Stance Detection in Russian: a Feature Selection and Machine Learning Based Approach

Sergey Vychegzhanin [000-0001-6456-7856] and Evgeny Kotelnikov [0000-0001-9745-1489]

Vyatka State University, Kirov, Russia
{vychegzhanin.sv, kotelnikov.ev}@gmail.com

Abstract. The huge scale and constant increase of the data volume in social media has led to a high demand for automatic means of such content analysis, specifically stance detection. This term stands for the task of assigning stance labels (“for” and “against”) with respect to a discussion topic.

In the paper we tackle stance detection for Russian texts from social network “VKontakte” with the use of machine learning methods – the support vector machine, *k*-nearest neighbors, Naïve Bayes, AdaBoost, and decision trees. Also we apply the Recursive Feature Elimination (RFE) algorithm for feature selection and explore the impact of morphological analysis on the quality of the task solution.

The best results (F_1=84.3%) are achieved by using of the SVM and vector model with relatively small set of normalized words chosen by RFE.

Keywords: stance detection, SVM, Recursive Feature Elimination, social media analysis

1 Introduction

Over the last 10–15 years Web 2.0 services and social media have shown huge growth, for example, in January 2017 the number of active accounts of Facebook was 1.87 billion, and the total number of social network users worldwide was more than 2.5 billion[1]. “User-generated content is the lifeblood of social media” [Obar, Wildman, 2015]. Such content, especially in text form, is a potential source of useful information for government agencies, commercial companies and individuals. The huge scale and constant increase of the data volume in social media has led to a high demand for automatic means of such content analysis [Zafarani et al., 2014].

The subject of this article is the stance detection, or stance classification, that is the task of assigning stance labels with respect to a discussion topic [Sridhar, 2015]. The main labels in this task are "for" (“support”, “pro”, “favor”) and “against” (“oppose”, “con”, “anti”). Also the labels "none" (“neither”), denoting the situation, when one cannot deduce stance from the text [Mohammad et al., 2016], and “observing”, indicating the repetition of the previous opinion [Ferreira, Vlachos, 2016] are used.

[1] https://www.statista.com.

The set of targets in relation to which the position is expressed can consist of one object, for example, “legalization of abortion” or “feminist movement” [Mohammad et al., 2016], pair of objects, for example, “iPhone vs. Blackberry” [Somasundaran, Wiebe, 2009] or more objects, for example, “left, right and other political orientations” [Malouf, Mullen, 2008].

The spectrum of areas of objects is very wide [Anand et al, 2011; Mohammad et al., 2016]: politics (“communism vs. capitalism”, “Donald Trump”), religion (“God's existence”), socially significant topics (“climate change”, “death penalty”), products (“Firefox vs. Internet Explorer”, “Mac vs. PC”) and even games and entertainment (“Superman vs. Batman”, “cats vs. dogs”).

The data sources are: congressional floor debates [Thomas et al., 2006; Burfoot et al., 2011], discussion forums [Somasundaran, Wiebe, 2009; Walker et al., 2012], social networks such as Twitter [Rajadesingan, Liu, 2014; Sobhani et al., 2016], online news articles [Ferreira, Vlachos, 2016] and comments [Sobhani et al., 2015].

Stance detection can be done at the author level, when it is considered that the author’s position does not change during the discussion, and at the document (user post) level, when it is assumed that the author’s position may change [Sridhar et al., 2015].

Stance detection can be applied in information retrieval, text summarization, recommendation systems, targeted advertising, political polling, product reviews, and fact checking [Mohammad, 2015; Sridhar et al., 2015; Elfardy et al., 2015; Ferreira, Vlachos, 2016].

Systems that try to determine automatically the position of the author of the text, face a number of difficulties:

- expression in the same text (post) or in different texts of the same author of the opposite positions with respect to the object (sides). Authors often give arguments in favor of their position and against another position, sometimes in one post. Sometimes even the authors can recognize in some way the rightness of the opponent, without sharing his position as a whole [Somasundaran et al., 2009];
- the author's position may change during the discussion [Sridhar et al., 2015];
- the target object may not be mentioned in the text [Mohammad, 2015];
- authors with different positions use the same or similar lexicon [Agrawal et al., 2003];
- a difficult language for analysis – comparisons, irony, sarcasm and other rhetorical devices [Malouf, Mullen, 2008].

Stance detection is closely related to sentiment analysis, argumentation mining and argument-based opinion mining, but does not coincide with them. Sentiment analysis refers to the task of automatically determining the polarity of a given text, whether it is positive, negative, or neutral [Mohammad, 2015]. The author's position and his sentiment in relation to the same object may not coincide. For example, in the sentence “*I do not like the party Yabloko, but I will vote for it*,” negative polarity is expressed, but the position is “for”. It turns out that the share of such sentences is quite high. For instance, in a dataset of tweets manually annotated for stance and sentiment [Sobhani et al., 2016] the share of negative tweets in which the position “for” is expressed is 12.8%, and the share of positive tweets with the position “against” is

13.8%. Another difference is that sentiment is changing more dynamically than stance, which is usually quite stable [Elfardy et al., 2015].

There is also a link between stance detection and aspect-based sentiment analysis [Liu, 2012]. Authors can evaluate not the object as a whole, but its individual aspects, expressing different positions regarding them [Somasundaran et al., 2009]. Thus, differentiating aspects can help in determining the position of the author.

Argumentation mining is a research area involved with the automatic extraction of argumentation structure from text [Moens et al., 2007]. Argument-based opinion mining aims to determine the arguments on which the users base their stance without recovering the argumentation structure [Boltužić, Šnajder, 2014]. Methods from both areas can be useful for solving the problem of stance detection.

In our work we studied the users' messages from Internet forums and the Russian social network "VKontakte"[2], in which the position for or against children vaccinations was expressed. Each post was regarded separately, so it was considered that the position of the author does not change.

2 Previous work

In the works devoted to stance detection, two main approaches can be distinguished:

1) the approach, which takes into account the discourse and the links between the posts (utterances), usually using graph-based methods [Agrawal et al., 2003; Thomas et al., 2006; Malouf, Mullen, 2008; Anand et al., 2011; Walker et al. 2012; Hasan, Ng, 2013];

2) the approach, considering each post in isolation [Somasundaran et al., 2009; Sobhani et al., 2015; Mohammad et al., 2016].

One of the first works, devoted to stance detection, was the article by Agrawal et al. [2003]. It considered the newsgroups discussions and the authors were divided into opposite camps based on the analysis of the link structure of the interactions' network and finding a maximum cut in the graph. The texts of the posts were not taken into account. An interesting peculiarity of many newsgroups is that people are more likely to respond to a message when they do not agree, than when they agree.

Thomas et al. [2006] explore transcripts of U.S. Congressional floor debates. The linear Support Vector Machine (SVM) with unigrams as features is used for classification. It is shown that considering same-speaker and different-speaker agreement information slightly improves the accuracy of the analysis.

Malouf and Mullen [2008] analyze U.S. informal political discussion posts. For stance detection they use approaches from sentiment analysis – PMI-IR [Turney, 2002] and from text classification – Naïve Bayes [McCallum, 1996]. They also use the approach based on the co-citation graph, for which the singular value decomposition followed by hierarchical clustering is performed. Cluster classes are defined based on the Naïve Bayes classifier. This approach turns out to be more accurate than straightforward Naïve Bayes.

[2] https://vk.com.

Anand et al. [2011] examine stance detection on posts across 14 various topics, such as “abortion”, “marijuana legalization” and “cats vs. dogs”. As characteristics they use n-grams (unigrams and bigrams), post's statistics, repeated punctuations, syntactic dependencies, and the set of context features computed for the immediately preceding post. Classification is performed on the basis of Naïve Bayes and rule-based classifier. The results of the analysis with considering the contextual characteristics and without them, are mixed.

Walker et al. [2012] use MaxCut over graph that represents the dialogic relations of agreement between speakers. In comparison with other classifiers (rule-based, Naïve Bayes and the SVM), the MaxCut-based algorithm shows some superiority.

In other works the discourse links between posts are not taken into account. For example, Somasundaran and Wiebe [2009] use an unsupervised approach, mining from web the polarity-target pairs, computing conditional probabilities with respect to topics. Overall stance of the post is computed with Integer Linear Programming.

Sobhani et al. [2015] first implement Non-Negative Matrix Factorization for text clustering, then make manual labelling by argument tags based on top keywords. Received argument tags are used in the SVM for stance classification.

In 2016, within the International Workshop on Semantic Evaluation (SemEval), a competition of systems for stance detection was held [Mohammad et al., 2016]. English tweets for stance towards the following six targets: “atheism”, “climate change is a real concern”, “feminist movement”, “Hillary Clinton”, “legalization of abortion”, and “Donald Trump” were studied. The winner was the baseline system of organizers based on the SVM and word and character n-grams (F_1=68.98%). The first place among the participants was taken by the MITRE system (F_1=67.82%), based on recurrent neural networks and word embeddings.

Hasan and Ng [2013] used as datasets the debate posts on four topics: “abortion”, “gay rights”, “Obama”, and “marijuana”. Three types of models were used: binary classifiers (Naïve Bayes and SVM), sequence classifiers (Hidden Markov Models and linear-chain Conditional Random Fields), and fine-grained models, jointly determine the stance label of a debate post and the stance label of each of its sentences. One of the results was the inability to identify a clear leader between Naïve Bayes and the SVM.

In our work each post is considered in isolation; for classification a vector model with feature selection and several supervised classifiers are used, including the SVM, Naïve Bayes, k-nearest neighbors, AdaBoost and decision trees. To the best of our knowledge, this is the first work in which the stance detection task for Russian is solved and corresponding labelled corpus is created. The main goal of this research is an evaluation of machine learning performance for stance detection in Russian.

3 Text corpus

The text corpus is formed from Russian-language messages of users of Internet forums and social network “VKontakte”. To create the text corpus we used the messages containing opinions of users on the topic "Vaccinations for children". In the process of labeling each message was assigned with one of the following labels:

- *for*, if the author supports vaccinations;
- *against*, if the author is against vaccinations;
- *conflict*, if the author is a supporter of some vaccinations and an opponent of others;
- *none*, if on the basis of the message it is difficult to conclude whether the author is a supporter or an opponent of vaccinations.

Of the 22,000 analyzed texts, 1000 messages containing *for* and *against* labels were selected to the corpus. The labelling of corpus was made by three annotators. We used Fleiss' kappa statistical measure [Fleiss, 1971] to evaluate the inter-annotator agreement. Its value is equal to 0.87 that confirms the high quality of the labelling.

Statistical characteristics of the text corpus are presented in Table 1.

Table 1. Characteristics of the text corpus

Label	Number of texts	Total amount of words	Average text length, words	Lexicon size (lemmas)
for	500	35 326	70	4 108
against	500	34 167	68	3 992
for & against	1 000	69 493	69	6 007

Below the examples from the text corpus are given for each class of texts. User messages from the *for* class (with author spelling and punctuation preserved):

Example 1. "Privivaju. Immunolog podruga sem'i, tak chto problem net. Opjat' zhe na moj vzgljad, esli pridumali privivki, to ne prosto tak" ("I'm making vaccinations. The immunologist is the friend of the family, so there are no problems. Again, in my opinion, if they came up with vaccinations, it's not just that").

Example 2. "Nuzhno ee protivnikov svodit' v infekcionku na jekskursiju!" ("It is necessary to take its opponents to isolation hospital!").

User messages from the *against* class:

Example 3. "Mne tozhe plevat' na mnenija vrachej, chto nado delat' privivki. Nado nadejat'sja ne na vrachej a verit' v Boga i vsjo budet horosho." ("I couldn't care less about doctors' opinions. Don't rely on doctors but trust in God and everything will be all right").

Example 4. “A chto vas ostanavlivaet ne delat' privivki? Ne vizhu v nih nikakogo smysla, i ne ponimaju zachem moemu rebenku nuzhno vvodit' vsjakuju gadost' i podryvat' ego immunitet v mladencheskom vozraste!” (“And what stops you from not getting vaccinated? I do not see any sense in them, and I do not understand why someone needs to inject into my child any muck and undermine his immunity in infancy!”).

4 Results and discussion

4.1 Experiments’ design

The solution of the stance detection problem was accomplished using five methods of machine learning: Support Vector Machine (SVM), *k*-Nearest Neighbors (kNN), Naïve Bayes (NB), AdaBoost (AB), Decision Trees (DT). To optimize the parameters of the classifiers a five-fold nested cross-validation [Cawley, 2010] procedure was applied and to obtain objective estimates of the classification quality a five-fold cross-validation procedure was applied. Classifiers with the following parameters were used:

- SVM with linear, RBF and polynomial kernels, regularization coefficient $C = 10^p$, $p = [-1, 0, \dots, 6]$, gamma = 10^q, $q = [-6, -5, \dots, -1]$ and gamma = 1/n_features, where n_features – number of features;
- kNN with numbers of neighbors $k = [1..20]$;
- NB with multinomial distribution;
- AB with decision tree as base classifier;
- DT with the maximum depth of the tree: *max_depth* = [1..20].

The experiments were carried out using the machine learning library *scikit-learn* [Pedregosa et al., 2011]. To represent texts a vector space model was used. Each text was represented as an *n*-dimensional binary vector, the components of which characterized the presence or absence of the corresponding word in the text [Manning et al., 2008]. The set of words considered in the vector model is a dictionary of this model.

In the most of experiments, the words from the texts were transformed to the lemmas. Morphological analysis of the texts was carried out using the MyStem tool [Segalovich, 2003].

In total, five experiments were carried out, differing in the composition of the vector model dictionary:

- *Test1* – the dictionary was made up of all the words of the text corpus without transforming them to the lemmas;
- *Test2* – the dictionary was composed of all the words of the text corpus, transformed to the lemmas;
- *Test3* – the dictionary was made up of nouns, adjectives, verbs, adverbs and interjections, transformed to the lemmas;
- *Test4* – the dictionary from *Test3* with the added words “za” (“*favor”*) and “ne” (“*not*”); the word “protiv” (“*against*”) already exists in *Test3*;

- *Test5* – the dictionary was compiled on the basis of the dictionary from *Test2* using the Recursive Feature Elimination (RFE) with the five-fold cross-validation.

  RFE is a recursive repetition of following procedure [Guyon, 2002]:

1. Train the classifier (e.g., linear SVM).
2. Compute the ranking criterion for all features.
3. Remove the feature with the smallest ranking criterion.

The size of the dictionaries used in the experiments are shown in Table 2 (for *Test5* the values for each of five folds are given).

Table 2. The size of the dictionary

	Test1	Test2	Test3	Test4	Test5
Dictionary size	11 584	6 007	5 716	5 718	{ 661, 612, 468, 767, 959 } Average = 693

Two simple classifiers were used as baselines. The first classifier assigned to the text a label “against” if the word “against” was found in the text, and assigned a label “for” otherwise (BL1); the second classifier assigned to the text a label “for” if the word “for” was found, and assigned a label “against” otherwise (BL2).

4.2 Results

The quality of the classification was estimated using of macro F_1-measure. The values of the F_1-measure are in Table 3, the classifiers’ parameters, corresponding to the best values of the F_1-measure, are in Table 4.

Table 3. F_1-measure, % (*Test1 – Test5*)

No. of test	BL1	BL2	SVM	kNN	NB	AB	DT
Test1	55.1	57.8	**77.5**	64.0	75.8	74.9	67.4
Test2			76.5	60.0	**78.8**	73.2	63.0
Test3			72.2	55.8	**74.6**	67.1	60.7
Test4			76.0	61.8	**77.4**	72.4	64.7
Test5			**84.3**	68.6	80.0	74.1	66.6

Table 4. The classifiers’ parameters, corresponding to the best values of the F_1-measure

	SVM	kNN	DT
Test1	C = 100; kernel = rbf; gamma = 1/n_features	k = 6	max_depth = 16
Test2	C = 100; kernel = rbf; gamma = 1/n_features	k = 4	max_depth = 13
Test3	C = 10; kernel = rbf; gamma = 0.01	k = 3	max_depth = 16
Test4	C = 100; kernel = rbf; gamma = 1/n_features	k = 6	max_depth = 15
Test5	C = 100; kernel = rbf; gamma = 1/n_features	k =10	max_depth = 16

In *Test1* the SVM showed the best result, ahead of NB and AB by 1.7% and 2.6%, respectively. The reduction of the feature space by almost half (from 11 584 words to 6 007) due to the lemmatization of words (*Test2*) led to an improvement in quality for NB (from 75.8% to 78.8%) and a slight decrease for the remaining classifiers (from 1.0% for the SVM to 4.4% for DT).

The use of only nouns, adjectives, verbs, adverbs and interjections with lemmatization (*Test3*) reduced the quality for all classifiers (from 2.3% for DT to 6.1% for AdaBoost). This decline was due to the exclusion of the keywords «*favor*» and «*not*», as demonstrated by *Test4*, in which these words were added to *Test3*. The results of *Test4* were close to the results of *Test2* (the difference does not exceed 1.8%).

Thus, in experiments without feature selection (*Test1 – Test4*), a Naïve Bayes classifier with a multinomial distribution made it possible to obtain a better classification quality when using a dictionary composed of all lemmas of the text corpus (*Test2*: $F_1 = 78.8\%$). At the same time, the quality remains practically at the same level when only nouns, adjectives, verbs, adverbs and interjections are left, and the words «*favor*» and «*not*» (*Test4*: $F_1 = 77.4\%$).

The selection of the optimal dictionary size in *Test5* was based on 5-fold nested cross-validation using the RFE procedure and the SVM classifier with a linear kernel (C = 100) for each fold independently (see Table 2). Note that the Naïve Bayes classifier showed weaker results. Further the optimal dictionaries were used in each classifier to obtain F_1-measure quality scores; Table 3 shows the average F_1-measure for all five folds. It turned out that the application of the RFE procedure allows to improve significantly the quality of classification for classifiers the SVM, kNN and NB (by 6.8%, 4.6% and 4.1%, respectively). The difference between precision and recall for SVM and NB classifiers in this test does not exceed 1.0%. At that the AB and DT classifiers show the best quality on *Test1* with the maximum size of the feature space (11 584 words). Probably this is connected with the principle of building decision rules, underlying both DT and AB. However, their results on *Test5* only slightly differ from *Test1* (by 0.8%) with an average dictionary size of 693 words.

4.3 Analysis of the feature selection process

For the RFE-procedure and SVM-classifier an additional study was carried out: we constructed the dependence of the quality of the stance detection on the dictionary size, including the attributes with the best ranking criterion (Fig. 1). Note that this dependence is obtained for the test sets in the cross-validation procedure, therefore it cannot be used as a final rating of the qualifier quality. However, it allows you to draw certain conclusions about the feature space in the problem of stance detection.

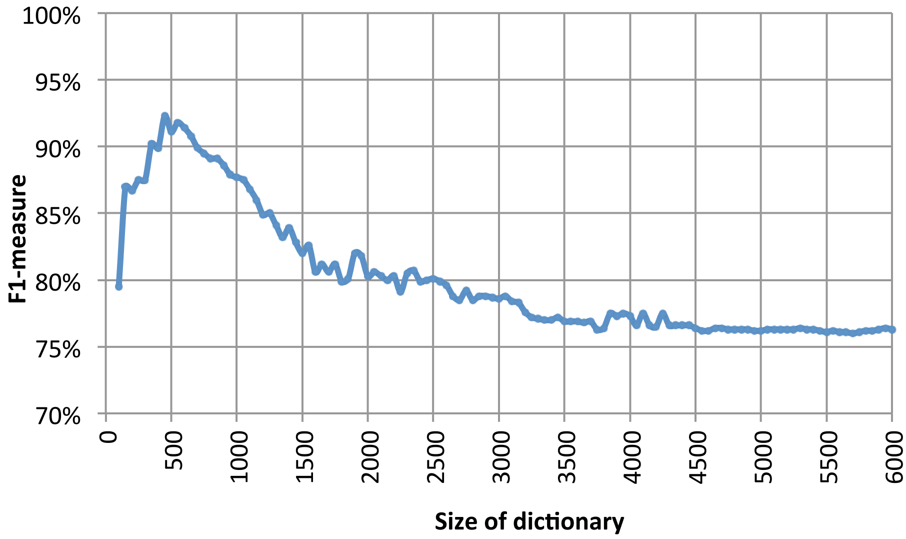

Fig. 1. Graph of the dependence of F_1-measure on the dictionary size

Fig. 1 shows that the maximum value of F_1-measure (92.3%) is achieved with the dictionary size of 450 words (we denote this dictionary as “optimal”). The words “convincing”, “schedule”, “for”, “bear (disease)”, “allow” have the highest weights in the “for” class, and the words “muck”, “evil”, “against”, “denial”, “cease” – in the “against” class.

Values of F_1-measure, greater than 90%, can be obtained with the number of features from 350 to 700. Further increase in the number of features leads to a lowering of the classification quality.

An analysis of the distribution of parts of speech in the optimal dictionary in comparison with the distribution of parts of speech in Russian National Corpus [RNC] was made (Table 5).

Table 5. Distribution of parts of speech in the optimal dictionary and in the Russian National Corpus

Part of speech	Optimal lexicon		Portion in RNC, %
	Count of words	Portion, %	
Nouns	148	32.9	28.5
Verbs	138	30.7	17.8
Adjectives	59	13.1	8.5
Adverbs	43	9.6	4.1
Others	62	13.8	41.1
Total	**450**	**100**	**100**

Table 5 shows that in the optimal dictionary the proportion of verbs, adjectives and adverbs is much larger than in Russian National Corpus. If it has long been known for adjectives and adverbs that they carry important information for the analysis of opinions [Turney, 2002], then the greater proportion of verbs relative to adjectives and adverbs in the optimal dictionary is somewhat surprising. For example, in the sentiment lexicon the proportion of verbs, adjectives and adverbs on average is, respectively, 22%, 37% and 16% [Kotelnikov, 2016]. At the same time there are only 12 (2 positive and 10 negative) verbs of the optimal dictionary, which are found in sentiment lexicons from [Kotelnikov, 2016].

Classifiers with dictionaries which contain 450 words were also tested, these dictionaries included words with maximum weights Term Frequency – Inverse Document Frequency (TF-IDF, *Test6*) [Jones, 2004] and Relevance Frequency (RF, *Test7*) [Lan et al., 2009] (See Table 6).

Table 6. F_1-measure, % (*Test5 – Test7*)

No. of test	SVM	kNN	NB	AB	DT
Test5 (RFE)	**84.3**	68.6	80.0	74.1	66.6
Test6 (TF-IDF)	77.2	65.0	**77.7**	73.1	63.3
Test7 (RF)	76.9	66.8	**81.6**	67.8	61.6

Table 6 shows that both TF-IDF and RF weighting methods reduce the quality for all classifiers as compared to RFE, except for the RF + NB bundle. But even in the latter case F_1-measure turns out to be lower than for RFE + SVM.

Thus, the feature selection based on the RFE procedure can significantly improve the quality of the classification, compared to other methods of selecting features, but it is very resource intensive: in our experiments with 5-fold cross-validation, the RFE procedure took 7.5 hours on an ordinary desktop computer.

5 Conclusion

The task of stance detection is important from the point of view of social media analysis. Our work showed that the traditional models and methods of machine learning and text classification, as well as the feature selection procedures, allow to obtain a sufficiently high quality of the analysis, about 85%. This quality is achieved by using the SVM-classifier with RBF-kernel, Recursive Feature Elimination procedure with words' lemmas as attributes. Multinomial Naïve Bayes classifier also has high marks (here our results coincide with the work [Hasan and Ng, 2013]), but nevertheless, in general, it is slightly worse than the SVM.

Another result of our work is the usefulness of using verbs as features along with nouns, adjectives and adverbs; negation «not», as well as specific words for this task – «favor» and «against».

In the future, we plan for the stance detection task on the one hand to use deep learning approaches such as distributed representations of words, on the other hand, to apply sequential labeling methods, such as conditional random fields.

Acknowledgments. This work was carried out as a part of the project «Research and Development of Sentiment Lexicons for Text Sentiment Analysis» of the Government Order No. 34.2092.2017/4.6 of the Ministry of Education and Science of the Russian Federation (2017-2019).

References

1. Agrawal, R., Rajagopalan, S., Srikant, R., Xu, Y.: Mining Newsgroups Using Networks Arising from Social Behavior. In: 12th International Conference on World Wide Web (WWW 2003), Budapest, pp. 529–535 (2003).
2. Anand, P., Walker, M., Abbott, R., Fox Tree, J.E., Bowmani, R., Minor, M.: Cats Rule and Dogs Drool!: Classifying Stance in Online Debate. In: 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, ACL-HLT 2011, Portland, Oregon, pp. 1–9 (2011).
3. Boltužić, F., Šnajder, J.: Back up your Stance: Recognizing Arguments in Online Discussions. In: First Workshop on Argumentation Mining, Baltimore, Maryland, pp. 49–58 (2014).
4. Burfoot, C., Bird, S., Baldwin, T.: Collective Classification of Congressional Floor-Debate Transcripts. In: 49th Annual Meeting of the Association for Computational Linguistics, Portland, Oregon, pp. 1506–1515 (2011).
5. Cawley, G.C., Talbot, N.L.C.: On over-fitting in model selection and subsequent selection bias in performance evaluation. JMLR, 11, 2079–2107 (2010).
6. Elfardy, H., Diab, M., Callison-Burch, C.: Ideological Perspective Detection Using Semantic Features. In: Fourth Joint Conference on Lexical and Computational Semantics (*SEM 2015), Denver, Colorado, pp. 137–146 (2015).
7. Ferreira, W., Vlachos, A.: Emergent: a novel data-set for stance classification. In: NAACL-HLT 2016, San Diego, California, pp. 1163–1168 (2016).
8. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Mach. Learn., 46 (1-3), 389–422 (2002).
9. Fleiss J. L.: Measuring nominal scale agreement among many raters. Psychological Bulletin, 76(5), pp. 378–382 (1971).
10. Hasan, K.S., Ng, V.: Stance Classification of Ideological Debates: Data, Models, Features, and Constraints. In: International Joint Conference on Natural Language Processing, Nagoya, pp. 1348–1356 (2013).
11. Jones, K. S.: A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation, 60(5), 493-502 (2004).
12. Kotelnikov, E.V., Bushmeleva, N.A., Razova, E.V., Peskisheva, T.A., Pletneva, M.V.: Manually Created Sentiment Lexicons: Research and Development. In: Computational Linguistics and Intellectual Technologies, 15(22), 281–295 (2016).
13. Lan, M., Tan, C. L., Su, J., Lu, Y.: Supervised and Traditional Term Weighting Methods for Automatic Text Categorization. In: IEEE Trans. on Pattern Analysis and Machine Intelligence, 31(4), 721–735 (2009).
14. Liu, B.: Sentiment analysis and opinion mining. In: Synthesis lectures on human language technologies, 5(1) (2013).
15. Malouf, R., Mullen, T.: Taking sides: User classification for informal online political discourse. In: Internet Research, 18, pp. 177–190 (2008).
16. Manning, C. D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval, Cambridge University Press, NY (2008).

17. McCallum, A.K.: Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering (1996). http://www.cs.cmu.edu/~mccallum/bow.
18. Moens, M.-F., Boiy, E., Palau, R.M., Reed, C.: Automatic detection of arguments in legal texts. In: 11th International Conference on Artificial Intelligence and Law, Palo Alto, California, pp. 225–230 (2007).
19. Mohammad, S.M.: Sentiment analysis: detecting valence, emotions, and other affectual states from text. In: Emotion Measurement (2015).
20. Mohammad, S.M., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: SemEval-2016 Task 6: Detecting Stance in Tweets, Proceedings of SemEval-2016, San Diego, California, pp. 31–41 (2016).
21. Obar, J.A., Wildman, S.: Social media definition and the governance challenge: An introduction to the special issue. Telecommunications policy, 39 (9), pp. 745–750 (2015).
22. Pedregosa, F. et al.: Scikit-learn: Machine Learning in Python. JMLR, 12, pp. 2825-2830 (2011).
23. Rajadesingan, A., Liu, H.: Identifying Users with Opposing Opinions in Twitter Debates. In: SBP 2014, LNCS 8393, pp. 153–160 (2014).
24. RNC: Russian National Corpus (http://ruscorpora.ru).
25. Segalovich, I.: A Fast Morphological Algorithm with Unknown Word Guessing Induced by a Dictionary for a Web Search Engine. In: MLMTA-2003, Las-Vegas (2003).
26. Sobhani, P., Inkpen, D., Matwin, S.: From Argumentation Mining to Stance Classification. In: 2nd Workshop on Argumentation Mining, Denver, Colorado, pp. 67–77 (2015).
27. Sobhani, P., Mohammad, S.M., Kiritchenko, S.: Detecting Stance in Tweets and Analyzing its Interaction with Sentiment. In: Fifth Joint Conference on Lexical and Computational Semantics (*SEM 2016), Berlin, pp. 159–169 (2016).
28. Somasundaran, S., Wiebe, J.: Recognizing Stances in Online Debates. In: 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, Singapore, pp. 226–234 (2009).
29. Sridhar, D., Foulds, J., Huang, B., Getoor, L., Walker, M.: Joint Models of Disagreement and Stance in Online Debate, Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, pp. 116–125 (2015).
30. Thomas, M., Pang, B., Lee, L.: Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In: EMNLP, Sydney, pp. 327–335 (2006).
31. Turney, P.D.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In: 40th Annual Meeting of the ACL, Philadelphia, Pennsylvania, pp. 417–424 (2002).
32. Walker, M.A., Anand, P., Abbott, R., Grant, R.: Stance Classification using Dialogic Properties of Persuasion. In: Conference of the North American Chapter of the ACL: Human Language Technologies, Montreal, pp. 592–596 (2012).
33. Zafarani, R., Abbasi, M.A., Liu, H.: Social Media Mining: An Introduction, Cambridge University Press (2014).