Automatic Term Extraction for Sentiment Classification of Dynamically Updated Text Collections into Three Classes

Yuliya Rubtsova

The A.P. Ershov Institute of Informatics Systems (IIS),
Siberian Branch of the Russian Academy of Sciences
yu.rubtsova@gmail.com

Abstract. This paper presents an automatic term extraction approach for building a vocabulary that is constantly updated. A prepared dictionary is used for sentiment classification into three classes (positive, neutral, negative). In addition, the results of sentiment classification are described and the accuracy of methods based on various weighting schemes is compared. The paper also demonstrates the computational complexity of generating representations for N dynamic documents depending on the weighting scheme used.

Keywords: Corpus linguistics, sentiment analysis, information extraction, text classification and categorization, social networks data analysis.

1 Introduction

We live in a constantly changing world. Peoples' style and way of life, behaviors and speech are all changing. Natural language is constantly transforming and developing together with conversational speech: new words are included in active vocabulary, while old ones cease to be used. New words are born every day, and about half of them are slang. Slang responds to changes in all spheres more quickly than other types of language and is so important to modern society that last year 40 neologisms, some of which are slang words, were added to the Oxford English Dictionary. Slang is actively used in colloquial speech and written communication on social networking sites, as well as to express an emotional attitude towards a particular issue. Users of social networks are among the first to start using new terms in everyday language. Among about 1000 new words included in the Oxford English Dictionary near 40 were terms that came from social networks, such as "srsly", "me time" and "selfie". Accordingly, it is necessary to consider slang when developing sentiment classifiers, in particular when creating vocabularies of emotional language. Moreover, since active vocabulary is regularly updated with new terms, vocabularies of emotional language should also be updated regularly, and the weights of the terms in these vocabularies must be recalculated.

This paper presents an approach to extracting terms and assigning them weights in order to build a vocabulary of emotional language that is constantly updated.

P. Klinov and D. Mouromtsev (Eds.): KESW 2014, CCIS 468, pp. 140–149, 2014.

There will be a comparison of methods based on various weighting schemes and the computational complexity of recalculating the weights of terms in the vocabulary depending on the methods used will be demonstrated. All experiments to classify texts into three sentiment classes (positive, neutral, negative) were performed on two collections:

- Collection of short posts from microblogs [1];
- News collection.

2 Overview of Term Weighting Schemes

There are different approaches to the extraction of evaluative words from texts and the determination of their weight in the collection. In [2], the authors use a thesaurus to expand a vocabulary of evaluative words that had been collected manually. In corpus linguistics, methods of extracting terms based on measuring the relevance of a term to a collection are widely used, for example, the well-known methods based on the TF-IDF weighting scheme [3]. In [4], the authors show that variants of the classic TF-IDF scheme adapted to sentiment analysis task provide significant increases in accuracy in comparison to binary unigram weights. They tested their approach on a wide selection of data sets and demonstrated that classification accuracy enhanced.

The functioning of most existing methods of automatic and semi-automatic word extraction from texts are based on the assumption that all the data are known in advance, accessible and static. For example, to use a method based on the TF-IDF scheme [3], it is necessary to know the frequency each term occurs in the document, which means that the data set should not be changed during calculation. This greatly complicates computation is required for data calculation in real time. For example, when adding a new text to the collection, it is necessary to recalculate the weights for all terms in the collection. The computational complexity of recalculating all the weights in the collection is $O(N^2)$.

The Term Frequency – Inverse Corpus Frequency (TF-ICF) measure has been proposed [5, 6] in order to solve the problem of searching for terms and calculating their weights in real time. Information on the usage frequency of a term in other documents of the collection is not required in order to calculate TF-ICF, so the computational complexity is linear. The results of methods based on TF-ICF and TF-IDF have been compared [3] in order to evaluate the effectiveness of a method based on the TF-ICF weighting scheme for the task of extracting evaluative terms for a vocabulary of emotional language.

The formula for calculating the TF-IDF measure is as follows:

$$tfidf = tf \times log \frac{T}{T(t_i)} \quad (1)$$

Where tf is the frequency with which the term occurs in the collection (of positive or negative tweets), T is total number of texts in the positive and negative collections, and $T(t_i)$ is the number of texts in the positive and negative collections containing the term.

The formula for calculating the TF-ICF measure is as follows:

$$tf.icf = tf \times \log\left(1 + \frac{|C|}{cf(t_i)}\right) \quad (2)$$

Where C is the number of categories and cf is the number of categories in which the term to be weighed occurs.

In the TF-IDF scheme, both weighing factors assess the term at the document level. The proposed TF-ICF scheme is mixed: TF evaluates the term at the document level, ICF at the category level. Another approach for TF-ICF was suggested by Lertnatteed [7, 8], he proposed the use of a TF-ICF scheme in which the TF factor evaluates term frequencies at the category level, as would the ICF factor [6].

To test the effectiveness of approaches based on the selected weighting schemes, we proceed as in [9], taking 5 terms from a real corpus and evaluating the calculation of the term's weight depending on the collection it belongs to (positive, negative, neutral). The selected terms are as follows: "obidno" (it's a shame), "plokho" (bad), "lyublyu" (I love), "konechno" (of course) and "vremya" (time). Based on the frequency of the term usage in a collections suppose that first two terms belong to the class of negative posts, the following two – the positive class; the latter term is neutral and occurs equally often in the positive and negative collections. Tables 1-3 indicate the weight of each term depending on the method used and the collection it belongs to.

Table 1. A practical example of applying the methods for the category of positive tweets

Term	Frequency	tf-idf	tf-icf
Obidno (it's a shame)	55	0.000109944	0.00000871077
Plokho (bad)	424	0.000772331	0.0000671521
Lyublyu (I love)	2517	0.004197502	0.000398636
Konechno (of course)	1070	0.001950132	0.000169464
Vremya (time)	1313	0.002186481	0.00020795

Table 2. A practical example of applying the methods for the category of negative tweets

Term	Frequency	tf-idf	tf-icf
Obidno (it's a shame)	844	0.001687134	0.000133671
Plokho (bad)	1448	0.002637583	0.000229331
Lyublyu (I love)	1391	0.002319716	0.000220303
Konechno (of course)	665	0.001211998	0.000105321
Vremya (time)	1377	0.002293057	0.000218086

Table 3. A practical example of applying the methods for the category of neutral tweets

Term	Frequency	tf-idf	tf-icf
Obidno (it's a shame)	32	0.0000639672	0.00000506808
Plokho (bad)	152	0.000276873	0.0000240734
Lyublyu (I love)	61	0.000101727	0.00000966103
Konechno (of course)	280	0.000510315	0.0000443457
Vremya (time)	1321	0.002199803	0.000209217

Although the method based on the idf scheme ignores the category a term belongs to, and the weight values for positive, neutral and negative terms in the idf column should be identical, adding tf causes there to be a difference in this column.

As a result, the test sample shows that methods based on TF-IDF and TF-ICF schemes give similar results on static collections. This means that both methods attribute the word "bad" to the negative category, and the word "love" to the positive category, not vice versa. An analogous experiment, which showed similar results, was conducted to calculate and compare the weight of parts of speech for the three collections. That means we can expect accuracy result using methods based on TF-ICF for sentiment classification.

3 Corpora Characteristics

3.1 Short Text Corpus

In a previous paper [1], the author describes an approach to building a Russian-language corpus of short texts based on posts from social network Twitter. Twitter is a social networking and microblogging service that allows users to write messages in real time. Often, tweets are directly posted from a mobile device at the place where the event is taking place, which adds emotion to posts. Due to the platform's limit, the length of a post on Twitter may not exceed 140 characters. In connection with this aspect of the service (short posts, which are published in real time, possibly using mobile devices), people use abbreviations, shorten words, use emoticons, and make spelling mistakes and typing errors. As Twitter has the features of a social network, users are able to express their opinion on a variety of issues, ranging from the quality of cellphones to international economic and political developments. This is why the Twitter platform has attracted the attention of researchers.

There are no publicly available prepared general-topic corpora of short texts in Russian, which is why the stream Twitter API was used to assemble a collection consisting of about 15 million short posts. The corpus was put together over several weeks in late 2013 and early 2014.

The method described in [10] showed the effectiveness of using emoticons (special symbols denoting emotions in written communications), for the automatic text classification into positive and negative classes. The emotion of the post can be determined with high accuracy if the author included a symbol that designates emotion.

For this reason, vocabularies of characters representing the positive or negative attitude of the author were constructed. For example, the icon :) stands for a positive emotion, :(a negative one. Since the length of a post is limited to 140 characters, it was assumed that an emoticon used to express emotion refers to the whole post, and not just a part of it.

Posts with positive and negative sentiments were searched for in accordance with the written symbols for emotions and two collections were formed. These collections will be used for further analysis of posts with positive and negative sentiments and the identification of patterns in positive and negative posts.

To form a collection of neutral posts were taken text from news microblogging accounts.

Filtering [1] was carried out to maintain experimental integrity:

- Texts containing both positive and negative emotions were deleted from the collection. Such texts cannot be automatically attributed to either collection of posts (positive or negative).
- Not informative tweets (less than 40 characters long) were deleted.

On the basis of raw collection using a method [10] and the filtration proposed by the author [1] was formed a balanced corpus, comprising the following collections:

- collection of positive posts – 114 991 entries;
- collection of negative posts – 111 923 entries;
- collection of neutral posts – 107 990 entries.

The ratio of word forms and unigrams in the collections is shown in Table 4. The corpus is publicly available [11].

Table 4. The ratio of unigrams and unique unigrams in the short text collections

Type of collection	Number of unigrams in the collection	Number of unique unigrams in the collection
Positive posts	1 559 176	150 720
Negative posts	1 445 517	191 677
Neutral posts	1 852 995	105 239

3.2 News Corpus

News collections were assembled on news websites. Experts manually tagged the corpus by positive, neutral and negative collections. The difference between the news collection and the short texts collection are that news items are less emotional, their vocabulary is more neutral and there are few slang words, abbreviations and obscene expressions. Typically, news texts do not contain spelling errors. News texts do not contain symbols that denote emotions in a written form (emoticons). News texts are significantly longer than 140 characters.

The corpus of news texts consists of the following collections:

- collection of positive documents that consists of 22 976 news items;
- collection of negative documents that consists of 21 592 news items;
- collection of neutral documents that consists of 22 381 news items.

The ratio of word forms and unigrams in the collections is presented in Table 5.

Table 5. The ratio of unigrams and unique unigrams in the news collections

Type of collection	Number of unigrams in the collection	Number of unique unigrams in the collection
Positive news items	4 553 010	104 001
Negative news items	10 400 699	202 354
Neutral news items	7 667 441	155 538

4 Preparation for the Experiment

4.1 Preparation Texts for the Classifier

Before using the text classifier, the texts must be converted to vector format. That's why the collection of short texts, as described in 3.1, was subjected to filtration. In order to produce an emotive vocabulary, the following were filtered out from the collection:

- Punctuation – commas, colons, quotation marks (exclamation marks, question marks and ellipses were retained);
- References to significant personalities and events – the attitude towards them may vary over time, but a classifier trained on "old texts" will not be able to adapt quickly;
- Proper names;
- Numerals (references to years and time were retained);
- All links were replaced with the word "Link" and were taken into consideration as a whole.

The final vocabulary contains 21 481 words.

Using formulas 1 and 2, the weights of each word in the vocabulary were calculated and stored for the corpus of short texts.

4.2 The Classifier

The proven [12, 13] support vector method was used to classify text into three classes. Since computational complexity of LibSVM [14] is rather high on large amount of sparse vectors, the LibLinear library [15] – a modification of the LibSVM algorithm with a linear kernel – was used for classification.

4.3 Quality Assessment

Accuracy, precision, recall and F-measure were selected as measures to evaluate the classification of texts into three classes.

Accuracy rate is the percentage of test set samples that are correctly classified.

Precision and recall were calculated using a confusion matrix. For example, the confusion matrix for the collection of short texts using a weight calculation method based on TF-IDF is represented in Figure 1. The dimension of the matrix corresponds to the number of classes for classification – and is equal to three. The columns of the matrix are reserved for expert solutions, the rows for the classifier's solutions. When we classify a document from the test sample, we increment the number at the intersection of the row of the class returned by the classifier and the column of the class to which the document really belongs.

Twitter TF-IDF				
	0.958	0.965	0.987	0.923
0.955		-1	0	1
0.976	-1	21855	71	455
0.908	0	547	19616	1429
0.981	1	251	194	22531

Fig. 1. Confusion matrix for the collection of short texts. The weights of words were calculated using the method based on the TF-IDF scheme.

Precision (3) is equal to the ratio of the corresponding diagonal element in the matrix and the sum of the entire row of the class. Recall (4) is the ratio of the diagonal element in the matrix and the sum of the entire column of the class. Formally:

$$Precision_x = \frac{A_{x,x}}{\sum_{i=1}^{n} A_{x,i}} \tag{3}$$

$$Recall_x = \frac{A_{x,x}}{\sum_{i=1}^{n} A_{i,x}} \tag{4}$$

The F-measure is the harmonic mean of precision and recall. If the precision or recall tend to zero, it tends to zero. F-measure is calculated according to the formula (5):

$$F = 2\frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

5 Results of the Experiment

Several experiments were conducted on two different datasets in order to compare the precision of the classifier depending on the selected method to determine the weight

of a term in a collection. Text collections are constantly updated therefore it is necessary to constantly update the dictionary, and recalculate the weights of the terms in the dictionary.

The first experiment was conducted on the short text corpus, for which it was randomly divided into training and test collection. The ratio of positive, neutral and negative texts was preserved in the training (267 924 documents) and test collections (66 980 documents). The results are shown in Table 6.

Table 6. Comparison of TF-IDF and TF-RF for a collection of short texts

	TF-IDF	TF-ICF
accuracy	95.5981	95.0664
Precision	0.955204837	0.94984672
Recall	0.958092631	0.953112184
F-measure	0.956646554	0.95147665

There is insignificance of the difference between the two methods' results when applied to the short text corpus due to the data sparseness. Despite the fact that the precision of the method based on the TF-ICF scheme is lower, it is evident from the table that this error is negligible. Therefore, methods based on the TF-ICF scheme may be applied to calculate weights in dynamically updated collections of short texts.

A similar experiment was carried out on longer texts – the collection of news items. The news collection was also divided into training (111 214 documents) and test collections (27 802 documents). The experiment showed that methods based on the TF-ICF scheme show significantly worse results on long texts than those based on TF-IDF. The results are shown in Table 7.

Table 7. Comparison of TF-IDF and TF-RF for the collection of news items

	TF-IDF	TF-ICF
accuracy	69.8619	58.1397
Precision	0.698624505	0.581402868
Recall	0.709246342	0.61278022
F-measure	0.703895355	0.596679322

6 Conclusion

This paper shows an approach to automatically constructing vocabularies of emotional language. A vocabulary is based on prepared collections and is general-topic, i.e. does not belong to any predetermined domain. The weights in the vocabularies are calculated using methods based on two weighting schemes. The computational complexity of the methods for updating a collection by adding new posts has been determined. In contrast to methods based on recalculating the weights of every term in the collection, the computational complexity of the method based on TF-ICF is linear.

Despite the fact that the precision of classifying long texts into three classes is significantly reduced when using methods based on TF-ICF, the precision of short text classification is only slightly reduced.

The software module obtained as a result of this paper makes it possible to dynamically update a vocabulary of emotive language, monitor and record lexical changes over time, and add new terms to the active vocabulary and recalculate the weight of these terms depending on the collection that they belong to.

The further prospect of this paper include the use of N-grams and morphological tagging on the collection of news items in order to reduce the difference between methods based on the TF-IDF and TF-ICF schemes, as well as to increase the accuracy of text classification into three classes: positive, neutral, negative.

References

1. Rubtsova, Y.: A method for development and analysis of short text corpus for the review classification task. In: Proceedings of Conferences Digital Libraries: Advanced Methods and Technologies, Digital Collections, RCDL 2013, pp. 269–275 (2013)
2. Hu, M., Liu, B.: Mining and Summarizing Customer Reviews. In: KDD 2004, Seattle, pp. 168–177 (2004)
3. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Journal of Information Processing and management 24(5), 513–523 (1988)
4. Paltoglou, G., Thelwall, M.: A study of information retrieval weighting schemes for sentiment analysis. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, July 11-16, pp. 1386–1395 (2010)
5. Jones, K.S.: A Statistical Interpretation of Term Specificity and Its Application in Retrieval. J. Documentation 28(1), 11–21 (1972)
6. Reed, J., Jiao, Y., Potok, T.: TF-ICF: A new term weighting scheme for clustering dynamic data streams. In: Proceedings of the 5th International Conference on Machine Learning and Applications, USA, pp. 258–263 (2006)
7. Lertnattee, V., Theeramunkong, T.: Analysis of inverse class frequency in centroid-based text classification. In: Proceedings of the 4th International Symposium on Communication and Information Technology, Japan, pp. 1171–1176 (2004)
8. Lertnattee, V., Theeramunkong, T.: Improving Thai academic web page classification using inverse class frequency and web link information. In: Proceedings of the 22nd International Conference on Advanced Information Networking and Applications Workshops, Japan, pp. 1144–1149 (2008)
9. Jones, K.S.: A Statistical Interpretation of Term Specificity and Its Application in Retrieval. J. Documentation 60(5), 493–502 (2004)
10. Read, J.: Using Emoticons to Reduce Dependency in Machine Learning Techniques for Sentiment Classification. In: Proceedings of the Student Research Workshop at the 2005 Annual Meeting of the Association for Computational Linguistics, pp. 43–48. Ann Arbor, Michigan (2005)
11. Short text collection, `http://study.mokoron.com`
12. Lan, M., Tan, C.L., Su, J., Lu, Y.: Supervised and Traditional Term Weighting Methods for Automatic Text Categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(4), 721–735 (2009)

13. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34, 1–47 (2002)
14. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines (2001), http://www.csie.ntu.edu.tw/cjlin/libsvm
15. LIBSVM – A Library for Support Vector Machines, http://www.csie.ntu.edu.tw/~cjlin/libsvm/ (retrieved on July 02, 2014)