Twitter-based Sentiment Analysis

Jakob Gruber
0203440

Matthias Krug
0828965

Stefanie Plieschnegger
0926102

Christian Proske
1328245

Mino Sharkhawy
1025887

ABSTRACT

Sentiment analysis has become hugely popular in recent years. Especially Twitter provides a wealth of data for a variety of topics, which can be processed and classified in order to provide an approximate measure of the overall opinion. However, classification of Twitter-based data is in some ways different to traditional text mining and introduce additional challenges. In this paper, the typical steps and problems of classifying tweets are outlined including preprocessing steps, training, and evaluation.

Categories and Subject Descriptors

H.3 [**Information Systems**]: Information Search and Retrieval

Keywords

Sentiment Analysis, Opinion Mining, Twitter, Classifier, Natural Language Processing

1. INTRODUCTION

The general opinion about a specific product or service has a great influence on its reputation. People often want to know what others think about a special product they might be willing to buy, about a new movie, or about a hotel they are going to book. Companies may also be interested in their customers' opinions, politicians may wish to receive feedback, and social organizations may have interest in an ongoing debate [8]. The world wide web provides many ways for people to distribute their experiences and sentiments. Machine learning algorithms make it easier to process and evaluate those sentiments and are therefore able to provide an overall opinion to a certain topic. This type of analysis is called opinion mining or sentiment analysis [7, 9]. Clearly, there are some challenges when assessing the opinion of people, especially when classifying microblogging services like Twitter[1]. The underlying paper gives an overview of different sentiment analysis approaches and outlines special problems related to the classification of microblogging services. These challenges are discussed in section 2. In section 3 we describe transformations of tweets and feature selection. Section 4 deals with training and evaluation of classifiers. In section 5 we introduce related work, and section 6 concludes the paper.

[1] `http://twitter.com`

2. ANALYZING DATA

The analysis of text and the classification whether its content should be considered positive, negative or neutral, is the core functionality in sentiment analysis. We first discuss general considerations of classifying data and subsequently outline additional challenges related to Twitter-based data.

2.1 Text Mining vs. Sentiment Analysis

One would assume that text mining and sentiment analysis are very similar. Text mining e.g. may deal with classifying documents by topic, which represents one of the easier tasks. Topic-based text classification tries to match a text into a category like sport, politics etc. and therefore topic-related words are identified in order to classify a text. However, sentiment analysis requires focus on typical sentiment words such as "hate", "love", "like", "regret". When it comes to identifying the overall sentiment, it turns out that a text may contain both positive and negative sentiment words:

"My new smartphone is really cool, the display is just gorgeous. The battery life is really bad, however."

It may even contain positive words within a generally negative meaning (sarcasm):

"Oh, of course - I have a lot of time. Just keep on using my money for paying those really fast and friendly authorities."

Obviously, this makes the task of sentiment analysis more difficult. Moreover, it is hard to teach a machine patterns such as sarcasm or how to identify the intended meaning behind words [7, 9].

Another crucial point are dependencies; *Topic and domain dependencies* mainly focus on the problem that sentiments can have a different meaning depending on the underlying topic or domain. E.g. a word such as "unpredictable" may have a positive meaning if it is used in a movie review, but could have a negative sentiment when used to describe the behavior of a car. *Temporal dependencies* describe the prob-

lem of training a classifier with data from a certain time-period and applying it for data classification of another time-period. These dependencies may have an influence on the accuracy of classification as well [10, 9].

2.2 Twitter-based Data

Twitter is a form of microblog: users can post short messages of up to 140 characters, often in reaction to certain events. This results in people using abbreviations, emoticons, slang and (intentional) spelling mistakes in order to fit and express their opinion accurately. For example, users sometimes emphasize certain words by using uppercase characters or repeating vowels (e.g. "I'm feeling happyyyyy"). Moreover, Twitter uses special characters such as the "@" (targets direct the post to a specific user) and "#" (hashtags signify that the post refers to a certain topic). Another problem is that Twitter data may contain spam. The fact that tweets can be retweeted should also be kept in mind and may have an influence on the classifier, depending on the strategy of classification. All these special characteristics play a huge role when analyzing tweets [1, 10].

3. PREPROCESSING TWEETS

In order to achieve the most precise result when classifying tweets, some preprocessing steps are suggested. Preprocessing describes the cleansing of data and the text preparation for classification. It needs to be remarked that not all steps are necessary and it may also depend on the data set and the selected classifier how much influence these preprocessing tasks have. Generally, preprocessing contains data-cleansing steps, so called `transformations` and feature selection, also called `filtering` [4].

3.1 Transformation

Transformation contains all steps that make the data easier to classify: stripping whitespace, normalization (e.g. to lower case the text), methods for dealing with noisy data etc. In the following, some of these approaches from [12, 8, 3, 1, 9, 4] are briefly described.

3.1.1 Extract Noisy Data

The extraction of noisy data includes the removal of e.g. advertisement/spam.

3.1.2 Stopword Removal

One of the most popular preprocessing steps is stopword filtering. Stopwords are defined as words that do not contain additional sentimental information and can therefore be removed from the text, as it will only make the text shorter, but does not lead to information loss. Such stopwords are for example: "a", "the", "about", and "is".

3.1.3 Stemming

This approach deals with the identification of words that have a similar or identical meaning but are not spelled the same due to grammatically reasons, e.g. identifying "was" as a form of "be".

3.1.4 Emoticon Dictionary

Emoticons in tweets may be replaced with its actual meaning. This approach requires a list with all emoticons and its interpretation. Then those could be labeled according classifications like extremely-positive, positive, neutral, negative, extremely-negative as suggested in [1].

3.1.5 Stripping Emoticons

However, the approach of stripping out emoticons of the training data has also been suggested: [3] consider emoticons as noisy data and experienced a better performance for training maximum entropy modeling (MaxEnt) and support vector machine (SVM) classifiers, although the test data may include emoticons.

3.1.6 Acronym Dictionary

This preprocessing approach deals with the use of likely abbreviations in Twitter-data. Typical acronyms in tweets are e.g. "lol" (laugh out loud), "brb" (be right back), "gr8" (great) etc.

3.1.7 Replacing URLs and Targets

Another preprocessing approach is replacing all URLs in tweets with a special tag — this way the actual URL will not have an influence when classifying the data, only the fact that there is a URL will have an impact. The same can be done with targets (already mentioned in section 2.2).

3.1.8 Replace Negations

The replacement of all negative words like "non" or "never" by a tag "not" ease the classification as well.

3.1.9 Replacing Repeated Characters

As pointed out in section 2.2, repeated letters are sometimes used to emphasize words. In order to make these words comparable they may be normalized by replacing all characters that are repeated more than two (suggested in [3]) or more than three times (suggested in [1]). The word "happyyyyy" would become "happyy" (respectively "happyyy" if using the three-times-replacing approach). The strategy of replacing sequences by three characters makes the use of emphasized and normal words distinguishable.

3.2 Feature Selection

Features are words or phrases of a text which are then used determine sentiment. During training, classifiers attempt to form correlations between features and their label, while a fully trained classifier attempts to deduce the appropriate label for a set of given features. There are several approaches with varying effectiveness, some of which are described in the following.

3.2.1 Tokenization and N-grams

The data needs to be separated in order to use the words as features. Normally, the text is split by spaces and punctuation marks. In addition there are approaches to keep words like "don't" as one word [8]. Tokenized words are also known as unigrams. Using n-grams means that combinations of words are used. Unigrams are therefore combined, depending on the `n`. Approaches include combining unigrams and bigrams as features [7, 3].

3.2.2 Part of Speech

Part of speech (POS) tags deal with the syntactic analysis of sentences. For example, adjectives are assumed to provide sentimental meaning [7]. POS taggers require adaption to the specific properties of tweets. One such approach can be found in [2].

3.2.3 Opinion Words

Some words such as "love", "hate", "beautiful", "great" are known to carry special sentiment. Also, phrases like `"All that glitters isn't gold."` may be associated with special meaning during feature selection [7].

3.2.4 Twitter Specifics

Due to their informal nature and inherent length limits, tweets must must be treated differently in sentiment analysis as compared to other text forms such as movie reviews.

In general, acronyms, slang expressions, and emoticons rise in importance while techniques requiring full sentence structures such as POS tagging are less effective [5]. Some approaches assume that URLs, emoticons, and hashtags carry special meanings and are therefore also considered as appropriate features [2].

4. CLASSIFYING TWEETS

Twitter has become a popular resource for sentiment analysis, as it provides a REST API[2] and a stream API[3] to retrieve tweets and therefore makes the collection of data easy. In this section the basic approach of training and testing classifiers will be outlined.

4.1 Data Set

A corpus is the starting point of each sentiment analysis, containing the data that will be used to train a classifier. In the case of supervised machine learning approaches, the data set needs to be labeled, i.e. each training element must be assigned a sentiment category (such as positive, negative, or neutral).

There are several publicly available data sets as outlined in [5]. Another approach is the collect custom data as suggested in [8].

4.2 Training

Both unsupervised and supervised classifiers are in use. Unsupervised ones rely on signal words, typical phrases, and patterns of POS tags. See [7] and [9] for more details.

However, most sentiment classifications are based on supervised learning which requires a labeled data set (e.g. positive and negative), which will be separated into a training set and a testing set. Among the most popular classifiers in sentiment analysis are the Naive Bayes and Support Vector Machine (SVM) algorithms.

Naive Bayes is a simple algorithm that generally performs well in sentiment analysis domains. It calculates the likeliness that one object belongs to a class. It has been shown that preprocessing and feature selection play an important role in order to improve the accuracy of naive bayes [13, 12].

SVM classifiers usually perform better than Naive Bayes. Their approach is to separate the positive and negative training vectors of the data set with a maximum margin [13].

4.3 Experiments and Evaluation

The evaluation of classifiers is the same as for traditional machine learning algorithms. In order to find the best classifier, the preprocessing steps described in section 3 are usually used in different combinations. Moreover, k-fold-cross-validations is a common approach where data is split into k folds, using $k-1$ folds as training data and 1 as testing set, repeated k times so that each fold will be used as testing set once.

Typically the most important benchmark figures are accuracy, precision, recall, and F-measure. The number of recognized true positives (TP), true negatives (TN), as well as false negatives (FN) and false positives (FP) are used as follows [4, 12, 11, 8, 13]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FN + FP}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F-measure} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}$$

5. RELATED WORK

Sentiment analysis of Twitter data has been the focus of various researchers. One of the results, described in [1] claims to have an average accuracy of around 60-75 % (depending on the selected features and labels). The approach of testing different machine learning algorithms (Naive Bayes, MaxEnt and SVM) combined with different features in [3] revealed an average accuracy of 80 %, however. This is underlined by the case study in [6], which experienced a similar result.

6. CONCLUSION

When analyzing tweets, specific characteristics like the limited size, slang, hashtags, targets, etc. must be considered. We have listed typical preprocessing steps for Twitter data and provided an overview of classification approaches. Since Twitter provides free access to its data, and people or organizations are interested in aggregated opinions, it is likely that sentiment analysis of tweets will become an even more popular research area in the future.

7. REFERENCES

[1] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau. Sentiment analysis of twitter data. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 30–38, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[2] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama,

[2] `https://dev.twitter.com/docs/api/1.1`

[3] `https://dev.twitter.com/docs/streaming-apis`

J. Flanigan, and N. A. Smith. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 42–47, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[3] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, pages 1–12, 2009.

[4] E. Haddi, X. Liu, and Y. Shi. The role of text pre-processing in sentiment analysis. *Procedia Computer Science*, 17:26 – 32, 2013.

[5] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis: The good the bad and the omg!, 2011.

[6] J. Lin and A. Kolcz. Large-scale machine learning at twitter. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 793–804, New York, NY, USA, 2012. ACM.

[7] B. Liu. Sentiment analysis and subjectivity. In *Handbook of Natural Language Processing, Second Edition. Taylor and Francis Group, Boca*, 2010.

[8] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[9] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[10] J. Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, ACLstudent '05, pages 43–48, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[11] M. Sokolova, N. Japkowicz, and S. Szpakowicz. Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In *AI 2006: Advances in Artificial Intelligence*, pages 1015–1021. Springer, 2006.

[12] S. Ting, W. Ip, and A. H. Tsang. Is naïve bayes a good classifier for document classification? *International Journal of Software Engineering and Its Applications*, 5(3):37–46, 2011.

[13] Q. Ye, Z. Zhang, and R. Law. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Systems with Applications*, 36(3, Part 2):6527 – 6535, 2009.