Methods for Sentiment Analysis of Twitter Messages

Anton Barhan, Andrey Shakhomirov
Saint-Peterburg State University of Aerospace Instrumentation
Saint-Petersburg, Russia
Barhan.anton@gmail.com

Abstract

This paper deals with the study of existing sentiment analysis methods, as well as development and implementation of methods for sentiment analysis of Twitter messages.

The method designed includes algorithms for determining the polarity of a message and its aspects.

The purpose of this paper is to investigate and develop methods for analysis of the tonality of messages in the social media. To achieve this purpose, the following tasks were set:

1. Review the existing automatic sentiment analysis methods.
2. Study the text features of social media messages in the context of developing methods for their sentiment analysis.
3. Develop a method for automatic sentiment analysis of Twitter messages.

Index Terms: Sentiment analysis, Polarity of a messages, Social media.

I. Introduction

According to the definition given in the book Pang B. & Lee L. Opinion Mining and Sentiment Analysis, sentiment analysis involves automatic analysis of opinions and emotive lexicons expressed in a text.

In the analysis of a text tonality, it is considered that text information on the Internet is divided into two classes: facts and opinions. The definition of an opinion is a key concept.

Opinions are divided into two types:

1. Simple opinion.
2. Comparison.

A. Opinion

A simple opinion contains the statement of an author about one entity. It can be stated directly: “I was pleasantly surprised with the furniture assembly quality”, or implicitly: “After the treatment, my health became stronger”. In both cases, a simple opinion usually has a positive or negative sentiment. In the analysis of the tonality of a text, the following formal definition is given for the first type of opinion: a tuple of five elements (entity, feature, sentiment_value, holder, time) is called a simple opinion, where entity is the object about whose aspect (feature) the author (holder) made an opinion at the time (time). There are 3 types of emotions (sentiment_value): positive, negative and neutral. Neutral emotion means that the text does not contain an emotional component.

Entity is a person, organization, event, product or topic of discussion. Therefore, in various publications, entity is also called object or topic. Often, an entity can be represented as a hierarchical tree of components and sub-components (Fig. 1). Each component is related to a set of attributes. In the above definition of an opinion, aspect means both the components and their attributes. In particular case, an aspect is the entity itself.

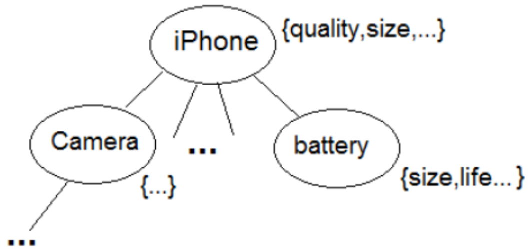

Fig.1. An entity has components and attributes associated with them

An entity aspect has several names: feature, aspect, facet. The term “opinion source” is often used instead of “opinion holder”. These differences in terminology exist because of the specificity of the texts analyzed in the works of researchers and do not alter the substance of the definition of an opinion.

For example, according to the message posted by a user mgerasimov, the user was “frankly disappointed with the twitter new interface for iphone :(”

“TwitterforiPhone” – entity, “new interface” – the entity aspect, “@mgerasimov” – opinion holder, “11 Dec” – date when the message was sent, “frankly disappointed” and “:(“ – part of the message, which suggests that the opinion about the entity aspect is negative.

The second type of opinions – comparison – can be divided into three kinds:

1. Comparison of entity aspects in favor of one (non-equal gradable comparisons).
2. Equating aspects of different entities (equative comparisons).
3. Superiority of one entity over the other (superlative comparisons).

Comparisons of the first kind of “entity aspect 1 is in some way superior to entity aspect 2”, for example: “The GalaxyTab screen is designed with higher quality than its rivals.” The second type expresses the similarity of aspects of different entities, such as: “Both Android and iOS are equally convenient to develop applications for them.” An example of the third type can be the sentence “The Canon device was the cheapest in that shop.”

The opinion of the second type is defined as a tuple (E1, E2, A, po, holder, time), where E1 and E2 are sets of compared entities on the aspect A, po – a set of entities that the opinion holder preferred, time – the time when the opinion was published.

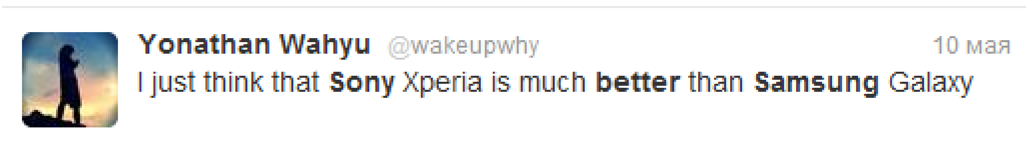

Fig 2. An example of a message containing the second type of opinion

For example, for the above message (Fig. 2), the opinion tuple is as follows: ({SonyXperia}, {SamsungGalaxy}, {common},{SonyXperia},@wakeupwhy, 10 May)

In contrast to the tuple defining the first-type opinion, the opinion tuple of the second-type opinion does not contain an assessment of the author’s emotions.

In text tonality analysis, we often encounter the term “subjectivity” associated with the concept of opinion. Let us give the definition of an objective and subjective sentence:

An objective sentence expresses the actual information about something, whereas a subjective sentence expresses someone’s personal feelings and assumptions.

Sentences containing opinions are usually subjective. Therefore, text analysis for the presence of subjective information is often a subtask of identifying the polarity of that text.

II. METHODS FOR SENTIMENT ANALYSIS OF TWITTER MESSAGES

A. Studying the features of Twitter messages

Let us proceed to solving the first task. In the context of tasks on sentiment analysis of text messages, the main features of Twitter messages are:

1. Small size of messages – 140 characters.
2. Presence of slang, abbreviations and grammatical errors.
3. Presence of special characters.
4. Use of hyperlinks to other users and to external resources.

Fig 3. Twitter posts

B. Implementation of algorithm for determining the polarity of messages

For an efficient algorithm to determine the polarity of messages, it was decided to use machine learning methods. The solution to this problem includes selection of measures used to assess the efficiency of algorithms, selection of attributes used for classification, selection of a suitable training set and efficiency testing of algorithm work.

C. Selecting a measure of efficiency of algorithms

Traditionally the efficacy of text classification is defined in terms of precision and completeness. Let us interpret these terms to determine polarity of documents:

TABLE I

	Classified as positive	Classified as negative
Class positive	TP	FN
Class negative	FP	TN

Let in a collection of N documents, N_p documents have positive sentiment (belong to class positive) and N_n documents have negative sentiment (belong to class negative) .

After classifying these documents, class positive had TP documents correctly classified under it and FP documents wrongly classified under it, while class negative had TN documents correctly classified under it and FN documents wrongly classified under it. Then, in relation to class positive:

Precision is the ratio of number of documents correctly classified under class positive to the number of all documents classified under class positive:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

Completeness is the ratio of the number of documents correctly classified under class positive to the number of documents of class positive in the collection:

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

D. Selection of features

Method of determining the polarity, which was realized in this work, does not presuppose use of aprioristic assumptions on what words and symbols can be contained in the messages that belong to some class. That means that not all features are a priori equal.

N-grams are among frequently used features employed when the task determining polarity is solved. In this work any sequence of letters is a word, and any sequence of n words is n-gram of n order.

For instance, the sentence: «It rained today.;)» includes only the following n-grams of n-1 order and n-2 order: «today», «It», «rained», «rained today», «It rained».

Parts of speech are used as features in a number of papers. This is explained by the fact that opinion contains subjective vocabulary. For instance, paper [10] includes the vocabulary of adjectives and adverbs as terms that express emotions. Just for this reason the decision was taken to select n-grams as features from parts of speech.

As it was shown earlier such characteristics of Twitter messages as slang words and emoticons signal about emotional coloring of the statement. The set of frequently used emoticons was chosen as features.

E. Selection of methods and their comparison:

1) Support Vector Machines: Support vector machines (SVMs) belong to the family of linear classifiers. The purpose of linear classification is to search for a linear hyperplane in a feature space dividing all entities into two classes.

The basic idea of SVMs is to search for a separating hyperplane that has the maximum distance from the points nearest to it in the feature space.

In the case of linearly separable sample, search for a hyperplane can be written down as an optimization problem:

$$\begin{gathered} \frac{1}{2}\|\omega\|^2 \rightarrow \min_{\omega,b} \\ y_i\left(\omega^T x_i + b\right) \geq 1,\ j = 1, \ldots, m, \end{gathered} \tag{3}$$

where $\frac{1}{\|\omega\|}$ is the gap between the hyperplane and the points of both the first and second class, nearest to it, $y_i\left(\omega^T x_i + b\right)$ – the product of the point class value and its position relative to the hyperplane.

For a more general case of linearly inseparable sample, the algorithm can commit errors on the training entities. The new optimization problem is to minimize the error:

$$\begin{aligned}&\frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{l} e_i \rightarrow \min_{\omega,b}\\&y_i\left(\omega^T x_i + b\right) \geq 1,\ j = 1,\ldots,k,\\&e_i \geq 0, i = 1,\ldots,k,\end{aligned} \tag{4}$$

The variables e_i are the values of errors for a sample of k elements. The constant C is used to find a compromise between maximization of the gap and minimization of the total error on the training set.

2) Naive Bayes Classifier: The naive Bayes classifier is a probabilistic classifier based on applying the Bayes' theorem and (naive) statistical independence assumptions of random variables

$$p\left(C \mid F_1,\ldots,F_2\right) = \frac{p(C)\,p(F_1,\ldots,F_2 \mid C)}{p(F_1,\ldots,F_2)} \tag{5}$$

The main advantage of this classifier is its low computational complexity and optimality, provided there is real independence of features.

F. Selection of training sets

Using supervised learning methods requires a training set. Normally, a training set consists of examples of that area in which the classifier will be applied.

A housing consisting of 8,000 sentences, whose polarity is defined was compiled as the training and testing sets. Some of these sentences were retrieved from a mapped out housing provided by the authors Minqing Hu and Bing Liu for free access. The other part was derived from the online system Sentiment140 for sentiment analysis of Twitter messages.

All examples of the training set received were derived from the opinions on electronic technology, namely mobile phones, tablets, and players.

Cross-validation method was used to test the polarity-determining algorithm. The cross-validation procedure is as follows:

1. The training set is partitioned into training and testing subsets.
2. For each partition, the algorithm is trained on the training set, and then on the testing set.
3. The mean value of test results conducted on the testing set is the result of cross-validation algorithm.

In this paper, partitioning into sets was carried out randomly. There is equal probability of each sentence falling into one of the two sets.

The table below shows the results of algorithm testing; *n*-grams, which include words and emoticons, were chosen as features.

TABLE II
MESSAGE POLARITY: N-GRAMS OF WORDS WERE CHOSEN AS FEATURES

Algorithm	Unigrams		Biograms		Jointly	
	Precision	Completeness	Precision	Completeness	Precision	Completeness
Bayes	0.73	0.7	0.72	0.68	0.72	0.71
Support vector machines	0.81	0.74	0.76	0.67	0.81	0.72

The following table shows the testing results, where parts of speech and bigrams of parts of speech were chosen as features.

TABLE III
MESSAGE POLARITY: N-GRAMS OF PARTS OF SPEECH WERE CHOSEN AS FEATURES

Algorithm	Parts of speech		Bigrams of parts of speech	
	Precision	Completeness	Precision	Completeness
Bayes	0.6	0.55	0.54	0.48
Support vector machines	0.65	0.57	0.56	0.53

Support vector machines trained on unigrams of words and emoticons turned out to be the most effective configuration in terms of precision and completeness.

One of the most effective configuration in terms of precision and completeness turned out to be the method of support vectors, learnt on unigrams and emoticons.

G. The problems revealed in comparisons of methods and the task of aspects revelation

The problem of learning with teacher is a creation of training corpus with the examples from the field, where classifier will be used. However, the vocabulary methods possess the similar problem: weight of vocabulary terms, made for one field may not be adequate for the other. The task of aspects extraction is often solved with assistance of learning methods without tutor and statistics methods. We made a decision to use debugged C-value filter, but it was necessary to figure out the main problem of aspects selection.

The task of aspects extraction can be regarded as task of terms extraction, frequently used by the authors of opinions. [7,10]. The researchers consider that terms describing aspects can be single nouns and word combinations with a noun, often seen in the opinions of the same type. The ones with corpus frequency over 1% were identified from all n-grams that meet this demand.

The selected n-grams that consist of two or more words, pass the compactness test. If n-gram is compact in at least two sentences, then it enters the list of aspects.

Compactness is determined in the following way:

- Let *f* – be *n*–gram of *n*–words, *s* – sentence that contains all words from *f* (probably placed in different order).
- If distance between any two words next to f in a sentence s makes no more than three words, then f is compact in this definite sentence.

The terms that have one word also pass the statistic check for purity. The sentences that include the term are selected. Then the sentences that include the term are found in this selection. The sentences, which don’t contain n-grams that passed the compactness test, are counted among the sentences identified. When the number of these sentences is higher than certain experimental threshold, the term gets to the list of aspects.

The value of all n-grams that contain just definite parts of speech, entering set of documents, is calculated in accordance with formula (6).

$$C-value(t)=\begin{cases}\log_2(len(term))\cdot freq(term),\ \text{если}\ |e\_terms|=0\\ \log_2(len(term))\cdot\left(freq(term)-\frac{1}{|e\_terms|}\sum_{elder\in e\_terms} freq(elder)\right)\end{cases}\qquad(6)$$

Where term – *n*-gram, *e*-terms – is a set that consists of higher order *n*-grams, which contain term, $|e\_terms|$ cardinal number *elder* – is the element of this set. The length of term in symbols – $len(term)$.

Let us look at the example that illustrates the work of C-value method. Let in the corpus of opinions about cell phones the bigram «retina display» occurs 8 times, trigrams that contain it «great retina display» and «retina display worse» occur 3 and 2 times correspondingly. Then in accordance with formula (6):

$$\begin{aligned} &C-value(retina\ display) = \log(13)\cdot\left(8-\frac{1}{2}(2+3)\right)\approx 20 \\ &C-value(great\ retina\ display) = \log(18)\cdot 3\approx 13 \\ &C-value(retina\ display\ worse) = \log(18)\cdot 2\approx 8 \end{aligned} \tag{7}$$

If experimentally established threshold *C*-value for this corpus is 15, then only *n*-gram «retina display» will enter numerous terms-aspects.

H. Testing results with application of C-value filter

The method of spread is based on extraction of terms from sentences connected with each other, but Twitter messages are independent, as a rule. Therefore the sets of full-text reviews of electronic technology, made up manually, were used as training sets.

The process of creation of training set is simplified by the fact that methods of learning with teacher do not require selection of pairs (such as class) as in the case of learning with teacher.

I. Testing efficacy

The sentences from marked body are used for testing. 1500 sentences from reviews of two digital cameras and 1820 sentences from review of MP3 player were taken. The aspects were figured out in each of these sentences. The results of testing the implemented algorithm are shown in Table IV:

TABLE IV
EXTRACTION OF ASPECTS

Domain	Precision	Completeness
Cameras	0.65	0.76
Digital players	0.62	0.71

J. Implementation

An algorithm for sentiment analysis of Twitter messages was developed as part of the social network investigation system. One of the modules of this system is the sentiment module for text messages.

Java was chosen as the language for solving the task because the entire system being implemented is developed on it.

The OpenNLP library was used to solve problems associated with text preprocessing – identifying sentences, words, and parts of speech. The LIBLINEAR library was used to solve the problem of identifying text polarity. The licenses of both libraries allow their free use in academic and commercial projects.

Let us proceed to the general scheme of the system. In the figure below, the arrows indicate the data transfer direction and between which components data are transferred (Fig. 4). The NLP module identifies the parts of speech and partitions incoming text into *n*-grams. At the feature extraction stage, a feature vector is formed from the text. Text polarity is the result of binary classification, it takes the value 1 and -1.

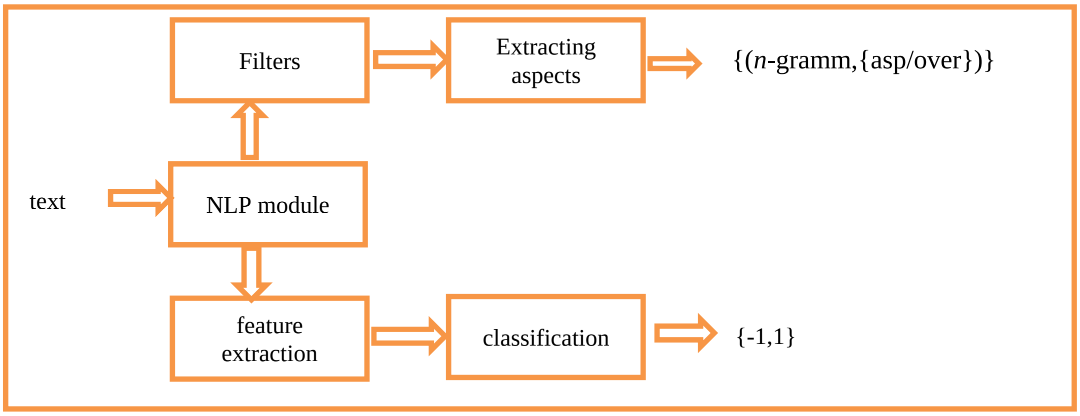

Fig. 4. Transfer direction components data

III. CONCLUSION

The following problems were solved under this paper:

1. Existing automatic sentiment analysis methods were reviewed.
2. The text features of social media messages in the context of developing methods for their sentiment analysis were studied.
3. A method for automatic sentiment analysis of Twitter messages was developed.
4. An experimental assessment of the efficiency of the methods was carried out. The polarity-determining algorithm showed results at the level of contemporary studies.

REFERENCES

[1] Cha, M., Haddadi, H., Benevenuto, F., and Gummadi, K.P. Measuring, “User Influence in Twitter: The Million Follower Fallacy”, *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media (ICWSM)*, Washington, May 2010.

[2] SportSense [HTML] (ec2.compute1.amazonaws.com/sportsense/).

[3] Pang B. & Lee L., “Opinion Mining and Sentiment Analysis”, *Foundations and Trends in Information Retrieval*, v.2 n.1-2, January, 2008 - pp.1-135.

[4] Kerstin Denecke, “Using SentiWordNet for multilingual sentiment analysis’’, *IEEE 24th International Conference on Data Engineering Workshop*, 2008, pp. 507-512.

[5] Bo Pang, Lillian Lee, Shivakumar Vaithyanathan, “Thumbs up? Sentiment Classification using Machine Learning Techniques”, *Language*, 2002, pp. 79-86

[6] Dmitry Davidov, Oren Tsur and Ari Rappoport, “Enhanced Sentiment Learning Using Twitter Hashtags and Smileys”, *Proceeding of the 23rd international conference on Computational Linguistics (COLING)*, 2010.

[7] Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li, “User-level sentiment analysis incorporating social networks”, *Proceedings of the Sixteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2011.

[8] Luciano Barbosa and JunlanFeng, “Robust Sentiment Detection on Twitter from Biased and Noisy Data”, In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, Beijing, 2010.

[9] Bing Liu., Sentiment Analysis and Subjectivity, *Handbook of Natural Language Processing*, Second Edition, 2010.

[10] Mike Thelwall, Kevan Buckley, and Georgios Paltoglou, “Sentiment in Twitter Events”, *Journal of the American Society for Information Science and Technology*, archive Vol. 62, 2011.