Experiments with DBpedia, WordNet and SentiWordNet as resources for sentiment analysis in micro-blogging

Hussam Hamdan[*,**,***]
hussam.hamdan@lsis-.org
*LSIS
Aix-Marseille Université CNRS
Av. Esc. Normandie Niemen,
13397 Marseille Cedex 20,
France

Frederic Béchet[**]
frederic.bechet@lif-.univ-mrs.fr
**LIF
Aix-Marseille Université CNRS
Avenue de Luminy
13288 Marseille Cedex 9,
France

Patrice Bellot[*,***]
patrice.bellot@lsis-.org
***OpenEdition
Aix-Marseille Université CNRS
3 pl. V. Hugo, case n°86
13331 Marseille Cedex 3,
France

Abstract

Sentiment Analysis in Twitter has become an important task due to the huge user-generated content published over such media. Such analysis could be useful for many domains such as Marketing, Finance, Politics, and Social. We propose to use many features in order to improve a trained classifier of Twitter messages; these features extend the feature vector of uni-gram model by the concepts extracted from DBpedia, the verb groups and the similar adjectives extracted from WordNet, the Senti-features extracted using SentiWordNet and some useful domain specific features. We also built a dictionary for emotion icons, abbreviation and slang words in tweets which is useful before extending the tweets with different features. Adding these features has improved the f-measure accuracy 2% with SVM and 4% with NaiveBayes.

1 Introduction

In recent years, the explosion of social media has changed the relation between the users and the web. The world has become closer and more "real-time" than ever. People have increasingly been part of virtual society where they have created their content, shared it, interacted with others in different ways and at a very increasingly rate. Twitter is one of the most important social media, with 1 billion tweets[1] posted per week and 637 million users[2].

With the availability of such content, it attracts the attention from who want to understand the opinion and interestingness of individuals. Thus, it would be useful in various domains such as politics, financing, marketing and social. In this context, the efficacy of sentiment analysis of twitter has been demonstrated at improving prediction of box-office revenues of movies in advance of their release (Asur and Huberman, 2010). Sentiment Analysis has been used to study the impact of 13 twitter accounts of celebrated person on their followers (Bae and Lee, 2012) and for forecasting the interesting tweets which are more probably to be reposted by the followers many times (Naveed, Gottron *et al.*, 2011).

However, sentiment analysis of microblogs faces several challenges, the limited size of posts (e.g., maximum 140 characters in Twitter), the informal language of such content containing slang words and non-standard expressions (e.g. *gr8* instead of *great*, *LOL* instead of *laughing out loud*, *goooood* etc.), and the high level of noise in the posts due to the absence of correctness verification by user or spelling checker tools.

Three different approaches can be identified in the literature of Sentiment Analysis, the first approach is the lexicon based which uses specific types of lexicons to derive the polarity of a text, this approach is suffering from the limited size of lexicon and requires human expertise to build the lexicon (Joshi, Balamurali *et al.*, 2011). The second one is machine learning approach which uses annotated texts with a given label to learn a statistical model and an early work was done on a movie review dataset (Pang, Lee et al., 2002). Both lexicon and machine learning approaches can be

[1] http://blog.kissmetrics.com/twitter-statistics/
[2] http://twopcharts.com/twitter500million.php

*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 455–459, Atlanta, Georgia, June 14-15, 2013. ©2013 Association for Computational Linguistics

combined to achieve a better performance (Khuc, Shivade et al. 2012). The third one is social approach which exploits social network properties and data for enhancing the accuracy of the classification (Speriosu, Sudan *et al.*, 2011; Tan, Lee et al. 2011; Hu, Tang *et al.*, 2013) (Hu, Tang et al., 2013) (Tan, Lee *et al.*, 2011).

In this paper, we employ machine learning. Each text is represented by a vector in which the features have to be selected carefully. They can be the words of the text, their POS tags (part of speech), or any other syntactic or semantic features.

We propose to exploit some additional features (section 3) for sentiment analysis that extend the representation of tweets by:

- the concepts extracted from DBpedia[3],
- the related adjectives and verb groups extracted from WordNet[4],
- some "social" features such as the number of happy and bad emotion icons,
- the number of exclamation and question marks,
- the existence of URL (binary feature),
- if the tweet is re-tweeted (binary feature),
- the number of symbols the tweet contains,
- the number of uppercase words,
- some other senti-features extracted from SentiWordNet[5] such as the number of positive, negative and neutral words that allow estimating a score of the negativity, positivity and objectivity of the tweets, their polarity and subjectivity.

We extended the unigram model with these features (section 4.2). We also constructed a dictionary for the abbreviations and the slang words used in Twitter in order to overcome the ambiguity of the tweets.

We tested various combinations (section 4.2) of these features, and then we chose the one that gave the highest F-measure for negative and positive classes (submission for Tweet subtask B of sentiment analysis in twitter task of SemEval2013 (Wilson, Kozareva et al. 2013)). We tested different machine learning models: Naïve Bayes, SVM, IcsiBoost[6] but the submitted runs exploited SVM only[6].

The rest of this paper is organized as follows. Section 2 outlines existing work of sentiment analysis over Twitter. Section 3 presents the features we used for training a classifier. Our experiments are described in section 4 and future work is presented in section 5.

2 Related Work

We can identify three main approaches for sentiment analysis in Twitter. The lexicon based approaches which depend on dictionaries of positive and negative words and calculate the polarity according to the positive and negative words in the text. Many dictionaries have been created manually such as ANEW (Aaffective Norms for English Words) or automatically such as SentiWordNet (Baccianella, Esuli et al. 2010). Four lexicon dictionaries were used to overcome the lack of words in each one (Joshi, Balamurali et al. 2011; Mukherjee, Malu et al. 2012). Automatically construction of a Twitter lexicon was implemented by Khuc, Shivade *et al.* (2012).

Machine learning approaches were employed from annotated tweets by using Naive Bayes, Maximum Entropy *MaxEnt* and Support Vector Machines (SVM) (Go, Bhayani et al. 2009). Go *et al.* (2009) reported that SVM outperforms other classifiers. They tried a unigram and a bigram model in conjunction with parts-of-speech (POS) features; they noted that the unigram model outperforms all other models when using SVM and that POS features decline the results. N-gram with lexicon features and microbloging features were useful but POS features were not (Kouloumpis, Wilson et al. 2011). In contrast, Pak & Paroubek (2010) reported that POS and bigrams both help. Barbosa & Feng (2010) proposed the use of syntax features of tweets like retweet, hashtags, link, punctuation and exclamation marks in conjunction with features like prior polarity of words and POS of words, Agarwal *et al.* (2011) extended their approach by using real valued prior polarity and by combining prior polarity with POS. They build models for classifying tweets into positive, negative and neutral sentiment classes and three models were proposed: a unigram model, a feature based model and a tree kernel based model which presented a new tree representation for tweets. Both combining unigrams with their features and combining the features with the tree kernel outperformed the uni-

[3] http://dbpedia.org/About
[4] http://wordnet.princeton.edu/
[5] http://sentiwordnet.isti.cnr.it/
[6] http://code.google.com/p/icsiboost/

gram baseline. Saif *et al.* (2012) proposed to use the semantic features, therefore they extracted the hidden concepts in the tweets. They demonstrated that incorporating semantic features extracted using AlchemyAPI[7] improves the accuracy of sentiment classification through three different tweet corpuses.

The third main approach takes into account the influence of users on their followers and the relation between the users and the tweets they wrote. Using the Twitter follower graph might improve the polarity classification. Speriosu, Sudan et al. (2011) demonstrated that using label propagation with Twitter follower graph improves the polarity classification. Tan, Lee et al. (2011) employed social relation for user-level sentiment analysis. Hu, Tang et al. (2013) proposed a sociological approach to handling the noisy and short text (SANT) for supervised sentiment classification, they reported that social theories such as Sentiment Consistency and Emotional Contagion could be helpful for sentiment analysis.

3 Feature Extraction

We used different types of features in order to improve the accuracy of sentiment classification.

— Bag of words (uni-gram)

The most commonly used features in text analysis are the bag of words which represent a text as unordered set of words. It assumes that words are independent from each other and also disregards their order of appearance. We used these features as a baseline model.

— Domain specific features

We extracted some domain specific features of tweets which are: presence of an URL or not, the tweet was retweeted or not, the number of "*Not*", the number of happy emotion icons, the number of sad emotion icons, exclamation and question marks, the number of words starting by a capital letter, the number of @.

— DBpedia features

We used the DBpedia Spotlight[8] Web service to extract the concepts of each tweet. For example, for the previous tweet, the DBpedia concepts for Chapel Hill are (*Settlement, PopulatedPlace, Place*). Therefore, if we suppose that people post positively about settlement, it would be more probable to post positively about Chapel Hill.

— WordNet features

We used WordNet for extracting the synonyms of nouns, verbs and adjectives, the verb groups (the hierarchies in which the verb synsets are arranged), the similar adjectives (synset) and the concepts of nouns which are related by the relation is-a in WordNet.

We chose the first synonym set for each noun, adjective and verb, then the concepts of the first noun synonym set, the similar adjectives of the first adjective synonym set and the verb group of the first verb synonym set. We think that those features would improve the accuracy because they could overcome the ambiguity and the diversity of the vocabulary.

- Senti-features

We used SentiWordNet for extracting the number and the scores of positive, negative and neutral words in tweets, the polarity (the number of positive words divided by the number of negative ones incremented by one) and subjectivity (the number of positive and negative words divided by the neutral ones incremented by one).

4 Evaluations

4.1 Data collection

We used the data set provided in SemEval 2013 for subtask B of sentiment analysis in Twitter (Wilson, Kozareva et al. 2013). The participants were provided with training tweets annotated positive, negative or neutral. We downloaded these tweets using the given script. Among 9646 tweets, we could only download 8498 of them because of protected profiles and deleted tweets. Then, we used the development set containing 1654 tweets for evaluating our methods. The method which gave the highest accuracy for the average of positive and negative classes was chosen for the submitted runs. Lastly, we combined the development set with training set and built a new model which predicted the labels of the 3813 tweets in the test set.

[7] http://www.alchemyapi.com/

[8] http://dbpedia-spotlight.github.io/

4.2 Experiments

We have done various experiments using the features presented in Section 3 with SVM model using linear kernel and the following parameters: weighting value=1, degree=3, cost=1, nu=0.5 and seed=1. We firstly constructed feature vector of tweet terms which gave 0.52% for f-measure of the negative and positive classes. Then, we augmented this vector by the similar adjectives of WordNet which improves a little the f-measure, particularly for the positive class. After that, we added the concepts of DBpedia which also improved the quality of the positive class and declined the negative one. Finally, we added all the verb groups, senti-features and domain specific features which improved the f-measure for both negative and positive classes but particularly for the positive one. Table 1 presents the results for each kind of feature vector.

Feature vector		Uni-gram	+adjectives	+DBpedia	+verb groups+ syntactic + senti-features
f-measure	**Positive**	0.603	0.619	0.622	**0.637**
	Negative	**0.443**	0.436	0.417	0.440
	Neutral	0.683	0.685	**0.691**	0.689
	Avg neg+pos	0.523	0.527	0.520	**0.538**

Table 1. The results of different feature vectors using linear SVM model (degree=3, weight=1, nu=0.5)

Feature vector		Uni-gram	+adjectives	+DBpedia	+verb groups+ syntactic + senti-features
f-measure	**Positive**	0.514	**0.563**	0.562	0.540
	Negative	0.397	0.422	**0.427**	0.424
	Neutral	0.608	**0.652**	0.648	0.636
	Avg neg+pos	0.456	0.493	**0.495**	0.482

Table 2. The results of different feature vectors using a NaiveBayes approach.

We remark that the DBpedia concepts improved the accuracy, and just the similar adjectives and group verbs of WordNet improved it, but the other synonyms and concepts declined it. The reason may be linked to a perturbation added by the synonyms. Moreover, the first synonym set is not necessary to be the most suitable one. Many domain specific and Senti-WordNet features improved the accuracy, but others did not, such as the number of neutral words, whether the tweet is reposted or not, the number of @ and the number of #. So we excluded the features that declined the accuracy.

We have done some experiments using Naive-Bayes (Table 2). Naïve Bayes improved the accuracy of the negative and positive classes, and the highest f-measure was obtained by adding the adjectives and the DBpedia concepts. Using such features improved the f-measure for the positive and negative classes: about 2% with SVM and 4% with NaiveBayes. The improvement given by means of the Naïve Bayes model was more significant than the one obtained with SVM and needed fewer features, but the higher accuracy was obtained by SVM.

5 Discussion and Future Work

In this paper we experimented the value of using DBpedia, WordNet and SentiWordNet for the sentiment classification of tweets. We extended the feature vector of tweets by the concepts of DBpedia, verb groups and similar adjectives from WordNet, the senti-features from SentiWordNet and other domain specific features. We think that using other lexicon dictionaries with SentiWordNet is more useful, we did not use POS Tagger for detecting the part of speech. We augmented the feature vector by all these features. In fact, for some tweets this expansion is not the best strategy. However, it will be important to find out a way for selecting only the features that improve the accuracy.

We verified that the adjectives are useful features and we should now focus on extracting the suitable and similar adjectives. For the abbreviation *LOL* (loud of laughing), it might be more useful to replace it by *funny* or by another adjective that reflects the sentiment of the writer. However, we could enhance our dictionary by these adjectives. We could handle the emotion icons in a similar way.

We also plan to combine the results of different classifiers for improving the total accuracy.

References

Agarwal, A., B. Xie, et al. (2011). Sentiment analysis of Twitter data.Proceedings of the Workshop on Languages in Social Media. Portland, Oregon, Association for Computational Linguistics: 30-38.

Asur, S. and B. A. Huberman (2010). Predicting the Future with Social Media. Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01, IEEE Computer Society: 492-499.

Baccianella, S., A. Esuli, et al. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10), European Language Resources Association (ELRA).

Bae, Y. and H. Lee (2012). "Sentiment analysis of twitter audiences: Measuring the positive or negative influence of popular twitterers." J. Am. Soc. Inf. Sci. Technol.63(12): 2521-2535.

Barbosa, L. and J. Feng (2010). Robust sentiment detection on Twitter from biased and noisy data.Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Beijing, China, Association for Computational Linguistics: 36-44.

Go, A., R. Bhayani, et al. (2009). Twitter Sentiment Classification using Distant Supervision.

Hu, X., L. Tang, et al. (2013). Exploiting social relations for sentiment analysis in microblogging. Proceedings of the sixth ACM international conference on Web search and data mining.Rome, Italy, ACM: 537-546.

Joshi, A., A. R. Balamurali, et al. (2011). C-Feel-It: a sentiment analyzer for micro-blogs. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations. Portland, Oregon, Association for Computational Linguistics: 127-132.

Khuc, V. N., C. Shivade, et al. (2012). Towards building large-scale distributed systems for twitter sentiment analysis. Proceedings of the 27th Annual ACM Symposium on Applied Computing. Trento, Italy, ACM: 459-464.

Kouloumpis, E., T. Wilson, et al. (2011).Twitter Sentiment Analysis: The Good the Bad and the OMG! Fifth International AAAI Conference on Weblogs and Social Media.

Mukherjee, S., A. Malu, et al. (2012).TwiSent: a multistage system for analyzing sentiment in twitter. Proceedings of the 21st ACM international conference on Information and knowledge management.Maui, Hawaii, USA, ACM: 2531-2534.

Naveed, N., T. Gottron, et al. (2011). Bad News Travels Fast: A Content-based Analysis of Interestingness on Twitter. Proc. Web Science Conf.

Pak, A. and P. Paroubek (2010). Twitter as a corpus for sentiment analysis and opinion mining.

Pang, B., L. Lee, et al. (2002). Thumbs up?: sentiment classification using machine learning techniques. Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10, Association for Computational Linguistics: 79-86.

Saif, H., Y. He, et al. (2012).Semantic sentiment analysis of twitter.Proceedings of the 11th international conference on The Semantic Web - Volume Part I. Boston, MA, Springer-Verlag: 508-524.

Speriosu, M., N. Sudan, et al. (2011). Twitter polarity classification with label propagation over lexical links and the follower graph. Proceedings of the First Workshop on Unsupervised Learning in NLP. Edinburgh, Scotland, Association for Computational Linguistics: 53-63.

Tan, C., L. Lee, et al. (2011). User-level sentiment analysis incorporating social networks. Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. San Diego, California, USA, ACM: 1397-1405.

Khuc, V. N., C. Shivade, et al. (2012). Towards building large-scale distributed systems for twitter sentiment analysis. Proceedings of the 27th Annual ACM Symposium on Applied Computing. Trento, Italy, ACM: 459-464.

Wilson, T., Z. Kozareva, et al. (2013). "SemEval-2013 Task 2: Sentiment Analysis in Twitter." Proceedings of the 7th International Workshop on Semantic Evaluation. Association for Computational Linguistics.