



DETECTING SENTIMENT IN TWITTER DATA – SUPERVISED MACHINE LEARNING APPROACH FOR TWITTER SENTIMENT ANALYSIS IN PYTHON

JOANNA MICHALAK^a,

^a*Nicolaus Copernicus University in Torun, Poland*

ABSTRACT

Twitter is one of the most popular micro-blogging platforms where users publish their thoughts and opinions and much attention is paid to explore sentiment of these opinions. This paper focuses on the characteristic of Twitter, tweets and supervised machine-learning method for Twitter Sentiment Analysis. Discussion focuses on the following issues: access to the tweets and creating a database, the process of cleaning the database and process of tweets classification into positive and negative groups. The TSA process is presented in Python by simplified architecture.

ARTICLE INFO

Available online 23 December 2016

Keywords:
social media,
twitter sentiment analysis,
opinion mining,
machine learning

JEL: C81, M31.

Doi: 10.19197/tbr.v15i4.71

INTRODUCTION

Increasing amount of attention is paid to monitoring social media to explore sentiments expressed through them. One of the most popular micro-blogging platforms is Twitter (www.twitter.com). It is estimated that over 284 million users sent about 500 million messages and 1 billion unique monthly visits to sites with embedded tweets have been listed (about.twitter.com/company, 10.11.2016). Because the platform is powered by constantly growing messages from different social groups from all over the world, unlike traditional blogging platforms, Twitter is a dynamic forum (Giachanou & Crestani, (2016)). Users express their opinions about any service, product or event. According to (Giachanou & Crestani (2016)), Twitter has since evolved into a complex information-dissemination platform. The dimensions of Twitter messages may have the following

attributes: space, time, topic, mood and network structure. As a result, researchers began to focus attention on the possibilities of processing messages generated by users in an attempt to assess public opinion and sentiment. Customers exchange their past experience about products, services, events and even politicians. A key feature of social media is that it enables anyone to freely express views and opinions without disclosing their true identity (Liu, (2011)). Consequently, these opinions are highly valuable. However, these opinions are also hidden in a huge data stream and it is nearly impossible anyone to look through it. Basic measure for mining such data is transforming them into a structure that can be understood. There are a number of platforms that deal with this professionally, including www.brandwatch.com, www.tracker.com, associated with TV-rating Company, www.nielsensocial.com.

Sentiment Analysis and Opinion Mining are discussed in many publications; however, implementation of these solutions in Twitter is still being developed. Methods developed for the text analysis, including sentiment analysis, are verified to deal with the challenges posed by data from Twitter. This paper contains basic information about Twitter Sentiment Analysis (TSA) methods. Theoretical part includes a review of a supervised machine-learning method for Twitter Sentiment Analysis. Empirical part includes some results of sentiment mining from messages only as an illustration of discussion. Results were obtained by Python 3.4.

TWITTER - A GENERAL VIEW

Micro-blogs are a “group of Internet-based applications that build on the ideological foundations of Web 2.0 and allow the creation and exchange of user generated content” (Kaplan & Haenlein, (2011)). Twitter, created in 2006, is a service through which users can share messages, links to external websites, images or videos that are visible to users subscribed to their Twitter account (Giachanou & Crestani, (2016)). *Tweets* (short Twitter messages) are limited to 140 characters; messages can include links, videos and images (Figure 1 show example of a Twitter message). Important terms related with Twitter are listed below:

- Mentions: indication that the post mentions another user can be placed anywhere in the body of the tweet and are marked by *@username*.
- Replies: indicate that the post is an answer to another tweet; they are placed next to the username to create the reply.
- Followers: users that follow another user’s tweets and activity; they receive updates from those they follow.
- Retweet: RT symbol of a repetition of someone's message and passing it on. This may be a message in a partially modified form, but it is customary to keep the person from whom the information originates. Retweet is marked by *RT@username*. It is a powerful tool for disseminating information.
- Hashtags: indicate the affiliation of a tweet to a certain topic, marked by *#keyword*, assign a word to the category and allow others to keep track of conversations around the topic.

Big day today at @YaleSOM -- #YaleTechSummit16, @netimpact Case Competition, and first #admissions Interview Day of the year!

Figure 1. Example of a Twitter message
Source: twitter.com

Twitter offers a series of APIs to provide programmatic access to Twitter data, including reading tweets. The registration process includes:

- the need to have a Twitter account (with the specified e-mail and phone number),
- creating a new app (<http://apps.twitter.com>); then we can find information we need to authenticate our application: access token, access token secret, consumer key, consumer key secret.

Official documentation with described limitations can be found here: dev.twitter.com. Twitter APIs can be categorized into two classes: REST APIs and Streaming API [Figure 2]. According to Bonzanini (www.packtpub.com, 27.08.2016) when interacting with Twitter via REST API, we can search for existing tweets. Streaming API looks into the future. By keeping the HTTP connection open, we can retrieve all the tweets that match filter criteria, as they are published. REST APIs are useful when we want to search for tweets authored by a specific user or we want to access our own timeline, while the Streaming API is useful when we want to filter a particular keyword and download a massive number of tweets, for example about events.

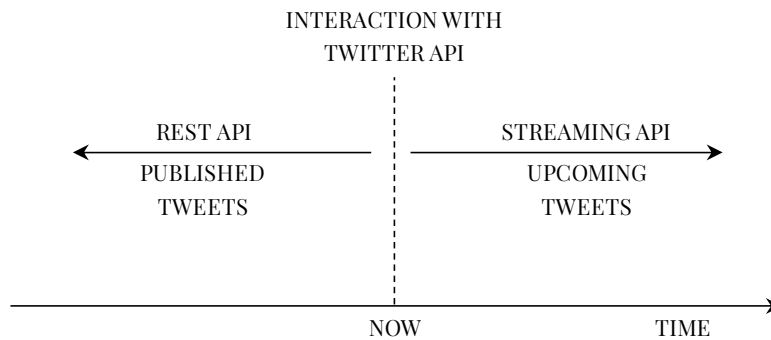


Figure 2. The time dimension of searching versus streaming
Source: Bonzanini M., Mastering Social Media Mining with Python, <https://www.packtpub.com/> [27.08.2016].

SENTIMENT MINING: RELATED WORKS

The list of works related to SA and TSA process is long; however, works listed below are important from the point of view of systematization of the fundamental issues raised in this work.

1. Discussion about sentiment analysis issues can be found in the following works:
 - (Pang & Lee, (2008)) provide a comprehensive overview of the SA approaches using various types of data. They analyzed the performance of different classifiers on the movie reviews.
 - (Tsytsarau & Palpanas, (2012)) organized their survey based on the main tasks of SA wherefore they present definitions and discuss problems and the various approaches.
 - (Liu & Zhang, (2012)) provided a comprehensive and detailed description of all the important concepts and topics related to SA.

2. Discussion about Twitter Sentiment Analysis issues can be found in the following works:
 - (Giachanou & Crestani, (2016)) performed a very good summary of all important issues related to the TSA process.
 - (Bermingham & Smeaton, (2010)) performed a study attempting to examine if the short length of tweets makes this task more difficult compared to longer texts.
 - (Saif et al. (2012)) described how they created a corpus from Twitter posts using emotion word hashtags.
 - (Petrovic et al. (2010)) performed the datasets for TSA: Edinburgh Twitter Corpus.

SENTIMENT ANALYSIS - GENERAL VIEW

According to Liu (Dodd, (2014)) “sentiment analysis [SA] is the field of study that analyzes people’s opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, and their attributes.” In addition, sentiment can be expressed with different strength/intensity levels. (Liu, (2012)) presents opinions in mathematical way, as a quadruple of:

$$\{e_i, a_{ij}, s_{ijkl}, h_k, t_l\},$$

[where e_i is the name of an entity, a_{ij} is an aspect of e_i , s_{ijkl} is the sentiment on aspect a_{ij} of entity e_i (positive, negative or neutral), h_k is the opinion holder and t_l is the time when the opinion is expressed by h_k]

Sentiment analysis can be investigated mainly on three levels (Liu, (2012)):

- Document level: the task at this level is to classify whether a whole opinion document expresses a positive or negative sentiment (does not apply to evaluation documents or comparison of multiple entities).
- Sentence level: the task at this level goes to the sentences and determines whether each sentence expressed a positive, negative, or neutral opinion.
- Entity and aspect level: instead of looking at language constructs (documents, paragraphs, sentences and so on), the aspect level directly looks at the opinion itself. It is based on the idea that an opinion consists of a sentiment (positive or negative) and a target (of opinion).

A sentence expresses a single sentiment form a single opinion holder. One sentence may include:

- a clear opinion about the product (for example: product *A* is amazing),
- more than one opinion (for example: product *A* has a very
- good lens and a good zoom),
- or may include conflicting opinions (for example: product *A* has a very
- good lens, but generally performs worse than others).

Complex sentences express different sentiments on different targets. Although a sentence may have an overall positive or negative tone, some of its components may express opposite opinions. Sentence level sentiment classification cannot deal with opinions in comparative sentences (Liu, (2012)).

Twitter Sentiment Analysis [TSA] should be seen as part of SA, which deals with text from Twitter. TSA of data is a difficult task and faces some challenges, which are related to the specific features of tweets:

- limitation to 140 characters – shorter messages make the analysis more difficult, especially if they contain links or photos (we must remember that TSA uses techniques that have been used to analyze longer texts), moreover, they are burdened with:
 - informal language (slang and new words),
 - sarcasm and irony,
 - negation,
 - stop words,
 - multilingual content,
 - incorrect English,
 - special characters (like emoticons).

More details about each of these challenges can be found in (Giachanou & Crestani, (2016)). All of these features are important and must be included in the TSA process.

THE PROCESSING OF TWITTER DATA

This work focuses on supervised machine learning method. Example of typical schema of TSA process is presented in Figure 3.

The machine-learning approach employs a machine-learning method and a number of different features to build a classifier that can detect tweet sentiment and provide a reliable output. According to (Giachanou & Crestani, (2016)), supervised machine learning is a technique where the task is to deduce features from tagged training samples. The training samples for supervised learning consist of large sets of examples for a particular topic. In supervised learning, every example of training data comes as a pair of input (vector quantity) and output value (desired result). As the training database can be enormous, it may pose problems with manual labeling of tweets into positive/negative groups. Consequently (Giachanou & Crestani, (2016)), (Go et al. (2009)) and (Read, (2005)) propose to apply the following technique to collect data: they used the emoticons to differentiate between negative and positive tweets. The main advantage is

that it helps in fast building of a training set. However, distant supervision methods have some disadvantages: the quality of this approach is low and it may result in a lower accuracy of classifiers.

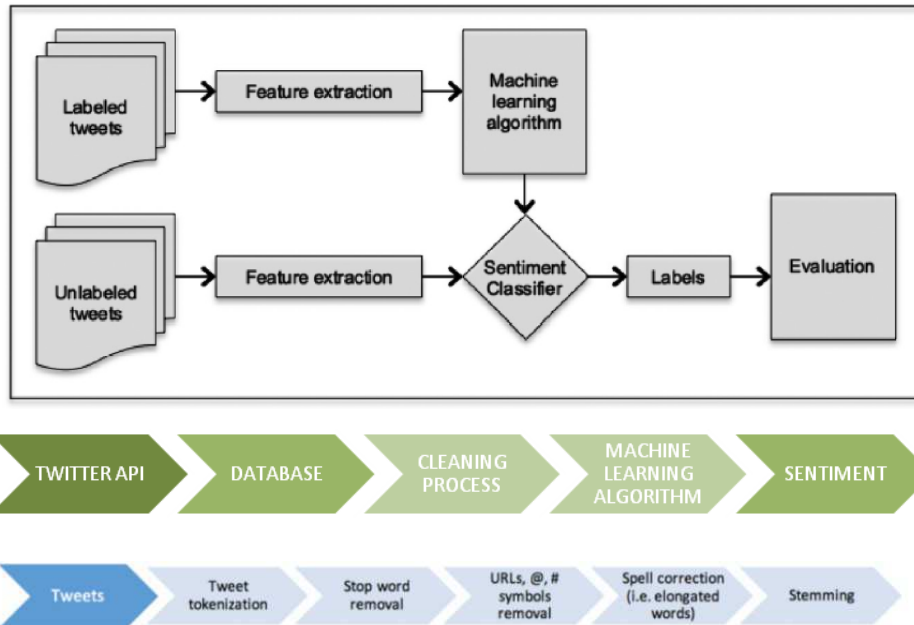


Figure 3. [a] Typical process of sentiment classification
 [b] Simplified architecture of Twitter Sentiment Analysis process in Python
 [c] Process of preprocessing tweets

Source: Dood [2014], Vasu [2013], Giachanou & Crestani [2016].

For TSA-supervised machine learning process, several training datasets have been built. The list below contains only a few examples of corpus (more can be obtained from the work by (Giachanou & Crestani, (2016)):

- The Edinburgh Twitter Corpus, (Petrovic & Osborne & Lavrenko, (2010)): Corpus contain 97 million twitter posts. It has been collected from November 11, 2009 until February 1, 2010.
- Stanford Twitter Sentiment Corpus (Go et al. (2009)), (help.sentiment140.com/for-students): The tweets were collected between April 6 and June 25, 2009 and they should contain at least one emoticon. These messages are classified as either positive or negative: a tweet is considered positive if it contains emoticons such as :), :-), :) , :D, or =), whereas the tweet is negative if it contains :(, :-(, or : (. Neutral tweets are not considered. The link to training data and test data: <http://twittersentiment.appspot.com/>.
- Sanders Corpus: dataset consists of 5,513 manually annotated tweets with respect to four different targets: Apple, Google, Microsoft and Twitter. Each tweet was annotated as positive, negative, neutral, or irrelevant given its topic, resulting in 570 positive, 654 negative, 2,505 neutral, and 1,786 irrelevant tweets.

- Obama-McCain Debate: dataset consist of the 3,238 collected tweets manually labeled as positive, negative, mixed or other. Tweets were collected during the presidential debate (Obama-McCain).

The ML algorithms analyze data and generate an output function, which is used to map new data sets to respective classes. The following classifiers are used: Naïve Bayes, Maximum Entropy, Support Vector Machines, Multinomial Naïve Bayes, Logistic Regression, Random Forest and Conditional Random Filed.

The measures of algorithm for determining the polarity of messages are as follows: accuracy, precision, recall and F-score.

SML approach is very common and useful for TSA process but some authors indicate its limitations: the performance depends on the number of training datasets and they usually require large amounts of annotated texts to achieve a high performance (Giachanou & Crestani, (2016)). Moreover, as a result, classifiers can be domain-dependent – it means that classifiers get good results on the domain they were trained and in order to progress to a different domain they must be retrained.

Figure [2b] consist of a simplified architecture of Twitter Sentiment Analysis process in Python. For each stage, practical guidelines are listed in stages (1-4).

The problem was treated as a binary classification between positive and negative groups. Examples of messages come from database, which was created during the season of selected TV shows from November 2016¹ (here, messages have been selected that do not allow to identify the name of the show). Discussion concerns the following issues: access to the tweets and creating a database, the process of cleaning the database and using SML for classification of tweets into positive and negative groups. The process presented here is of simplified nature.

Stage 1. Twitter API

Public stream API was used for data collection, as it allows access to a global stream of twitter data that could be filtered as required. Downloading a dataset of tweets via Twitter API using Python 3.4 can be done using a *twitter* or *tweepy* library. Twitter's API returns result in a JSON format, which can be parsed in a Python script. The data can be saved in a txt, csv or SQL format. Simple Python code for access to Twitter is shown below²:

```
consumerKey="XXXXXX"
consumerSecret="XXXXX"
oauthToken="XXXXXX"
oauthSecret="XXXXXXXX"
twitter_stream = TwitterStream(auth=OAuth(oauthToken, oauthSecret, consumerKey,
consumerSecret))
stream = twitter_stream.statuses.filter(track='keywords')
```

¹ At the time of preparing this work, the database is not yet ready to be introduced.

² <http://www.tweepy.org>.

A tweet can contain a lot of information [dev.twitter.com/overview/api/tweets], for example:

- contributors (id, id_str, screen_name) – indicating the author of tweet,
- coordinates – geographic location,
- created_at – UTC time when a tweet was created,
- entities (hashtags, urls, user_mention).

```
{"created_at":"Thu May 12 19:49:29 +00002016",  
"id":"00000000000000",  
"id_str":"00000000000000",  
"text":"XYZ",  
"source":"XYZ",  
"truncated":false,  
"in_reply_to_status_id":null  
"lang":"en"  
"coordinates":  
  //  
  [-77.119759,38.791645],  
  [-76.909393,38.791645],  
  [-76.909393,38.995548],  
  [-77.119759,38.995548].//  
"retweet_count":1585}
```

The issue at this stage is to determine criteria that are related to research goal (called: keywords). This is one of the challenging fields of mining data from Twitter. Some messages may contain selected keywords but do not relate to the purpose of the study – it will be difficult to see them in the database, so possibly they need to be minimized. There are no precise recommendations, however it is good to expand the keyword list by the number that will allow a precise identification of the required message.

Stage 2. Database

It seems that the optimal approach is to use a SQL database, which is supported by some advantages like easier segmentation of the data (for example: *id*, *text* and *time* attributes of the tweets can be separated by columns) and easy integration with Python programming language. Writing to the SQL database can be done using the *sqlite3* module³. Below is the example of code for creating database in Python:

```
conn = sqlite3.connect('SPNtraining.db')  
conn.execute("CREATE TABLE twitter(ID INT, date TEXT, tweet TEXT)")  
conn.execute("INSERT INTO SPN(ID, date,tweet) VALUES(?,?,?)")
```

Columns in the created table can contain all the information contained in tweets mentioned in stage 1. In practice, two data sets are required: training and related to purpose of the research. As it was already mentioned, manual labeling of messages for

³ <https://docs.python.org/2/library/sqlite3.html>

training data may, for some reason, be impossible. We can assume that emoticons (☺,☹) can be successfully used as a noise label in order to automatically classify tweets in positive/negative groups as emoticons represent the overall sentiment contained in tweets.

For this project, with only illustrative purpose, the movie reviews database that is the part of the NLTK corpus was used. The NLTK corpus *movie_reviews* data set has the reviews, and they are labeled already as positive or negative.

Stage 3. Cleaning process

Cleaning process involves removing any unwanted content from the input tweets. It includes the following steps: remove re-tweets, convert all text to lower case, remove the links, remove white spaces, remove hashtags, remove punctuation, stop word and stemming (stemming is the process of reducing inflected words to their word stem). Example of a Python script is listed below (Dood, (2014)):

```
#Remove retweet
data = data.replace('RT', '')
#Convert all text to lower case
data = data.lower()
#Convert links to URL
data = re.sub('((www\.[^\s]+)|(https?://[^\s]+))','URL',data)
#Convert @user to AT_USER
data = re.sub('@[^\s]+','AT_USER',data)
#Remove white spaces
data = re.sub('\s+', '', data)
#Remove hashtag
data = re.sub(r'#([^\s]+)', r'\1', data)
#remove white space from beginning and end
data = data.strip()
#stopWord list
stopWordList = stopwords.words('english')
```

Table 1. Example of input and output data in a cleaning process

input tweet	output tweet
@XYZ just not at the same time :(('AT_USER time :(,)
I dont want us to be strangers again. :(('i dont want us strangers again. :(,)
Not good :(('not good! :(,)
RT @XYZ: i feel you :(https://t.co/xxxxx	(' AT_USER feel :(URL
@XYZ SORRY :(('AT_USER sorry :(,)

Source: own work.

The stop words⁴ list was taken from NLTK tool for Python. According to the challenges of TSA, after the cleaning process of the data and the use of stop words, it can be noted that some tweets contain only one word. It creates classification problems (Table 2): tweet *'just not at the same time'* after the cleaning process consist only one word *'time'* which generates some classification problems. General expression of the tweet is negative but algorithm may allocate it in the positive group. The word *'time'* does not represent emotional context.

Stage 4. Machine Learning Classifier and sentiment output

In this stage, it can clearly be noticed that there is a tendency to use following tools:

- Weka (<http://www.cs.waikato.ac.nz/>): an open-source collection of machine-learning algorithms for data mining tasks, which was created at the University of Waikato.
- LingPipe (<http://alias-i.com/lingpipe/index.html>): a toolkit for processing text using computational linguistics.
- Python, NLTK (<http://www.nltk.org/>): a leading platform for building Python programs to work with human language data.

Creating a classifier includes a number of basic steps (www.nltk.org/book/cho6.html):

- The first step in creating a classifier is deciding what features of the input are relevant and how to encode those features.
- After defining feature extractor, a list of examples and corresponding class labels must be prepared.
- Use the feature extractor to process the data and divide the resulting list of feature sets into a training set and a test set.
- Examine the classifier to determine which features are the most effective for differentiation.
- Train the classifier.

Table 2. Results of classification - sentiment

tweet - correct	sentiment
want strangers again.	Negative/ correct
not good!	Negative/ correct
AT_USER dark and... know rest... sad day	Negative/ correct
tweet - may include problems with classification	sentiment
AT_USER feel	positive/negative
AT_USER time	positive/negative

Source: own work.

⁴ Stop Words are common words that have low discrimination power and are usually filtered out before processing the text.

Table 2 includes some examples of classification divided into two groups: tweet labeled correctly and tweet which may include some classification problems. Correct labeling implies that emotional context of tweet complies with label. Tweets with uncertain outcome are those messages that match both the positive and negative sentiment (in case of a binary classification). Naïve Bayes algorithm was used for classification (with the accuracy of 69.57%).

CONCLUSIONS AND FUTURE WORKS

The objective of this work was to discuss key issues related to the TSA process with some introduction to Python. As shown here, TSA process it is a very complex problem. Some recommendations to improve whole process may be made:

- Different filtering criteria can be made for creating datasets.
- Training data influences results – distant supervision has some disadvantages: quality of this approach is low and may result in lower accuracy of classifiers – so the recommendation would be to have the training data domain-dependent with manually labeled tweets.
- Stop words: as (Giachanou & Crestani, (2016)) have shown, a typical pre-compiled stop-words list may not be suitable for Twitter and may influence the TSA performance. Generally, these stop-word lists may include some words expressing emotional context, for example *'like'* which may be important in the TSA process.
- The selection of features is very important for the effectiveness of the supervised methods. The process should include an analysis of the comparative effectiveness of different methods and selection of the best in the context of the study. There are numerous studies on this issue; a good summary can be found in (Giachanou & Crestani, (2016)).

It is worth mentioning that the following new directions for improving Twitter mining are taken:

- Performance of individual classifiers has been applied, also known as ensembles classifiers (combining classifiers). Related works: (Lin & Kolcz (2012)), (da Silva et al. 2014), (Hassan et al. 2013).
- Deep learning – also known as deep structured learning is a machine learning field. It is applied to solve perceptual problems such as image recognition and understanding of natural languages. Deep learning uses neural networks to learn many levels of abstraction (Giachanou & Crestani, (2016)).

It is also indicated that according to dynamic growth of social media, omitting information with their emotional context can be a big mistake. Data volume is exploding and so is its value. Twitter generates a huge amount of real-time data, including who your customers are, what they like, where they come from and how they feel about something. Automatically identifying emotions expressed in Twitter text has a number of applications, including:

- customer relation management;
- determining popularity of products, services, events and governments;

- gathering critical feedback about problems in newly released products;
- capturing sentiment on a trend line so it can identify sudden positive or negative spikes;
- identifying if positive/negative opinions come from influential social media personality;
- messages are public, so it is also possible to analyze the competition;
- often, by monitoring social media, decision-makers can quickly identify problems and handle them.

In summary, digging in this information can be useful to make smarter business decisions and exploiting the full potential of data can have a real impact on profits.

Future work will concentrate on:

- developing tools for TSA, and
- comparative analysis of classifiers.

REFERENCES

- Asur, S., & Huberman, B. (2010). Predicting the future with social media. In Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01 (WI-IAT'10). *IEEE Computer Society*, Washington, DC, 492–499.
- Barbosa, B., & Feng J. (2010). Robust sentiment detection on twitter from biased and noisy data. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters (COLING'10), *Association for Computational Linguistics*, Stroudsburg, PA, 36–44.
- Barham, A., & Shaknomyrov, A. Methods for sentiment Analysis of Twitter Messages, Proceeding of the 12th Conference of FRUT Association, available online at: <https://fruct.org/publications/abstract12/files/Bar.pdf>, 19.11.2016.
- Bermingham, A., & Smeaton, A. (2010). Classifying sentiment in microblogs: Is brevity an advantage? In Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM'10). ACM, New York, 1833–1836.
- Bollen, J., & Mao, H., & Zeng X. (2010). Twitter mood predicts the stock market. *J. Comput. Sci.* 2, 1–8.
- Bonzanini, M., Mastering Social Media Mining with Python, available online at: <https://www.packtpub.com/> [27.08.2016].
- da Silva, F., & Hruschka, E., & Hruschka E. (2014). Tweet sentiment analysis with classifier ensembles. *Decision Supp. Syst.* 66, 170–179.
- Dodd, J., Twitter Sentiment Analysis, Final Project Report, at: <http://trap.ncirl.ie/1868/1/johndodd.pdf>
- Giachanou, A., & Crestani, F. (2016) Like it or not: A survey of Twitter sentiment analysis methods, *ACM Comput. Surv.* 49, 2, Article 28.
- Kaplan, A., & Haenlein, M. (2011). The early bird catches the news: Nine things you should know about micro-blogging. *Business Horizons* 54, 105–113.
- Kiritchenko, S., & Zhu, X., & Mohammad S. (2014). Sentiment analysis of short informal texts. *J. Artif. Intell. Res.*, 723–762.

- Lin, J., & Kolcz A. (2012). Large-scale machine learning at twitter. In Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data (SIGMOD'12). *ACM, New York*, 793–804.
- Liu, B. (2012). Sentiment Analysis and Opinion Mining. *Morgan & Claypool Publishers*. available online at: ww.cs.uic.edu/~liub/FBS/SentimentAnalysis-and-OpinionMining.pdf [19.11.2016]
- Liu, B., & Zhang, L. (2012). A survey of opinion mining and sentiment analysis. In Mining Text Data. *Springer, New York*, 415–463.
- Martínez-Cámara, E., & Martín-Valdivia, T., & Montejo-Ráez, U. (2012). Sentiment analysis in twitter. *Nat. Lang. Eng.* 20, 1–28.
- O'Connor, B., & Balasubramanyan, R., & Routledge, B., & Smith, N. (2010). From tweets to polls: Linking text sentiment to public opinion time series. In Proceedings of the 4th International AAI Conference on Weblogs and Social Media (ICWSM'10). *AAAI Press*.
- Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In Proceedings of the 7th on International Language Resources and Evaluation Conference (LREC'10). *European Language Resources Association (ELRA)*, 1320–1326.
- Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis, *Foundations and Trends in Information Retrieval* 2, 1–135.
- Pang, B., & Lee, L. & Vaithyanathan, S. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10 (EMNLP'02). *Association for Computational Linguistics*.
- Petrović, S., & Osborne, M., & Lavrenko, V. (2010). The Edinburgh twitter corpus. In Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media (WSA'10). *Association for Computational Linguistics*, Stroudsburg, PA, 25–26.
- Saif, H., & Fernández, M., & He, Y., & Alani, H. (2012). On stopwords, filtering and data sparsity for sentiment analysis of twitter. In Proceedings of the 9th International Language Resources and Evaluation Conference (LREC'14). *European Language Resources Association (ELRA)*, 810–817.
- Saif, H., & He, Y., & Alani, H., (2012). Alleviating data sparsity for twitter sentiment analysis. In Workshop on Making Sense of Microposts (#MSM2012), Big Things Come in Small Packages at the 21st International Conference on the World Wide Web (WWW'12). *CEUR-WS.org*, 2–9.
- Tai, Y. J., & Kao, H., (2013). Automatic domain-specific sentiment lexicon generation with label propagation. In Proceedings of the International Conference on Information Integration and Web-Based Applications & Services (IIWAS'13). *ACM, New York*, 53–62.
- Tsytsarau, M., & Palpanas, T. (2012). Survey on mining subjective data on the web. *Data Min. Knowl. Discov.* 24, 478–514.
- Vasu, J. (2013). Prediction of movie success using sentiment analysis of tweets. *The International Journal of Soft Computing and Software Engineering [IJSCE]*, Vol. 3, No. 3, Special Issue: The Proceeding of International Conference on Soft Computing and Software Engineering 2013 [SCSE'13], San Francisco State University, CA, U.S.A.
- Zhang, L., & Ghosh, R., & Dekhil, M., & Hsu, M., & Liu, B. (2011). Combining Lexiconbased and Learning-based Methods for Twitter Sentiment Analysis. Technical Report.

about.twitter.com/company.

cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf.

help.sentiment140.com/for-students.

<http://alias-i.com/lingpipe/index.html>.

<http://apps.twitter.com>.
<http://twittersentiment.appspot.com/>.
<http://www.cs.waikato.ac.nz>.
<http://www.sananalytics.com/lab/twitter-sentiment/>.
<http://www.tweepy.org/>.
<https://docs.python.org/2/library/sqlite3.html>.
www.aclweb.org/anthology/W10-0513.
www.brandwatch.com.
www.nielsen.com.
www.tracker.com.
www.twitter.com.