A Framework for Figurative Language Detection Based on Sense Differentiation

Daria Bogdanova
University of Saint Petersburg
Saint Petersburg
dasha.bogdanova@gmail.com

Abstract

Various text mining algorithms require the process of feature selection. High-level semantically rich features, such as figurative language uses, speech errors etc., are very promising for such problems as e.g. writing style detection, but automatic extraction of such features is a big challenge. In this paper, we propose a framework for figurative language use detection. This framework is based on the idea of sense differentiation. We describe two algorithms illustrating the mentioned idea. We show then how these algorithms work by applying them to Russian language data.

1 Introduction

Various text mining algorithms require the process of feature selection. For example, authorship attribution algorithms need to determine features to quantify the writing style. Previous work on authorship attribution among computer scientists is mostly based on low-level features such as word frequencies, sentence length counts, n-grams etc. A significant advantage of such features is that they can be easily extracted from any corpus. But the study by Batov and Sorokin (1975) shows that such features do not always provide accurate measures for authorship attribution. The linguistic approach to the problem involves such high-level characteristics as the use of figurative language, irony, sound devices and so on. Such characteristics are very promising for the mentioned above tasks, but the extraction of these features is extremely hard to automate. As a result, very few attempts have been made to exploit high-level features for stylometric purposes (Stamatatos, 2009). Therefore, our long-term objective is the extraction of high-level semantically rich features.

Since the mentioned topic is very broad, we focus our attention only on some particular problems and approaches. In this paper, we examine one of such problems, the problem of automatic figurative language use detection. We propose a framework for figurative language detection based on the idea of sense differentiation. Then, we describe two algorithms illustrating the mentioned idea. One of them is intended to decide whether a usage is literal by comparing the texts related to the target expression and the set of texts related to the context itself. The other is aimed at grouping instances into literal and non-literal uses and is based on DBSCAN clustering (Ester et al, 1996). We illustrate then how these algorithms work by applying them to Russian language data. Finally, we propose some ideas on modifications which can significantly improve the accuracy of the algorithms.

2 Related Work

Sporleder and Li (April 2009) proposed an unsupervised method for recognition of literal and non-literal use of idiomatic expressions. Given an idiom the method detects the presence or absence of cohesive links between the words the idiom consists of and the surrounding text. When such links exist, the occurence is considered as a literal usage and as a non-literal when there are no such links. For most idioms the experiments showed an accuracy above 50% (it varies between 11% and 98% for different idioms). The authors then proposed an improvement of the algorithm (Li and Sporleder, August 2009) by adding the Support Vector Machine classifier as a second stage. They use the mentioned above unsupervised algorithm to label the training data for the supervised classifier. The average accuracy of the improved algorithm is about 90%. Our approach is also based on the idea of the relatedness between the expression and the surrounding context. Unlike the mentioned study, we do not focus our attention only on idioms. So far we have mostly dealt with ex-

Proceedings of the ACL 2010 Student Research Workshop, pages 67–72,
Uppsala, Sweden, 13 July 2010.

pressions, which are not necessarily idiomatic by themselves, but become metaphors in a particular context (e.g. "she is the *sunshine*", "life is a *journey*") and expressions that are invented by an author (e.g. "*my heart's in the Highlands*"). Moreover, the improved algorithm (Li and Sporleder, August 2009) is supervised, and our approach is unsupervised.

The study by Katz and Giesbrecht (2006) is also supervised, unlike ours. It also considers multiword expressions that have idiomatic meanings. They propose an algorithm, which computes the vectors for literal and non-literal usages and then use the nearest neighbor classifier to label an unseen occurence of the given idiom.

The approach proposed by Birke and Sarkar (2006) is nearly unsupervised. They constructed two seed sets: one consists of literal usages of different expressions and the other consists of non-literal usages. They calculate the distance between an occurence in question and these two sets and assign to the occurence the label of the closest set. This work, as well as ours, refers to the ideas from Word Sense Disambiguation area. Unlike our approach, the authors focus their attention only on the detection of figuratevely used verbs and, whereas we only refer to the concepts and ideas of WSD, they adapt a particular existing one-word disambiguation method.

As we have already said, we deal with different types of figurative language (metaphors, metonymies etc.). However, there are some works aimed at extracting particular types of figurative language. For example, Nissim and Markert (2003) proposed a machine learning algorithm for metonymy resolution. They state the problem of metonymy resolution as a classification task between literal use of a word and a number of predefined metonymy types.

3 Sense Differentiation

We could treat a figurative meaning of a word as an additional, not common meaning of this word. Actually, some metaphors are quite common (e.g. *eye of a needle*, *leg of a table*, etc.) and are called catachretic metaphors. They appear in a language to remedy the gap in vocabulary (Black, 1954). These metaphors do not indicate an author's writing style: an author uses such metaphor for an object because the language has no other name for that object. Therefore the algorithms we are developing do not work with this type of metaphors.

Our approach to figurative language detection is based on the following idea: the fact that the sense of a word significantly differs from the sense of the surrounding text usually indicates that the word is used figuratively. Two questions arise immediately:

1. How do we represent the sense of both the word and the surrounding context?

2. How do we find out that these senses differ significantly?

To answer the first question, we refer to the ideas popular in the Word Sense Disambiguation community: sense is a group of contextually similar occurrences of a word (Schütze, 1996). Hence, we represent the senses of both a word and its context as sets of documents related to the word and the context respectively. These sets can be obtained e.g. by searching Wikipedia, Google or another web search engine. For a word the query can be the word itself. As for a text, this query can be formulated as the whole text or as a set of some words contained in this text. It seems to us that querying the lexical chains (Halliday and Hasan, 1976) extracted from the text should provide better results than querying the whole text.

As soon as we have a sense representation for such objects as a word and a text, we should find a way to measure the difference between these sense representations and find out what difference is strong enough for the considered occurence to be classified as a non-literal usage. One way to do this is representing sets of documents as sets of vectors and measuring the distance between the centers of the obtained vector sets. Another way is to apply clustering techniques to the sets and to measure the accuracy of the produced clustering. The higher the accuracy is, the more different the sets are.

Besides, this can be done by calculating text-to-text semantic similarity using for example the measure proposed by Mihalcea et al (2006). This is rather difficult in case of the Russian language because at the moment there is no WordNet-like taxonomies for Russian.

In the next section, we propose two algorithms based on the mentioned above idea. We state the algorithms generally and try to find out experi-

mentally what combination of the described techniques provides the best results.

4 Finding the Distance to the Typical Context Set

The algorithm is intended to determine whether a word (or an expression) in a given context is used literaly or not.

As it was mentioned above, we decided to represent senses of both an expression and a context as sets of documents. Our hypothesis is that these document sets differ significantly if and only if an expression is used figuratevely. Thus, the algorithm decides whether the occurence is literal by comparing two sets of documents: the *typical context set*, which represents a sense of the expression, and the *related context set*, which represents a sense of the context. A naive way to construct the *typical context set* is searching some searching engine (e.g. Google) for the expression. Given a context with a target expression, the *related context set* can be constructed as follows:

1. Remove the target expression from the context;
2. Extract the longest lexical chains from the resulting context;
3. For every chain put to the set the first N articles retrieved by searching a searching engine for the chain;

After constructing the sets the algorithm should estimate the similarity between these two sets. This, for example, can be done by applying any clustering algorithm to the data and measuring the accuracy. Evidently, the higher the accuracy of the obtained clustering is, the more separated the sets are. It means that, when the usage is literal, the accuracy should be lower because we try to make two clusters out of data that should appear as the only cluster.

We hypothesize that in case of non-literal usages these two sets should be significantly separated.

Our experiments include two stages. During the first one we test our idea and estimate the parameters of the algorithms. During the second stage we test the more precise algorithm obtained during the first stage.

For the first stage, we found literal and non-literal occurences of the following Russian words and expressions:

вьюга (snowstorm), дыхание (breath), кинжальный (dagger), плясать (dance), стебель гибкий (flexible (flower) stalk), утонуть (be drowned), хрустальный (crystal), шотландская волынка (bagpipes), мед (honey), лекарство (medicine).

For every expression, the *typical context set* consists of the first 10 articles retrieved by searching Google for the expression. In order to construct the second set we removed the target expression from the context and manually extracted lexical chains from the texts, although, the process of lexical chains extraction can be done automatically. However the algorithms on lexical chains extraction usually use WordNet to calculate the relatedness, but as it was already mentioned WordNet for the Russian language does not exist yet. Another way to calculate semantic relatedness is using Wikipedia (Mihalcea, 2007; Turdakov and Velikhov, 2008), but it takes much effort. The second set for each occurence consists of the first 10 articles retrieved by searching Google for the extracted chains. Then we applied k-means clustering algorithm ($k = 2$) to these sets. To evaluate the clustering we used measures from the clustering literature. We denote our sets by $G = g_1, g_2$ and the clusters obtained by k-means as $C = c_1, c_2$. We define a mapping f from the elements of G to the elements of C, such that each set g_i is mapped to a cluster $c_j = f(g_i)$ that has the highest percentage of common elements with g_i. Precision and recall for a cluster $g_i, i = 1, 2$ are defined as follows:

$$Pr_i = \frac{\mid f(g_i) \cap g_i \mid}{\mid f(g_i) \mid} \text{ and } Re_i = \frac{\mid f(g_i) \cap g_i \mid}{\mid g_i \mid}$$

Precision, Pr, and recall, Re, of the clustering are defined as the weighted averages of the precision and recall values over the sets:

$$Pr = \frac{1}{2}(Pr_1 + Pr_2) \text{ and } Re = \frac{1}{2}(Re_1 + Re_2)$$

F_1-measure is defined as the harmonic mean of precision and recall, i.e.,

$$F_1 = \frac{2 \times Pr \times Re}{Pr + Re}.$$

Table 1 shows the results of the clustering. For 9 expressions out of 10, the clustering accuracy is higher in case of a metaphorical usage than in case of a literal one. Moreover, for 9 out of 10

	Figurative usage			Literal usage		
	Pr	*Re*	*F*	*Pr*	*Re*	*F*
вьюга	0,85	0,85	0,85	0,50	0,50	0,50
дыхание	0,83	0,75	0,79	0,65	0,60	0,63
кинжальный	0,85	0,85	0,85	0,70	0,65	0,67
плясать	0,95	0,95	0,95	0,66	0,65	0,66
стебель гибкий	0,85	0,85	0,85	0,88	0,85	0,86
утонул	0,85	0,85	0,85	0,81	0,70	0,75
хрустальный	0,95	0,95	0,95	0,83	0,75	0,78
шотландская волынка	0,88	0,85	0,86	0,70	0,70	0,70
мед	0,90	0,90	0,90	0,88	0,85	0,87
лекарство	0,90	0,90	0,90	0,81	0,70	0,75

Table 1: Results provided by k-means clustering

metaphorical usages, F-measure is 0,85 or higher. And for 7 out of 10 literal usages, F-measure is 0,75 or less.

The first stage of the experiments illustrates the idea of sense differentiation. Based on the obtained results, we have concluded, that F-measure value equal to 0,85 or higher indicates a figurative usage, and the value equal to 0,75 or less indicates a literal usage.

At the second stage, we applied the algorithm to several Russian language expressions used literally or figuratively. The accuracy of the k-means clustering is shown in Table 2.

Figurative usages			
живой костер из снега и вина	0,76	0,55	0,64
лев	1,00	1,00	1,00
иней	0,90	0,90	0,90
ключ	0,95	0,93	0,94
лютый зверь	0,88	0,85	0,87
рогатый	0,92	0,90	0,91
терлась о локоть	0,88	0,85	0,86
иглою снежного огня	0,95	0,95	0,95
клавишей стая	0,76	0,55	0,64
горели глаза	0,95	0,95	0,95
цветок	0,80	0,80	0,80
загорелся	0,91	0,90	0,90
Literal usages			
ловил рыбу	0,71	0,70	0,70
играл в футбол	0,74	0,70	0,71
детство	0,66	0,65	0,66
кухня	0,88	0,85	0,87
снег	0,95	0,95	0,95
весна	0,50	0,50	0,50
пить кофе	0,85	0,85	0,85
танцы	0,90	0,90	0,90
платье	0,65	0,65	0,65
человек	0,81	0,70	0,75
ветер	0,85	0,85	0,85
дождь	0,91	0,90	0,90

Table 2: Testing the algorithm. Accuracy of the k-means clustering

For 75% of metaphorical usages F-measure is 0,85 or more as was expected and for 50% of literal usages F-measure is 0,75 or less.

5 Figurative Language Uses as Outliers

The described above approach is to decide whether a word in a context is used literally or not. Unlike the first one, the second approach we propose, deals with a set of occurences of a word as to label every occurence as 'literal' or 'non-literal'. We formulate this task as a clustering problem and apply DBSCAN (Ester et al, 1996) clustering algorithm to the data. Miller and Charles (1991) hypothesized that words with similar meanings are often used in similar contexts. As it was mentioned, we can treat a meaning of a metaphoric usage of an expression as an additional, not common for the expression. That's why we expect metaphorical usages to be ouliers, while clustering together with common (i.e. literal) usages. Theoretically, the algorithm should also distinguish between all literal senses so that the contexts of the same meaning appear in the same cluster and the contexts of different meanings - in different clusters. Therefore, ideally, the algorithm should solve word sense discrimination and non-literal usages detection tasks simultaneously.

For each Russian word shown in Table 3, we extracted from the Russian National Corpora (`http://ruscorpora.ru/`) several literal and non-literal occurences. Some of these words have more than one meaning in Russian, e.g. ключ can be translated as a *key* or *water spring* and the word коса as a *plait*, *scythe* or *spit*.

word	**literal**	**non-literal**
бабочка (butterfly, bow-tie)	12	2
иней (frost)	14	2
ключ (key, spring(water))	14	2
коса (plait, scythe, spit)	21	2
лев (lion, Bulgarian lev)	17	5
лук (onion, bow)	17	1
мука (flour, pain)	21	2
пыль (dust)	14	4

Table 3: Data used in the first experiment

All the documents are stemmed and all stopwords are removed with the SnowBall Stemmer (`http://snowball.tartarus.org/`) for the Russian language.

As it was mentioned above, this algorithm is aimed at providing word sense discrimination and non-literal usages detection simultaneously. So far we have paid attention only to the non-literal usages detection aspects. DBSCAN algorithm is a density-based clustering algorithm designed to

discover clusters of arbitrary shape. This algorithm requires two parameters: ε (eps) and the minimum number of points in a cluster (minPts).

We set minPts to 3 and run the algorithm for different eps between 1.45 and 1.55.

As was mentioned, so far we have considered only figurative language detection issues: The algorithm marks an instance as a figurative usage iff the instance is labeled as an outlier. Thus, we measure the accuracy of the algorithm as follows:

$$precision = \frac{|\, figurative\ uses \,| \bigcap |\, outliers \,|}{|\, outliers \,|},$$

$$recall = \frac{|\, figurative\ uses \,| \bigcap |\, outliers \,|}{|\, figurative\ uses \,|}.$$

Figures 1 and 2 show the dependency between the eps parameter and the algorithm's accuracy for different words.

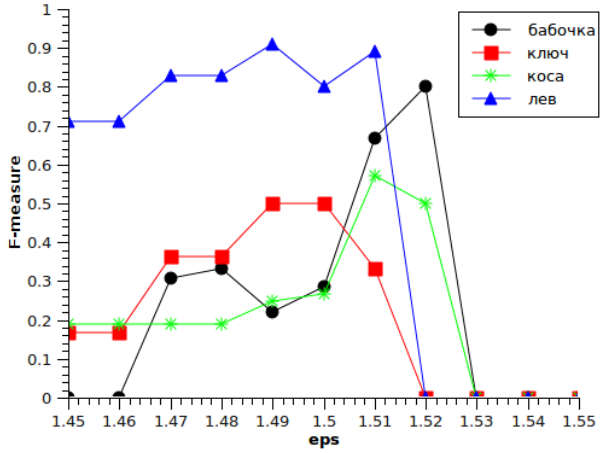

Figure 1: Dependency between eps and F-measure

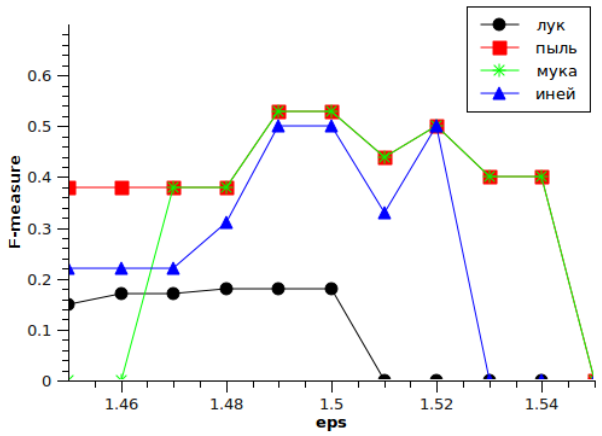

Figure 2: Dependency between eps and F-measure

Table 4 shows "the best" eps for each word and the corresponding accuracies of metaphor detection

word	eps	precision	recall
бабочка	1.520	0.66	1.00
иней	1.520	0.50	0.50
ключ	1.500	0.33	1.00
коса	1.510	0.40	1.00
лев	1.490	1.00	0.83
лук	1.505	0.17	1.00
мука	1.525	0.67	0.50
пыль	1.505	0.50	0.60

Table 4: The best eps parameters and corresponding accuracies of the algorithm

6 Future Work

So far we have worked only with tf-idf and word frequency model for both algorithms. The next step in our study is utilizing different text representation models, e.g. second order context vectors. We are also going to develop an efficient parameter estimation procedure for the algorithm based on DBSCAN clustering.

As for the other algorithm, we are going to distinguish between different figurative language expressions:

- one word expressions
  - monosemous word
  - polysemous word
- multiword expressions

We expect the basic algorithm to provide different accuracy in case of different types of expressions. Dealing with multiword expressions and monosemous words should be easier than with polysemous words: i.e., for monosemous word we expect the second set to appear as one cluster, whereas this set for a polysemous word is expected to have the number of clusters equal to the number of senses it has.

Another direction of the future work is developing an algorithm for figurative language uses *extraction*. The algorithm has to *find* figuratively used expressions in a text.

7 Conclusion

In this paper, we have proposed a framework for figurative language detection based on the idea of sense differentiation. We have illustrated how this

idea works by presenting two clustering-based algorithms. The first algorithm deals with only one context. It is based on comparing two context sets: one is related to the expression and the other is semantically related to the given context. The second algorithm groups the given contexts in literal and non-literal usages. This algorithm should also distinguish between different senses of a word, but we have not yet paid enough attention to this aspect. By applying these algorithms to small data sets we have illustrated how the idea of sense differentiation works. These algorithms show quite good results and are worth further work.

Acknowledgments

This work was partially supported by Russian Foundation for Basic Research RFBR, grant 10-07-00156.

References

Vitaly I. Batov and Yury A. Sorokin. 1975. Text attribution based on objective characteristics. *Seriya yazyka i literatury*, 34, 1.

Julia Birke and Anoop Sarkar. 2006. A Clustering Approach for the Nearly Unsupervised Recognition of Nonliteral Language. *Proceedings of EACL-06*

Max Black. 1954. Metaphor. *Proceedings of the Aristotelian Society*, 55, pp. 273-294.

Martin Ester, Hans-Peter Kriegel, Jörg Sander and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, AAAI Press, pp. 226231

Michael Halliday and Ruqaiya Hasan. 1976. Cohesion in English. *Longman, London*

Graham Katz and Eugenie Giesbrecht. 2006. Automatic identification of non-compositional multiword expressions using latent semantic analysis. *Proceedings of the ACL/COLING-06 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*

Linlin Li and Caroline Sporleder. August 2009. Classifier combination for contextual idiom detection without labeled data. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pp. 315-323.

Rada Mihalcea, Courtney Corley and Carlo Strapparava. 2006. Corpus-based and Knowledge-based Measures of Text Semantic Similarity. *Proceedings of AAAI-06*

Rada Mihalcea. 2007. Using Wikipedia for Automatic Word Sense Disambiguation. *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL 2007)*

George A. Miller and Walter G. Charles. 1991. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):128.

Malvina Nissim and Katja Markert. 2003. Syntactic features and word similarity for supervised metonymy resolution. *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (ACL-03) (Sapporo, Japan, 2003). 56-63.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1), pp. 97-123

Caroline Sporleder and Linlin Li. April 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. *Proceedings of EACL-09*

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3): 538-556.

Denis Turdakov and Pavel Velikhov. 2008. Semantic Relatedness Metric for Wikipedia Concepts Based on Link Analysis and its Application to Word Sense Disambiguation *SYRCoDIS 2008*