Extraction of Domain-specific Opinion Words for Similar Domains

Ilia Chetviorkin
Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University
ilia2010@yandex.ru

Natalia Loukachevitch
Research Computing Center Lomonosov Moscow State University
louk_nat@mail.ru

Abstract

In this paper we consider a new approach for domain-specific opinion word extraction in Russian. We suppose that some domains have similar sentiment lexicons and utilize this fact to build an opinion word vocabulary for a group of domains. We train our model in movie domain and then utilize it to book and game domains. Obtained word list quality is comparable with quality of initial domain list.

1 Introduction

The web is full of customers' opinions on various products. Automatic extraction, processing and summarization of such opinions are very useful for future users. Opinions about products are often expressed using evaluative words and phrases that have a certain positive or negative sentiment. Therefore, important features in the qualitative classification of opinions about a particular entity are opinion words and expressions used in the domain. The problem is that it is impossible to compile a list of opinion expressions, which will be equally applicable to all domains, as some opinion phrases are used only in a specific domain while the others are context-oriented [Lu et. al., 2011]. Indeed, sentiment lexicons adapted to a particular domain or topic have been shown to improve task performance in a number of applications, including opinion retrieval [Jijkoun et. al., 2010], and expression-level sentiment classification [Choi and Cardie, 2009]. In addition there are several studies about context-dependent opinion expressions [Lu et. al., 2011].

The number of different domains is very large, and recent studies are focused on cross-domain approaches, to bridge the gap between the domains [Pan et al, 2010]. On the other side there are different subject fields that has similar sentiment lexicon. For example: «breathtaking» is an opinion word in entertainment (movies, books, games etc.) domain, but non-opinion in the politics domain. At the opposite side some words («evil», «treachery» etc.) have strong sentiment in politics domain, but are neutral in entertainment domain, these words do not express any opinion about a film, game or book.

Thus we suppose that different domains can be separated into clusters (for example: entertainment, digital goods, politics, traveling etc.) where domains of the same cluster have similar sentiment lexicons.

In this paper we focus on the problem of construction of a domain-specific sentiment lexicon in Russian, which can be utilized for various similar domains.

We present a new supervised method for domain-specific opinion word extraction. We train this method in one domain and then utilize it in two others. Then we combine extracted word lists to construct a general list of opinion words typical to this domain cluster.

Our approach is based on several text collections, which can be automatically formed for many subject areas. The set of text collections includes: a collection of product reviews with author evaluation scores, a text collection of product descriptions and a contrast corpus (for example, a general news collection). For each word in a review collection we calculate various statistical features using aforementioned collections and then apply machine learning algorithms for term classification.

To evaluate the effectiveness of the proposed method we conduct experiments on data sets in three different domains: movies, books and computer games. The results show that our approach can identify new opinion words specific to the given domain (for example "fabricated" in movie domain).

For further evaluation of the lexicon quality, we manually labeled extracted word lists, and our method is proved to be effective in construct-

Proceedings of the Workshop on Information Extraction and Knowledge Acquisition, pages 7–12,
Hissar, Bulgaria, 16 September 2011.

ing a qualitative list of domain-dependent sentiment lexicon. The results also demonstrate the advantage of combining multiple lists of opinion words over using any single list.

The reminder of this article is organized as follows. In Section 2 we describe the state-of-the-art in the opinion words extraction sphere, Section 3 describes our approach in the movie domain, in Section 4 we utilize our approach for two other domains and combine opinion word vocabularies for all three domains.

2 Related Work

Sentiment lexicon plays an important role in most, if not all, sentiment analysis applications, including opinion retrieval, opinion question answering and summarization, opinion mining [Ding et. al., 2008]. Even though supervised machine learning techniques have been shown to be effective for sentiment classification task [Pang and Lee, 2008], authors in [Choi and Cardie, 2009] demonstrate that including features from sentiment lexicons boosts classification performance significantly.

Generally there are three main approaches to the automatic identification of opinion words in texts.

The first approach is manual labeling, which is very labor-intensive and error-prone process. In addition the coverage of this approach is usually very low.

The second approach is based on information from a dictionary or a thesaurus. In this approach a small initial set of words is usually chosen manually, and then expanded with the help of dictionaries and thesaurus entries. The basic principle of this approach is that if a word has sentiment polarity, then its synonyms and antonyms have polarity too (orientation may change). Therefore, from the initial set of words, a new, more complete set of opinion words can be constructed [Hu and Liu, 2004, Neviarouskaya et.al., 2009]. In [Esuli and Sebastiani, 2005], dictionary definitions are used for opinion words extraction. The basic idea is that words with the same orientation have "similar" glosses.

The third approach – corpus-based training. This approach is based on finding rules and patterns in the texts [Kanayama and Nasukawa, 2006]. In [Turney, 2002] word polarity is calculated by comparing the co-occurrence statistics of various words with words “excellent” and “poor”. Authors assume that words with similar semantic orientation tend to co-occur. The resulting opinion orientation of the words is used to classify reviews to positive and negative.

There are some works that combine second and third approaches [Ding et. al., 2008]. More importantly, although existing works try to learn opinion words in a specific domain, few of them directly evaluate the quality of the generated lexicon.

3 Proposed method

In this section we will describe our method in respect to movie domain. We will train the model on the movie data and then try to utilize it in other domains.

3.1 Data Preparation

We collected 28773 film reviews of various genres from online recommendation service *www.imhonet.ru*. For each review, user’s score on a ten-point scale was extracted. We called this collection the **review collection**.

Example of the movie review:

Nice and light comedy. There is something to laugh - exactly over the humor, rather than over the stupidity... Allows you to relax and gives rest to your head.

We also needed a contrast collection of texts for our experiments. In this collection the concentration of opinions should be as little as possible. For this purpose, we had collected 17680 movie descriptions. This collection was named **description collection**.

One more contrast corpus was a collection of one million news documents. We had calculated document frequency of each word in this collection and used only this frequency list further. This list was named **news corpus**.

3.2 Collections with Higher Concentration of Opinions

We suggested that it was possible to extract some fragments of the reviews from review collection, which had higher concentration of opinion words. These fragments include:

- Sentences ending with a «!»;
- Sentences ending with a «…»;
- Short sentences, no more than 7 word length;
- Sentences containing the word «movie» without any other nouns.

We call this collection – **small collection**.

3.3 Statistical Features

Our task was to create a qualitative list of opinion words based on the calculation of various features. We used the following set of features for each word:

- Frequency of the word in the collection (i.e. number of occurences in all documents in the collection)
- The number of documents where the word occurs
- Weirdness
- TFIDF
- Deviation from the average score
- Word score variance
- Frequency of capitalized words

We will consider some of them in more detail.

Weirdness. To calculate this feature two collections are required: one with high concentration of opinion words and the other – contrast one. The main idea of this feature is that opinion words will be «strange» in the contexts of the contrast collection. This feature is calculated as follows [Ahmad et. al, 1999]:

$$\text{Weirdness} = \frac{w_s / t_s}{w_g / t_g}$$

where w_s – frequency of the word in special corpus, t_s – total count of words in special corpus, w_g – frequency of the word in general corpus, t_g – total count of words in general corpus. Instead of frequency one can use the number of documents where the word occurs.

TFIDF. There are many varieties of this feature. We used TFIDF variant described in [Callan et. al., 1992] (based on BM25 function):

$$\text{TFIDF} = \beta + (1-\beta)\cdot tf \cdot idf$$

$$tf_D(l) = \frac{\text{freq}_D(l)}{\text{freq}_D(l) + 0.5 + 1.5 \cdot \frac{dl_D}{\text{avg\_dl}}}$$

$$idf(l) = \frac{\log\left(\frac{|c| + 0.5}{df(l)}\right)}{\log(|c| + 1)}$$

freq(l) – number of occurrences of l in a document (collection),

dl(l) – length measure of a document,

avg_dl – average length of a document,

df(l) – number of documents in a collection (e.g. movie descriptions, news collection) where term l appears,

$\beta = 0.4$ by default,

|c| – total number of documents in a collection.

Deviation from the average score. As we mentioned above we had collected user's numerical score (on a ten point scale) for each review. The main idea of this feature is to calculate average score for each word (sum of review ratings where this word occurs divided into their number) in the collection and then subtract average score of all reviews in the collection from it.

$$dev(l) = \left| \frac{\sum_{i=1}^{n} m_i k_i}{k} - \frac{\sum_{i=1}^{n} m_i}{n} \right|$$

$$\sum_{i=1}^{n} k_i = k$$

where l – considered lemma, n – total count of the reviews in the collection, m_i – i-th review score, k_i – frequency of the lemma in the i-th review (may be 0).

Word score variance. Using review ratings we can calculate the score variance for each word. This feature can show us how often a word is used in reviews with significantly different scores. If a word has small deviation then it is used in reviews with similar scores and has high probability to be an opinion word.

$$Var(l) = \frac{\sum_{i=1}^{n} m_i^2 k_i}{k} - \left(\frac{\sum_{i=1}^{n} m_i k_i}{k} \right)^2$$

$$\sum_{i=1}^{n} k_i = k$$

where l – considered lemma, n – total count of the reviews in the collection, m_i – i-th review score, k_i – frequency of the lemma in the i-th review (may be 0).

Frequency of words, which start with the capital letter. The meaning of this feature is the frequency (in the review corpus) of each word starting with the capital letter and not located at the beginning of the sentence. With this feature we are trying to identify potential proper names, which are always neutral.

3.4 Feature and Collection Combinations

For our experiments we took top ten thousand words ordered by frequency from the movie review collection.

For each word from this list we had the following combinations of features and collections:

- TFIDF calculation using the pairs of collections: *small-news*, *small-description*, *opinion-news*, *opinion-description*;
- Weirdness calculation using the pairs of collections: *opinion-news* and *opinion-description* with document count and *small-description*, *opinion-description* with frequency;
- Deviation from the average score;
- Word score variance
- Word frequency in *opinion* and *small collections*;
- Total number of documents in the *opinion corpus*, where the word occurs;
- Frequency of capitalized words.

In addition, separately for description corpus we calculated the following features: frequency, document count, weirdness using *description-news collections* with document count and TFIDF using the same pair. Thus, each term had 18 features.

3.5 Algorithms and Evaluation

To train supervised machine learning algorithms we needed a set of labeled opinion words. We decided to label the full list of ten thousand words manually and then to use cross-validation. We marked up word as opinion one in case we could imagine it in any opinion context in the movie domain. All words were tagged by two authors.

As a result of our mark up we obtained the list of 3200 opinion words (1262 adjectives, 296 adverbs, 857 nouns, 785 verbs).

Our aim in this part of work was to classify words into two classes: opinion or neutral.

For this purpose Weka[1] data mining tool was used. We considered the following algorithms: *Logistic Regression* and *LogitBoost*. For all experiments 10 fold cross-validation was used.

Using aforementioned algorithms we obtained term lists, ordered by the predicted probability of their opinion orientation. To measure the quality of these lists we used *Precision@n* metric. This metric is very convenient for measuring the quality of list combinations and it can be used with different thresholds. For the algorithms quality comparison in different domains we chose n = 1000. This level is not too large for manual labeling and demonstrates the quality in an appropriate way.

The results of classification are in Table 1.

Logistic Regression	LogitBoost	Sum
66.00%	66.80%	70.90%

Table 1. *Precision@1000* of word classification

We noticed that the lists of opinion words extracted using two logistic algorithms differ significantly. So we decided to sum the weights of words in these two lists. The result of this summation can be found in the last column of the Table 1 and on the Figure 1.

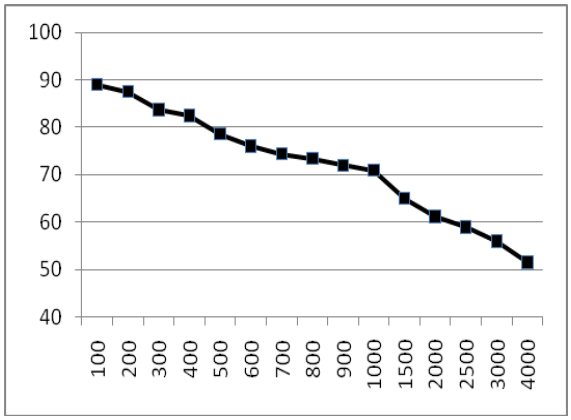

Figure 1. *Precision@n* in the Sum list (depending on n)

As the baseline for our experiments we used lists ordered by frequency in the review collection and Deviation from the average score. *Precision@1000* in these lists was 27.5% and 40.5% accordingly. Thus our algorithms gave significant improvements over baseline. All the other features can be found in Table 2.

Let us look at some examples of opinion words with the high probability value in the sum list:

Trogatel'nyi (affective), otstoi (trash), fignia (crap), otvratitel'no (disgustingly), posredstvenniy (satisfactory), predskazuemyi (predictable), ljubimyj (love) etc.

Obtained opinion word lists can be utilized in various sentiment analysis tasks. For example

[1] http://www.cs.waikato.ac.nz/ml/weka/

words can be used as features for document classification by the overall sentiment.

Feature	Collection	Precision @1000
TFIDF	small – news	39.2%
TFIDF	small – descr	36.3%
TFIDF	review – news	33.8%
TFIDF	review – descr	30.4%
Weirdness	review – news (doc. count)	51.1%
Weirdness	review – descr (doc. count)	47.7%
Weirdness	small – descr (frequency)	49.2%
Weirdness	review – descr (frequency)	46.0%
Deviation from the average score	review	40.5%
Word score variance	review	31.7%
Frequency	review	27.5%
Frequency	small	32.1%
Document Count	review	27.9%

Table 2. *Precision@1000* for different features

In [Chetviorkin et. al, 2011] we used opinion words in three-way review classification task and improved the quality of classification using opinion word weights.

3.6 Collection and Feature Selection

Finally, we studied the impact of each collection to the resulting quality of the opinion word classification. All collections (except review collection) were consequently excluded from constructing features. Additionally influence of the deviation from the average score, word score variance and frequency of words starting with capital letter were explored. In Table 3 results of classification with different feature sets can be found.

Thus, one can see that all collections and features improve the quality of classification. Exclusion of the description collection yields practically identical results for the sum list. Nevertheless this collection is very useful from model utilization in other domains (without it quality drops significantly).

Feature	Logistic Regression	Logit-Boost	Sum
All \ small collection	60.7%	66.7%	66.5%
All \ descr collection	61.3%	67.2%	70.6%
All \ news collection	66.1%	67.1%	69.0%
All \ deviation from the average score	64.4%	64.1%	68.6%
All \ word score variance	62.9%	64.3%	67.6%
All \ frequency of capitalized words	61.1%	61.7%	64.4%

Table 3. *Precision@1000* for different feature sets

4 Model Utilization to Similar Domains

In the previous section we constructed a new model for domain-specific opinion word extraction. We want to utilize this model in the other domains and evaluate the quality of obtained word lexicons and their combinations.

4.1 Data

We collected data on two more domains: book domain and computer games domain. The structure of the data was the same as for movie domain. Book and games review collections contained 16497 book reviews and 7928 game reviews of various genres accordingly. For each review, user's score on a ten-point scale was extracted.

The contrast collections of texts for book domain and games domain contained 24555 book descriptions and 1853 game descriptions.

Here we used the same **news corpus** as for movie domain.

4.2 Model Utilization and Evaluation

For new domains we extracted ten thousand the most frequent words (or all available words with frequency more then 3) and calculated all statistical features, which were described in Section 3.3. At the next step we applied our model trained in the movie domain to the book and games word lists. To evaluate the quality of word

classification in new domains we manually labeled first thousand of words in each list. The results of classification are in Table 4.

	Logistic Regression	LogitBoost	Sum
Books	69.60%	59.10%	72.20%
Games	49.40%	63.00%	62.90%

Table 4. Results of the classification in book and games domains.

At the final step we took linear combination of the words (sum of word weights) in each list from three different domains (6 lists). The *Precision@1000* of the obtained opinion word list was **82.0%.**

We supposed that this general opinion word lexicon could improve the quality of the best list obtained in the movie domain. We summed weights of the best combined list in movie domain and general one (from three domain lists). Weights of the latter list were normalized previously. The quality of obtained movie domain-specific word dictionary was **71.8%. So exploitation of opinion words from other similar domains improved extraction of opinion words in the initial domain (+1.26%).**

5 Conclusion

In this paper, we described a method for opinion word extraction for any domain on the basis of several domain specific text collections. We utilized our algorithm in different domains and showed that it had good generalization abilities. The quality of the combined list was significantly better then the quality of each single list. Usage of the combined list improved extraction of opinion words in the initial domain.

Acknowledgements. This work is partially supported by RFBR grant N11-07-00588-a.

References

Ahmad K., Gillam L., Tostevin L. 1999. *University of Surrey participation in Trec8: Weirdness indexing for logical documents extrapolation and retrieval* In the Proceedings of Eigth Text Retrieval Conference (Trec-8).

Callan J.P., Croft W.B., Harding S.M. 1992. *The INQUERY Retrieval System* Proc. of Database and Expert System Applications DEXA-92, 3rd International Conference on Database and Expert Systems Applications / A.M. Tjoa and I. Ramos (eds.). – Springer Verlag, New York, pp.78-93.

Chetviorkin I. and Loukachevitch N. 2011. *Three-way movie review classification.* In International Conference on Computational Linguistics Dialog.

Choi Y. and Cardie C. 2009 *Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification.* In EMNLP '09, pages 590–598.

Ding X., Liu B., and Yu P. S. 2008. *A holistic lexicon-based approach to opinion mining*. In WSDM '08, pages 231–240.

Esuli A., Sebastiani F. 2005 *Determining the Semantic Orientation of Terms through Gloss Classification*. In: Conference of Information and Knowledge Management

Hu M., Liu B. 2004. *Mining and Summarizing Customer Reviews.* KDD

Jijkoun V., de Rijke M., and Weerkamp W. 2010. *Generating focused topic-specific sentiment lexicons*. In ACL '10, pages 585–594,

Kanayama H. and Nasukawa T. 2006. *Fully automatic lexicon expansion for domain-oriented sentiment analysis.* In EMNLP '06, pages 355–363, Morristown, NJ, USA.

Lu Y., Castellanos M., Dayal U. and Zhai C. 2011. *Automatic Construction of a Context-Aware Sentiment Lexicon: An Optimization Approach* In Proceedings of the World Wide Web Conference (WWW)

Neviarouskaya A., Prendinger H., and Ishizuka M. 2009. *Sentiful: Generating a reliable lexicon for sentiment analysis*. In ACII, pages 1–6, sep. 2009.

Pan, S. J., Ni, X., Sun, J-T, Yang, Q. and Chen, Z. 2010. *Cross-Domain Sentiment Classification via Spectral Feature Alignment.* In Proceedings of the World Wide Web Conference (WWW)

Pang B., Lee L. 2008. *Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval*. Now Publishers

Turney P.D. 2002. *Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews*. In: Proceedings of ACL. pp. 417-424.