set N can also be applied to model word-level metaphor detection across very different genres. In particular, we believe that our initial attempt to encode context and domain information for metaphor detection deserves further investigation.

4 Conclusion

We designed a novel set of features inspired by the analysis of PROMETHEUS, and used it to train and test models for word-level metaphor detection. The comparison against a strong set of baseline features demonstrates the effectiveness of the novel features at capturing the metaphoricity of words for proverbs. In addition, the novel features show a positive contribution for metaphor detection on "fiction" and "academic" genres. The experimental results also highlight the peculiarities of PROMETHEUS, which stands out as an especially dense, metaphorically rich resource for the investigation of the linguistic and computational aspects of figurative language.

References

Luisa Bentivogli, Pamela Forner, Bernardo Magnini, and Emanuele Pianta. 2004. Revising the Wordnet Domains Hierarchy: Semantics, Coverage and Balancing. In *Proceedings of the Workshop on Multilingual Linguistic Ressources*, pages 101–108. Association for Computational Linguistics.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, 3:993–1022.

George Aaron Broadwell, Umit Boz, Ignacio Cases, Tomek Strzalkowski, Laurie Feldman, Sarah Taylor, Samira Shaikh, Ting Liu, Kit Cho, and Nick Webb. 2013. Using Imageability and Topic Chaining to Locate Metaphors in Linguistic Corpora. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 102–110. Springer.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness Ratings for 40 Thousand Generally Known English Word Lemmas. *Behavior Research Methods*, 46(3):904–911.

Lou Burnard. 2000. Reference guide for the British National Corpus (World Edition).

BNC Consortium et al. 2001. The British National Corpus, version 2 (BNC World). *Distributed by Oxford University Computing Services*.

Dahklaoui Faycel. 2012. *Food Metaphors in Tunisian Arabic Proverbs*. Rice Working Papers in Linguistics 3/1.

Alfio Gliozzo, Carlo Strapparava, and Ido Dagan. 2004. Unsupervised and Supervised Exploitation of Semantic Domains in Lexical Disambiguation. *Computer Speech & Language*, 18(3):275–299.

Tom Griffiths. 2002. Gibbs Sampling in the Generative Model of Latent Dirichlet Allocation.

Pragglejaz Group. 2007. MIP: A Method for Identifying Metaphorically Used Words in Discourse. *Metaphor and Symbol*, 22(1):1–39.

Dirk Hovy, Shashank Srivastava, Sujay Kumar Jauhar, Mrinmaya Sachan, Kartik Goyal, Huiying Li, Whitney Sanders, and Eduard Hovy. 2013. Identifying Metaphorical Word Use with Tree Kernels. *Meta4NLP 2013*, page 52.

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. *NAACL HLT 2015*, page 11.

Bernardo Magnini, Carlo Strapparava, Giovanni Pezzulo, and Alfio Gliozzo. 2002. The Role of Domain Information in Word Sense Disambiguation. *Natural Language Engineering*, 8(04):359–373.

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):e62343.

Gözde Özbal, Carlo Strapparava, and Serra Sinem Tekiroğlu. 2016. PROMETHEUS: A Corpus of Proverbs Annotated with Metaphors. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

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.

Ekaterina Shutova and Simone Teufel. 2010. Metaphor Corpus Annotated for Source-Target Domain Mappings. In *LREC*, volume 2, pages 2–2.

Ekaterina Shutova. 2010. Automatic Metaphor Interpretation as a Paraphrasing Task. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1029–1037. Association for Computational Linguistics.

Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna A Kaal, and Tina Krennmayr. 2010. Metaphor in Usage. *Cognitive Linguistics*, 21(4):765–796.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor Detection with Cross-Lingual Model Transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 248–258. Association for Computational Linguistics.

Peter D Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and Metaphorical Sense Identification through Concrete and Abstract Context. In *Proceedings of the 2011 Conference on the Empirical Methods in Natural Language Processing*, pages 680–690.