ВЫСОКОТОЧНЫЙ МЕТОД ИЗВЛЕЧЕНИЯ АСПЕКТНЫХ ТЕРМИНОВ ДЛЯ РУССКОГО ЯЗЫКА

Майоров В. (vmayorov@ispras.ru),
Аванесов В. (avanesov@ispras.ru),
Андрианов И. (ivan.andrianov@ispras.ru),
Астраханцев Н. (astrakhantsev@ispras.ru),
Козлов И. (kozlov-ilya@ispras.ru),
Турдаков Д. (turdakov@ispras.ru)

Институт Системного Программирования РАН, Москва, Россия

Ключевые слова: извлечение аспектных терминов, анализ эмоциональной окраски, извлечение именованных сущностей, автоматическое извлечение терминов

A HIGH PRECISION METHOD FOR ASPECT EXTRACTION IN RUSSIAN

Mayorov V. (vmayorov@ispras.ru),
Andrianov I. (ivan.andrianov@ispras.ru),
Astrakhantsev N. (astrakhantsev@ispras.ru),
Avanesov V. (avanesov@ispras.ru),
Kozlov I. (kozlov-ilya@ispras.ru),
Turdakov D. (turdakov@ispras.ru)

Institute for System Programming of RAS, Moscow, Russia

This paper presents a work carried out by ISPRAS on aspect extraction task at SentiRuEval 2015. Our team submitted one run for Task A and Task B and got best precision for both tasks for all domains among all participants. Our method also showed the best F1-measure for exact aspect term matching for task A for automobile domain and both for Task A and Task B for restaurant domain.

The method is based on sequential classification of tokens with SVM. It uses local, global, syntactic-based, GloVe, topic modeling and automatic term recognition features. In this paper we also present evaluation of significance of different feature groups for the task.

Key words: Aspect Extraction, Sentiment Analysis, NERC, Syntax Trees, Topic modeling, GloVe, Automatic Term Recognition

Introduction

This paper describes participation in aspect extraction tasks of SentiRuEval 2015, which focuses on detecting aspect terms in reviews for restaurant and cars.

Aspect extraction is a part of object-oriented sentiment analysis. An author of a text can have different opinions relative to specific properties of an object called aspects. Aspect terms represent these aspects in particular text.

Organizers of the competition divided all aspect terms into three types: Explicit aspects, Implicit aspects, Sentiment facts (Lukashevich N. V. et. al. 2015). According to the task definition, «Explicit aspects denote some part or characteristics of a described object such as staff, pasta, music in restaurant reviews. [...] Implicit aspects are single words or single words with sentiment operators that contain within themselves as specific sentiments as the clear indication to the aspect category. In restaurant reviews the frequent implicit aspects are such words as tasty (positive+food) [...] Sentiment facts do not mention the user sentiment directly, formally they inform us only about a real fact, however, this fact conveys us a user's sentiment as well as the aspect category it related to. For example, sentiment fact *отвечала на все вопросы* (*answered all questions*) means positive characterization of the restaurant service".

SentiRuEval dataset was annotated with these three subtypes of aspect terms and participants were asked to extract separately only explicit aspect terms and all aspect terms. In the rest of the paper we will refer to explicit aspect extraction task as "Task A" and all aspect extraction task as "Task B".

Our aspect extraction system uses supervised machine learning with support vector machines (SVM) to classify each token of a review into classes which denote beginning or middle of an aspects or term outside aspect. We train our classifier only on explicit aspect terms in order to perform Task A, and use union of results of three different classifiers trained for extraction of each type of aspects separately.

Main challenge was search of good feature space. We define three groups of features: local features computed in the bounds of one sentence; global features calculated for one document; and features that use external resources.

The paper is organized as follows: Section 1 gives brief overview of the related work; in Section 2 we present full description of our method and feature space it uses; Section 3 provides evaluation for different combination of features for each task; in the final section we make conclusion for this work.

1. Related work

Aspect extraction task has been widely studied in recent years. There are four main approaches (Liu, 2012) for this task. The first approach is to extract frequent nouns and noun phrases (Hu & Liu, 2004) (Popescu & Etzioni, 2007) (Scaffidi et al., 2007). The second one utilizes opinion word and target relations (Hu & Liu, 2004) (Qiu et al., 2011) (Poria et al. 2014). These methods are based on the idea that opinion words (i.e. words or phrases that specify sentiment) are related to aspect expressions in reviews. The third approach uses topic modeling (Mei et al., 2007) (Branavan et al., 2008) (Li, Huang & Zhu, 2010). The last approach is based on supervised machine learning. The most effective methods were shown to be sequential learning, namely Hidden Markov Models (Jin & Ho, 2009) and Conditional Random Fields (Jakob & Gurevych, 2010) (Choi & Cardie, 2010).

2. Method description

2.1. Overview

User's opinion could be expressed in several ways. Each aspect in datasets provided by organizers was marked with one of five types of expression: *relevant* (aspect term mention is relevant for current review object), *comparison* (aspect term is mentioned in comparison with another object), *previous* (aspect term is mentioned in comparison with previous experience), *irrealis* (aspect term is mentioned to describe hypothetical not materialized state of things) and *irony* (aspect term is mentioned with irony). We merged all marks except *relevant* to one class *"other"* due to relatively small number of aspects with marks *comparison*, *irony* etc.

At first we tokenize all reviews and transform task into sequence labelling task: given list of tokens assign sequence of tags to each element of sequence. Our method assigns one of five following classes to each token:

1. Out of aspect term
2. Beginning of *relevant* aspect term
3. Middle of *relevant* aspect term
4. Beginning of *other* aspect term
5. Middle of *other* aspect term

Each token is classified using SVM with L2 regularization. Used features are briefly described below.

We use Texterra system (Turdakov et. al., 2014) as general NLP tasks solution for text tokenization, PoS tagging and morphological analysis. Also we use MaltParser (Nivre et al., 2007) trained on SynTagRus[1] corpora for syntactic parsing.

2.2. Local features

Local features are features that are computed using only sentence. The main local feature used in our method is classification labels of tokens in left window of size 2.

We note that aspect extraction task is very similar to named entity recognition task (NERC). So, we use some features that are successfully used in supervised machine learning NERC method (Zhang & Johnson, 2003). Used NERC features are described in section 2.2.1.

Because Russian language has free word order, we decided to use sentence syntactic structure based features (see section 2.2.2).

[1] http://www.ruscorpora.ru/instruction-syntax.html

2.2.1. NERC features

We note that aspect extraction task is very similar to named entity recognition task. So, as basic features we choose following features that are described in (Zhang & Johnson, 2003).

Token prefixes and suffixes of length 1–4; token word forms, POS tags, morphological properties, lemmas in sentence window of size 2; whether a token placed at start of a sentence; token mask (all digits in token are replaced to a special character) and some token spelling features in window of size 2 (are all characters in uppercase / digits or punctuation marks / non letters / digits or letters; is any character a digit; is first character in uppercase).

2.2.2. Syntactic features

We use following features based on sentence syntactic structure. Distance in sentence syntactic tree between current token and other tokens in window of size 3. Lemma, POS tag and token morphological properties for parent token (in terms of syntactic tree) and for each child token. Classification labels assigned to parent and children tokens in left window.

2.3. Global features

Global features are features that are computed using the whole document. We use some of features used for supervised machine learning based NERC method (Ratinov & Roth, 2009): relative frequency of classification labels for all tokens having an equal word form with current one in left window of size 1000; relative frequency of having upper case first character for all tokens having an equal word form with current one in left window of size 200; relative frequency of POS tags, morphological properties and lemmas for all tokens having an equal word form with current one in left window of size 200.

2.4. Features based on external resources

2.4.1. Glove

We also use word to vector space embedding as features. In order to obtain the embedding to 50-dimensional vector space we train GloVe (Pennington, 2014) on Russian Wikipedia. Unfortunately, the vectors assigned to words are non-interpretable but they are known to be similar (in terms of Euclidean distance) for similar words. In order to obtain interpretable features we discover clusters of words using a fuzzy clustering approach—Gaussian Mixture Model (GMM) with 200 clusters—the number of clusters is optimized via Bayesian Information Criterion which is known to be a sufficient estimate for GMM (Roeder and Wasserman, 1995). And finally, the posterior distribution of clusters given for the vector embedding of a word is used as features.

2.4.2. Topic Modeling

Topic modeling is a fuzzy clustering approach usually used to clusterize documents by topics. The very basic topic model—Probabilistic Latent Semantic Analysis (Hofmann, 1999) was employed. This model assumes that every document was drawn from a mixture of multinomial distributions over words. The components of the mixture are referred as topics. So, as a result of topic modeling, we obtain a distribution of words given the topic. Using Bayes' theorem we can easily compute the distribution of topics given the words. Finally, this distribution is used as a feature. The model was trained using a large unlabelled dataset of user's reviews. The tm[2] implementation was used.

2.4.3. Automatic Term Recognition

Since aspects are usually expressed by domain-specific terms, we check if the particular word-candidate is a part of domain-specific term. To do so, we apply methods for Automatic Term Recognition. Most of them, including those used by us, work as follows: take domain-specific text collection as an input; extract term candidates (n-grams filtered by the pre-specified part of speech patterns); compute features (e.g. frequency of term occurrences or tf-idf); and finally, classify or rank term candidates based on their feature vectors. In this work we skip the last step, i.e. we obtain the feature vector for each term candidate and then use it as follows: during a review text processing, we greedily search term candidates among word token sequences so that the longest appropriate term candidate is chosen, then we attach the corresponding feature vector to each word token from the matched sequence.

In particular, as an input text collection we use a combination of train and test data sets and also a set of documents crawled from the Web—namely, 44567 docs (82.6 Mb) from restoclub.ru for Restaurant domain and 7590 reviews (28.5 Mb) from otzovik.com for Automobile domain.

The following features are taken: 3 well-known features: Frequency; TF-IDF; C-Value (Frantzi et al., 2000) in modification that supports single-word terms (Lossio-Ventura et al., 2013); and 4 our features (Astrakhantsev, 2014): ExistsInKB—a boolean feature indicating if a term candidate is presented in Wikipedia; Link Probability—a probability of term candidate to be a hyperlink in Wikipedia; Key concept relatedness—a semantic relatedness value computed over Wikipedia to automatically found key concepts; PUATR—result of probabilistic Positive-Unlabeled classifier trained on top 100 term candidates (found by special method based on frequencies of nested occurrences) as positives and other candidates as unlabeled with all previously described features.

3. Evaluation

3.1. SVM parameter estimation

For SVM parameter estimation we perform 10-fold cross-validation on available training data with C parameter from 0.001 to 0.2 with step 0.001 in two settings (see Fig. 1). First settings is testing on training data (red line), the second settings is normal cross-validation (green line). As one can see, when to $C < 0.045$ F1 score grow for both train and test data.

For $C > 0.45$ F1 measure for train is grow and for test data it is stay almost same, thus we decided that this is frontier between over and underfitting. Thus we set C equals to 0.45

[2] https://github.com/ispras/tm

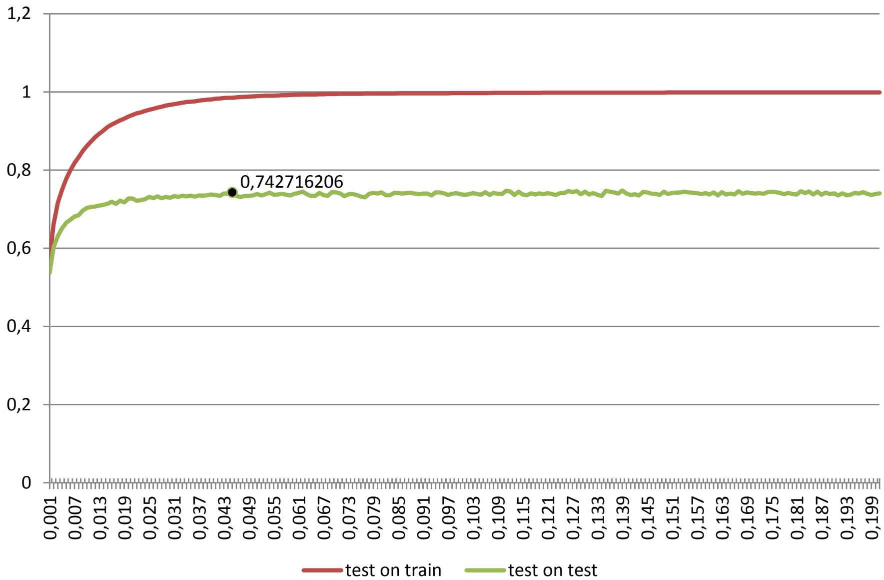

Fig 1. Method performance with different SVM parameter

3.2. Evaluation of feature groups impact

In order to understand impact of each feature group we sequentially remove each group from our feature set and measure method quality for task A. For quality measurement we perform repeated 10 times 10-fold cross-validation and compute 95% confidence interval for each quality metric. Results for automobile domain is presented in Table 1. Table 2 presents results for restaurant domain.

Table 1. Quality results (95% confidence intervals) for different features sets for Automobile domain (Task A)

features set	exact matching			partial matching		
	precision	recall	f1	precision	recall	f1
all	(0.7061; 0.7197)	(0.6500; 0.6618)	(0.6773; 0.6885)	(0.8080; 0.8200)	(0.6975; 0.7114)	(0.7493; 0.7604)
all—GloVe	(0.7107; 0.7249)	(0.6467; 0.6584)	(0.6775; 0.6891)	(0.8139; 0.8257)	(0.6888; 0.7015)	(0.7467; 0.7573)
all—TM	(0,7031; 0,7166)	(0,6427; 0,6548)	(0,6720; 0,6832)	(0,8061; 0,8181)	(0,6882; 0,7016)	(0,7431; 0,7540)
all—ATR	(0,7032; 0,7165)	(0,6414; 0,6537)	(0,6713; 0,6826)	(0,8066; 0,8185)	(0,6915; 0,7059)	(0,7452; 0,7565)
all—global	(0,7046; 0,7185)	(0,6509; 0,6633)	(0,6771; 0,6888)	(0,8068; 0,8190)	(0,6990; 0,7129)	(0,7496; 0,7609)

features set	exact matching			partial matching		
	precision	recall	f1	precision	recall	f1
all—syntactic	(0,7132; 0,7276)	(0,6582; 0,6706)	(0,6850; 0,6968)	(0,8155; 0,8268)	(0,7069; 0,7203)	(0,7579; 0,7685)
all—NERC	(0,6373; 0,6535)	(0,5120; 0,5253)	(0,5682; 0,5810)	(0,7655; 0,7798)	(0,5812; 0,5968)	(0,6611; 0,6747)

Table 2. Quality results (95% confidence intervals) for different features sets for Restaurant domain (Task A)

features set	exact matching			partial matching		
	precision	recall	f1	precision	recall	f1
all	(0,7122; 0,7260)	(0,6546; 0,6692)	(0,6830; 0,6942)	(0,7894; 0,8024)	(0,7012; 0,7143)	(0,7439; 0,7530)
all—GloVe	(0,7146; 0,7284)	(0,6529; 0,6672)	(0,6831; 0,6943)	(0,7956; 0,8080)	(0,6963; 0,7093)	(0,7438; 0,7528)
all—TM	(0,7140; 0,7281)	(0,6450; 0,6591)	(0,6786; 0,6896)	(0,7912; 0,8045)	(0,6884; 0,7017)	(0,7375; 0,7467)
all—ATR	(0,7106; 0,7247)	(0,6514; 0,6662)	(0,6805; 0,6920)	(0,7887; 0,8020)	(0,6972; 0,7106)	(0,7414; 0,7507)
all—global	(0,7118; 0,7256)	(0,6551; 0,6696)	(0,6831; 0,6941)	(0,7893; 0,8017)	(0,7045; 0,7177)	(0,7458; 0,7545)
all—syntactic	(0,7101; 0,7249)	(0,6570; 0,6713)	(0,6833; 0,6949)	(0,7947; 0,8076)	(0,7009; 0,7144)	(0,7461; 0,7554)
all—nerc	(0,6325; 0,6488)	(0,5109; 0,5265)	(0,5656; 0,5795)	(0,7426; 0,7571)	(0,5775; 0,5929)	(0,6504; 0,6627)

As one can see, only NERC features make a meaningful contribution to the method. Other feature groups are not so significant.

3.3. Method performance on SentiRuEval testing dataset

The quality of proposed method trained on all available training data with all described feature groups are presented in table 3 for task A and in table 4 for Task B. These results are obtained by SentiRuEval organizers.

Table 3. SentiRuEval Task A experiment results

Domain	exact matching			partial matching		
	precision	recall	f1	precision	recall	f1
Automobile	0.760041	0.621793	0.676118	0.856055	0.655098	0.730366
Restaurant	0.723656	0.573800	0.631871	0.807759	0.616549	0.689096

Table 4. SentiRuEval Task B experiment results

Domain	exact matching			partial matching		
	precision	recall	f1	precision	recall	f1
Automobile	0.770100	0.553546	0.636623	0.866178	0.549210	0.659989
Restaurant	0.733599	0.513197	0.596179	0.814496	0.479988	0.590601

Conclusion

We have described aspect term extraction system, which employs SVM with a broad set of features. This system perform with high precision and good F1-measure on all settings and showed one of the best results among 21 runs received for aspect extraction tasks of SentiRuEval.

In addition, we made evaluation of impact of different feature groups and found that features used for named entity recognition are most useful for aspect extraction too. We also found that removing some features could slightly improve results of cross-validation. One of the reasons for such phenomena is sparsity of feature set. Therefore we can guess that feature selection and dimensionality reduction could improve quality of the proposed method. In addition, we should note that due to lack of time, we estimated SVM parameter only on full feature set and use it for all experiments. However SVM parameter estimation for each feature combination can improve overall performance of the system. This make a slot for future improvement of the proposed method.

References

1. *Astrakhantsev N.,* (2014), Automatic term acquisition from domain- specific text collection by using Wikipedia, The Proceedings of ISP RAS [Trudy ISP RAN], vol. 26, issue 4, P. 7–20.
2. *Fangtao L., Huang M., Zhu X.,* (2010), Sentiment analysis with global topics and local dependency, in Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-2010).
3. *Frantzi K., Ananiadou S., Mima H.,* (2000), Automatic recognition of multi-word terms:. the c-value/nc-value method, International Journal on Digital Libraries, 3(2), 115–130.
4. *Jin Wei, Hung Hay Ho,* (2009), A novel lexicalized HMM-based learning framework for web opinion mining, in Proceedings of International Conference on Machine Learning (ICML-2009).
5. *Hofmann T.,* (1999), Probabilistic latent semantic indexing, in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, (pp. 50–57). ACM.

6. *Liu B.,* (2012), Sentiment analysis and opinion mining, Synthesis Lectures on Human Language Technologies, *5*(1), 1–167.
7. *Lossio-Ventura J. A., Jonquet C., Roche M., Teisseire M.,* (2013), Combining c-value and keyword extraction methods for biomedical terms extraction, In LBM'2013: 5th International Symposium on Languages in Biology and Medicine (pp. 45–49).
8. *Mei Qiaozhu, Xu Ling, Matthew Wondra, Hang Su, ChengXiang Zhai,* (2007), Topic sentiment mixture: modeling facets and opinions in weblogs, in Proceedings of International Conference on World Wide Web (WWW-2007).
9. *Niklas J., Gurevych I.,* (2010), Extracting Opinion Targets in a Single and Cross-Domain Setting with Conditional Random Fields, in Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2010).
10. *Nivre J., Hall J., Nilsson J., Chanev A., Eryigit G., Kübler S., Marsi E.,* (2007), MaltParser: A language-independent system for data-driven dependency parsing, Natural Language Engineering, *13*(02), 95–135.
11. *Pennington J., Socher R., Manning C. D.,* (2014), Glove: Global vectors for word representation, Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014), *12*.
12. *Poria S., Cambria E., Ku L. W., Gui C., Gelbukh A.,* (2014), A rule-based approach to aspect extraction from product reviews, SocialNLP 2014, 28.
13. *Popescu A. M., Etzioni O.,* (2007), Extracting product features and opinions from reviews, In Natural language processing and text mining (pp. 9–28), Springer London.
14. *Qiu G., Liu B., Bu J., Chen C.,* (2011), Opinion word expansion and target extraction through double propagation, Computational linguistics, 37(1), 9–27.
15. *Ratinov L., Roth D.,* (2009) Design challenges and misconceptions in named entity recognition, Proceedings of the Thirteenth Conference on Computational Natural Language Learning / Association for Computational Linguistics, pp. 147–155.
16. *Roeder K., Wasserman L.,* (1997), Practical Bayesian density estimation using mixtures of normals, Journal of the American Statistical Association, 92(439), 894–902.
17. *Scaffidi C., Bierhoff K., Chang E., Felker M., Ng H., Jin C.,* (2007), Red Opal: product-feature scoring from reviews, In Proceedings of the 8th ACM conference on Electronic commerce, pp. 182–191
18. *Turdakov D., Astrakhantsev N., Nedumov Y., Sysoev A., Andrianov I., Mayorov V., Fedorenko D., Korshunov A., Kuznetsov S.* (2014), Texterra: A Framework for Text Analysis, Proceedings of the Institute for System Programming of RAS [Trudy ISP RAN], volume 26, Issue 1, pp. 421–438.
19. *Yejin C., Cardie C.,* (2010), Hierarchical sequential learning for extracting opinions and their attributes, in Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL-2010).
20. *Zhang T., Johnson D.* (2003), A robust risk minimization based named entity recognition system, Proceedings of the seventh conference on Natural language learning at HLTNAACL 2003-Volume 4 / Association for Computational Linguistics, pp. 204–207.