Metaphor Detection in a Poetry Corpus by Applying Statistical Methods

Anonymized for blind reviewing

Abstract

Metaphor is indispensable in poetry. It demonstrates the creativity of the poet and also **assists in enhancing emotional and rhetorical devices**. Previous metaphor detection methods rely either on rule-based or statistical models, none of them applied to poetry. Our method, focusing on metaphor detection in a poetry corpus, combines rule-based and statistical models (Word Embeddings) to develop a **novel classification system**.

1 Introduction

Metaphors are crucial to the understanding of any literary text. A metaphor deviates from the normal linguistic flow and intends to create a strong statement that **no non-metaphoric text** can achieve. It is different than an idiom as it is possible to understand a metaphor even with no prior knowledge. Some examples of metaphors in poetry are:

- The hackles on my neck are fear
- My eyes are caves, chunks of etched rock

The early works on detecting metaphor in large text corpora were based on rules. Turney (2011) proposed the Concrete-Abstract rule: a concrete concept, when used to describe an abstract one, **represents** a metaphor. A phrase like "Sweet Dreams" is one such example. We use the Abstract-Concrete rule as one of the many features in our model. In experimentation, it has in fact proved to be quite useful in case of poetry as well.

Another of Turney's (2011) rules, Concrete Category Overlap (CCO), states that if both of the noun heads of a phrase are concrete, we can apply CCO instead of Concrete-Abstract. **SZP: a curcular definition** This is used as a feature in our model, but is applicable to a restricted set of cases.

Neuman (2013) proposes to categorize metaphor on the basis of POS tag sequences like Noun-Verb-Noun, Adjective-Noun, etc. We follow the same methodology to extract the set of sentences that may be possibly metaphorical in nature. **Our approach differs as we use WordEmbedding on Gigaword corpus to get word vector representations (vector difference and cosine similarity) of possible metaphorical word pairs.** Another difference is the addition of two more types of POS sequences that we encountered to be metaphorical in our Poetry Foundation poetry corpus.

Neuman's (2013) statistical model uses Mutual Information and selectional preference approach. He suggests using this through a large-scale corpus to find the most frequently occurring concrete nouns with a specific noun. Any noun outside this small set denotes a metaphor. Our experimentation does not directly involve finding selectional preference sets but instead we use Word Embeddings. **We find the selectional preference sets too limiting as word span is to be set before the experiments and some sentences exceed that limit, therefore the contextual meaning is lost.**

Shutova's (2016) statistical model detects metaphor just like ours. But her work involves more of a verb-centered approach which acts as a seed set for training data. Our work focuses more on noun-centered models and looks more to the application to poetry, not generically. **We focus more on noun-centered models, as we observed that poetry contains more noun-centered metaphors than verb-centered.**

Our current work is a subset of a larger project named "GraphPoem" Margento (2015) that involves the computational study of poetry and the

development of tools that can aid the academic study of poetry.

2 The Method

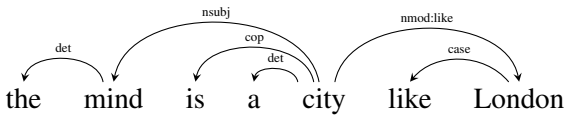
2.1 Building the Corpus

We have built our own corpus, because there is no publicly available corpus on poetry annotated for metaphors. Annotating poetry line by line can be laborious **as we have observed empirically that negative samples are too many.** To ease this task, we applied Neuman's approach: segregate metaphor on the basis of their POS tag sequence. We extracted all sentences from 12830 Poetry Foundation poems that match these tag sequence.

Type I metaphor are POS tag sequence of Noun-Verb-Noun where the verb is copular. We extended this to include the tag sequence Noun-Verb-Det-Noun, since we found many cases were being skipped due to the presence of a determiner. Type II are tag sequence of Noun-Verb-Noun with a regular or non-copula verb. Type III are tag sequence of Adjective-Noun.

In this paper, we focus on Type I metaphor. Other types would be detected in our future work. **We will also implement a tag-sequence-independent approach that employs a dependency parser to give all associations in a sentence. Using some of the associations like nsubj, dobj, etc., we will filter down to get word pairs that need to be checked for metaphor occurrence. All the other irrelevant associations will be discarded.**

Identifying head words in a sentence is in itself a very challenging task. In other words, it is like compressing a phrase to a word pair that may or may not be a metaphor. The POS tag sequence does not always provide a understandable word pair and critical words that may be of value, are lost. For these cases, the nouns highlighted by the POS tagger are not enough to identify the head of a sentence (or phrase). Therefore, we employ Stanford NLP Parser (Marneffe et al., 2006) to identify them. **We extract all nsubj associations from the sentence and if the head word is different from the earlier identified head, then it is updated.**



2.2 Annotating the Corpus

We extracted around 1500 sentences with type I metaphor tag sequence and annotated the first 700. To annotate, we employed majority voting. First, two independent annotators annotate the 700 sentences without any communication. Then Kappa is calculated. Its value came to around 0.39 and percent agreement to 66.79. Later, we involved a third annotator who decides a majority vote in case of disagreement. If one of the two annotators agrees to the other's justification then the disagreement finishes without the intervention of the third annotator.

While annotating, we encountered several sentences that were quite ambiguous and needed some more context to decide their inclination. In those specific cases, **SZP: such exceptions look suspect** annotators were allowed to go back to the poem in which they occurred and find the context. But in most cases, the sentence was enough to establish the metaphoric intent.

All sentences given to the annotators were marked to indicate where the head of the sentence lies, so that there is no confusion in case that there are more than one noun phrases. **For example:**

my eyes are caves , chunks of etched rock @2@

In the above example, the number 2 denotes that the word at location 2 i.e. "eyes" is a head word and therefore the second head would be "caves", as this is a Type 1 metaphor tagged sentence. Since this is obviously a metaphoric word pair, therefore the annotator will have to write "y" at the end of the sentence.

Further, the annotators were allowed to skip a sentence, in case of ambiguity or lack of context. Therefore, a sentence can be labeled as 'y' for metaphor, 'n' for non-metaphor and 's' for skipped sentence.

After annotating, we checked for the distribution of classes. Metaphor comprised 49.8%, non-metaphor 44.8% and skipped 5.4%. We have an almost balanced dataset, so we do not apply any sampling in our classification.

2.3 Rule-based Metaphor Detection

Firstly, we use Rule-based methods for our poetry dataset. We use the Abstract-Concrete and

Concrete Category Overlap rules established by Turney (2011). Abstract-Concrete rule needs the hypernym class of each noun. Therefore we use WordNet (1998). We get all hypernyms of head nouns and check for each parent till we reach the hypernym "abstract entity" or "physical entity". We use the first sense of WordNet as it is the most common usage **SZP: how do you know?** of that word.

Apart from the above rules, we also used a ConceptNet (Speer and Havasi, 2012) based feature. For each noun in our sentence, we extract the corresponding "SurfaceText" from ConceptNet. SurfaceText contain some associations between the specific word and real-world knowledge. For example, "car" gives the following associations:

- "drive" is related to "car"
- You are likely to find "a car" in "the city"

and so on.

The entities are already highlighted in the SurfaceTexts. We parse these associations and extract all the entities. There can be action associations as well:

- "a car" can "crash"
- "a car" can "slow down"

and so on.

These entities and actions are used to establish an overlap in the head nouns of our poetry sentences. We call this method ConceptNet Overlap. **We denote true if there is an overlap and false if not and is used as one of the features in our rule-based model.** **SZP: how can a method be a feature?!!**

2.4 Statistical-based Metaphor Detection

To capture the distortion of the context that a metaphor provides to a sentence, we use vector difference of the head words. The underlying idea is that the smaller the difference, the more connected the words would be. Conversely, a large difference implies disconnected words and hence a metaphor. We are capturing this difference in a 100-dimensional vector representation. We use difference of word vectors as the first statistical feature.

$$\begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ a_n \end{bmatrix} - \begin{bmatrix} b_1 \\ b_2 \\ . \\ . \\ b_n \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ . \\ . \\ c_n \end{bmatrix}$$

To get the word vectors of head words, we use a pre-trained Glove Gigaword corpus (Pennington et al., 2014). Earlier, we used a custom trained model based on British National Corpus (BNC, 2007) but switched to Glove to test on a bigger corpus. Another reason we tested on two different corpora was to remove any bias that may percolate due to the presence of common speech metaphors in the corpus.

We computed cosine similarity for all word vector pairs and included it as another feature for our model. We also use Pointwise Mutual Information of each word pair to capture the collocation information:

$$ln\frac{C(x,y).N}{C(x)C(y)},$$

where N is the size of corpus, C(x,y) is the frequency of x and y together, C(x) and C(y) is frequency of x and y in corpus respectively.

3 The Results

We tested on sentences that were extracted from 12830 Poetry Foundation poems and annotated manually. For training data, we used a combination of different datasets like Trofi(Birke and Sarkar, 2006) and Shutova's metaphor dataset (Mohammad et al., 2016) along with our own poetry dataset. We included other datasets with poetry to increase the training set and consequently to get better classification predictions.

Experiments	Precision	Recall	F-score
Rules (CA+CCO+CN)	0.615	0.507	0.555
PoFo poetry data	0.662	0.675	0.669
Trofi data	0.782	0.889	0.832
Shutova data	0.763	0.725	0.744
PoFo + Trofi + Shutova	0.741	0.822	0.779

Table 1: Results for class "metaphor"

Experiments	Precision	Recall	F-score
Rules (CA+CCO+CN)	0.462	0.408	0.433
PoFo poetry data	0.585	0.570	0.577
Trofi data	0.807	0.651	0.721
Shutova data	0.749	0.785	0.767
PoFo + Trofi + Shutova	0.731	0.627	0.675

Table 2: Results for class "non-metaphor"

Table 1 shows the class "metaphor" results. For rule based experiments, we included Concrete-Abstract, Conrete-Class-Overlap and ConceptNet

features. Training was done on 340 PoFo (Poetry Foundation) poetry sentences and testing on other set of 340. For PoFo data, the train and test was the same, but with word vector feature set instead of rules. For Trofi data, training and testing was done on 1771 instances each with the same feature set as PoFo. For Shutova data, training was done on 323 instances and testing on other 323. Lastly, all the above datasets are aggregated to test on 487 PoFo sentences. Training for this aggregated set was done on 3543 Trofi instances, 647 Shutova instances and 193 PoFo instances.

On analysis of results, it can be observed that overall best results are seen on Trofi data. But, on comparing PoFo results with the aggregate results, we can see that all the three metrics have drastically increased when the training data is large. Precision on isolated PoFo data is 0.662 whereas on aggregate data is 0.741. This also establishes that to detect metaphor in poetry, non-poetry data is as helpful as the poetry one.

It can be argued that the recall that we report is not the recall of metaphors in the whole poem but instead recall of the specific POS tag sequence that is extracted by our algorithm. There can be sentences that are metaphoric in nature, but are missed due to a different POS tag sequence. We totally agree to this viewpoint, and for the same reason, we are working on the type independent metaphor identification algorithm to handle those missing cases.

For data preprocessing, we do attribute selection by various algorithms like Pearson's, Infogain, Gain ratio, etc.. We report results only for the highest accuracy among these algorithms. For classification, we use several classifiers like RandomForest, SVM, NaiveBayes, JRip, etc.. We report only the highest accuracy achieved on these classifiers. For results in table 1, RandomForest classifier was used and for attribute selection Gain ratio evaluator.

Table 2 shows the results for class "non-metaphor". It is observed that though the precision of metaphor and non-metaphor classes are almost equal, recall of non-metaphor class is lower at 0.627 (it is 0.822 for class metaphor). On doing error analysis, it was seen that these "skipped" cases were mostly words that are archaic or poetic terms that do not have word vector representations. Still it is observed that statistical method scored better than the rule based methods for all metrics.

We also tested on 200 dimensional word vectors in order to investigate the impact of increasing the number of dimensions from 100 to 200 on accuracy metrics. Results showed that the accuracy dropped by 1% along with a slight decline in other metrics as well.

4 Conclusions and Future Work

Our preliminary results with Type 1 metaphor encourage us to work more and apply more methods in the future. We are already working on Type-independent metaphor identification to increase the recall of our analysis. For rule based methods, we may work on context overlap methods to remove the ambiguity between various senses that a word may possess and it may increase classification accuracy.

For statistical methods, there are many possibilities that we are looking into. Firstly, we are looking into applying phrase compositionality (Mikolov et al., 2013) to handle multiword expressions and phrases better. Since we are identifying metaphors in word pairs rather than the whole sentence, therefore the accuracy of vector representation for these words are very crucial. If the word pair extracted by the algorithm does not represent the whole phrasal meaning, then obviously classification would be incorrect at later stages. Secondly, we are looking into the application of deep learning classifiers like RNNs to further improve precision.

References

Turney P, Neuman Y, Assaf D, Cohen Y. 2011. Literal and metaphorical sense identification through concrete and abstract context. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK, July 2731: 680690.

Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006. Generating Typed Dependency Parses from Phrase Structure Parses. In LREC 2006.

Christiane Fellbaum. 1998. WordNet: An Electronic Lexical Database. W Cambridge, MA: MIT Press.

Neuman Y, Assaf D, Cohen Y, Last M, Argamon S, Howard N, et al. 2013. Metaphor Identification in Large Texts Corpora. PloS one, 8(4), e62343.

Robert Speer and Catherine Havasi. 2012. Representing General Relational Knowledge in ConceptNet 5. LREC 2012.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In Advances in Neural Information Processing Systems 26, pages 3111-3119.

Pennington Jeffrey, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014) 12.

Lou A., Inkpen D. and Tanasescu C. 2015. Multilabel Subject-Based Classification of Poetry. Nature, 2218, 30-7.

The British National Corpus 2007. The British National Corpus, version 3 (BNC XML Edition). Distributed by Oxford University Computing Services on behalf of the BNC Consortium.

Birke J. and Sarkar A. 2006. A Clustering Approach for Nearly Unsupervised Recognition of Nonliteral Language. In EACL.

Birke J. and Sarkar A. 2007. Active learning for the identification of nonliteral language. In Proceedings of the Workshop on Computational Approaches to Figurative Language (pp. 21-28). Association for Computational Linguistics.

Mohammad S. M., Shutova E. and Turney P. D. 2016. Metaphor as a medium for emotion: An empirical study. The* SEM 2016 Organizing Committee.

Shutova E., Kiela D. and Maillard J. 2016. Black holes and white rabbits: Metaphor identification with visual features. In Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 160-170).