FBK: Sentiment Analysis in Twitter with Tweetsted

Md. Faisal Mahbub Chowdhury
FBK and University of Trento, Italy
fmchowdhury@gmail.com

Marco Guerini
Trento RISE, Italy
marco.guerini@trentorise.eu

Sara Tonelli
FBK, Trento, Italy
satonelli@fbk.eu

Alberto Lavelli
FBK, Trento, Italy
lavelli@fbk.eu

Abstract

This paper presents the *Tweetsted* system implemented for the SemEval 2013 task on Sentiment Analysis in Twitter. In particular, we participated in Task B on Message Polarity Classification in the Constrained setting. The approach is based on the exploitation of various resources such as SentiWordNet and LIWC. Official results show that our approach yields a F-score of 0.5976 for Twitter messages (11th out of 35) and a F-score of 0.5487 for SMS messages (8th out of 28 participants).

1 Introduction

Microblogging is currently a very popular communication tool where millions of users share opinions on different aspects of life. For this reason it is a valuable source of data for opinion mining and sentiment analysis.

Working with such type of texts presents challenges for NLP beyond those typically encountered when dealing with more traditional texts, such as newswire data. Tweets are short, the language used is very informal, with creative spelling and punctuation, misspellings, slang, new words, URLs, genre-specific terminology and abbreviations, and #hashtags. These characteristics need to be handled with specific approaches.

This paper presents the approach adopted for the SemEval 2013 task on Sentiment Analysis in Twitter, in particular Task B on Message Polarity Classification in the Constrained setting (i.e., using the provided training data only).

The goal of Task B on Message Polarity Classification is the following: given a message, decide whether it expresses a positive, negative, or neutral sentiment. For messages conveying both a positive and a negative sentiment, whichever is the stronger sentiment should be chosen.

Two modalities are possible: (1) Constrained (using the provided training data only; other resources, such as lexica, are allowed; however, it is not allowed to use additional tweets/SMS messages or additional sentences with sentiment annotations); and (2) Unconstrained (using additional data for training, e.g., additional tweets/SMS messages or additional sentences annotated for sentiment). We participated in the Constrained modality.

We adopted a supervised machine learning (ML) approach based on various contextual and semantic features. In particular, we exploited resources such as SentiWordNet (Esuli and Sebastiani, 2006), LIWC (Pennebaker and Francis, 2001), and the lexicons described in Mohammad et al. (2009).

Critical features include: whether the message contains intensifiers, adjectives, interjections, presence of positive or negative emoticons, possible message polarity based on SentiWordNet scores (Esuli and Sebastiani, 2006; Gatti and Guerini, 2012), scores based on LIWC categories (Pennebaker and Francis, 2001), negated words, etc.

2 System Description

Our supervised ML-based approach relies on Support Vector Machines (SVMs). The SVM implementation used in the system is LIBSVM (Chang

*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 466–470, Atlanta, Georgia, June 14-15, 2013.

and Lin, 2001) for training SVM models and testing. Moreover, in the preprocessing phase we used TweetNLP (Owoputi et al., 2013), a POS tagger explicitly tailored for working on tweets.

We adopted a 2 stage approach: (1) during stage 1, we performed a binary classification of messages according to the classes *neutral* vs *subjective*; (2) in stage 2, we performed a binary classification of subjective messages according to the classes *positive* vs *negative*. We performed various experiments on the training and development sets exploring the use of different features (see Section 2.1) to find the best configurations for the official submission.

2.1 Feature list

We implement several features divided into three groups: contextual features, semantic features from context and semantic features from external resources. The complete list is reported in Table 1.

Contextual features are features computed by considering only the tokens in the tweets/SMS and the associated part of speech.

Semantic Features from Context are features based on words polarity. Emoticons were recognized through a list of emoticons extracted from Wikipedia[1] and then manually labeled as positive or negative. Negated words (feature n. 18) are any token occurring between *n't, not, no* and a comma, excluding those tagged as function words. Feature n. 19 captures tokens (or sequences of tokens) labeled with a positive or negative polarity in the resource described in Mohammad et al. (2009). The intensifiers considered for Feature n. 20 have been identified by implementing a simple algorithm that detects tokens containing anomalously repeated characters (e.g. *happyyyyy*). Feature n. 21 was computed by training the system on the training data and predicting labels for the test data, and then using these labels as new features to train the system again.

Semantic Features from external resources include word classes from the Linguistic Inquiry and Word Count (LIWC), a tool that calculates the degree to which people use different categories of words related to psycholinguistic processes (Pennebaker and Francis, 2001). LIWC includes about 2,200 words and stems grouped into 70 broad categories relevant to psychological processes (e.g., EMOTION, COGNITION). Sample words are shown in Table 2.

For each non-zero valued LIWC category of a corresponding tweet/SMS, we added a feature for that category and used the category score as the value of that feature. We call this *LWIC string* feature. Alternatively, we also added a separate feature for each non-zero valued LIWC category and set 1 as the value of that feature. This feature is called *LWIC boolean*.

We also used words prior polarity - i.e. if a word out of context evokes something positive or negative. For this, we relied on SentiWordNet, a broad-coverage resource that provides polarities for (almost) every word. Since words can have multiple senses, we compute the prior polarity of a word starting from the polarity of each sense and returning its *polarity strength* as an index between -1 and 1. We tested 14 formulae that combine posterior polarities in different ways to obtain a word prior polarity, as reported in (Gatti and Guerini, 2012).

For the *SWNscoresMaximum* feature, we select the prior polarity of the word in a tweet/SMS having the maximum absolute score among all words (of that tweet/SMS). For *SWNscoresPolarityCount*, we select the polarity (positive, negative or neutral) that is assigned to the majority of the words. As for *SWNscoresSum*, it corresponds to the sum of the prior polarities associated with all words in the tweet/SMS.

3 Experimental Setup

In order to select the best performing feature set, we carried out several 5-fold cross validation experiments on the training data. We report in Table 3 the best performing feature set. In particular, we adopted a 2 stage approach:

1. during the first stage we performed a binary classification of messages according to the classes *neutral* vs *subjective*;
2. in the second stage, we performed a binary classification of subjective messages according to the classes *positive* vs *negative*.

We opted for a two stage binary classification approach, since we observed that it produces slightly

[1] http://en.wikipedia.org/wiki/List_of_emoticons

Contextual Features	
1. noOfAdjectives	num
2. adjective list	string
3. interjection list	string
4. firstInterj	string
5. lastInterj	string
6. bigramList	string
7. beginsWithRT	boolean
8. hasRTinMiddle	boolean
9. endsWithLink	boolean
10. endsWithHashtag	boolean
11. hasQuestion	boolean
Semantic Features from Context	
12. noOfPositiveEmoticons	num
13. noOfNegativeEmoticons	num
14. beginsWithPosEmoticon	boolean
15. beginsWithNegEmoticon	boolean
16. endsWithPosEmoticon	boolean
17. endsWithNegEmoticon	boolean
18. negatedWords	string
19. indexOfChunksWithPolarity	string
20. containsIntensifier	boolean
21. labelPredictedBySystem	pos./neg./neut.
Semantic Features from External Resources	
22. LIWC string	string
23. LIWC boolean	string
24. SWNscoresMaximum	pos./neg./neut.
25. SWNscoresPolarityCount	pos./neg./neut.
26. SWNscoresSum	pos./neg./neut.

Table 1: Complete feature list.

LABEL	*Sample words*
CERTAIN	all, very, fact*, exact*, certain*, completely
DISCREP	but, if, expect*, should
TENTAT	or, some, may, possib*, probab*
SENSES	observ*, discuss*, shows, appears
SELF	we, our, I, us
SOCIAL	discuss*, interact*, suggest*, argu*
OPTIM	best, easy*, enthus*, hope, pride
ANGER	hate, kill, annoyed
INHIB	block, constrain, stop

Table 2: Word categories along with sample words

better results than a single stage multi-class approach (i.e. *neutral* vs *positive* vs *negative*).[2] Different combinations of classifiers were explored obtaining comparable results. Here we will report only the best results.

STAGE 1. The best result for stage (1), *neutral* vs *subjective*, obtained with 5-fold cross validation on training set only, accounts for an accuracy of 69.6%. Instead, the best result for stage (1), obtained with training on training data and testing on development data, accounts for an accuracy of 72.67%.

The list of best features is reported in Table 3. Feature selection was performed by starting from a small set of basic features, and then by adding the remaining features incrementally.

Contextual Features	
2. adjective list	string
3. interjection list	string
5. lastInterj	string
Semantic Features from Context	
12. noOfPositiveEmoticons	num
13. noOfNegativeEmoticons	num
18. negatedWords	string
19. indexOfChunksWithPolarity	string
20. containsIntensifier	boolean
Semantic Features from external resources	
23. LIWC boolean	string
24. SWNscoresMaximum	posi./neg./neut.

Table 3: Best performing feature set.

STAGE 2. In stage (2), *positive* vs *negative*, we started from the best feature set obtained from stage (1) and added the remaining features one by one incrementally. In this case, we kept *SWNscoresMaximum* without testing again other formulae; in particular, to compute words prior polarity, we also kept the *first sense* approach, that assigns to every word the SWN score of its most frequent sense and proved to be the most discriminative in the first stage *neutral* vs. *subjective*. We found that none of the feature sets produced better results than that obtained using the best feature set selected from stage (1). So, the best feature set for stage (2) is unchanged. We trained the system on the training data and tested it on the development data, achieving an accuracy of 80.67%.

4 Evaluation

The SemEval task organizers (Wilson et al., 2013) provided two test sets on which the systems were to be evaluated: one included Twitter messages, i.e. the same type of texts included in the training set,

[2]The average F-scores (pos and neg) for two stage and single stage approaches obtained using the official scorer, by training on the training data and testing on the development data, are 0.5682 and 0.5611 respectively.

while the other comprised SMS messages, i.e. texts having more or less the same length as the Twitter data but (supposedly) a different style. We applied the same model, trained both on the training and the development set, on the two types of data, without any specific adaptation.

The *Twitter test set* was composed of 3,813 tweets. Official results show that our approach yields an F-score of 0.5976 for Twitter messages (11th out of 35), while the best performing system obtained an F-score of 0.6902. The confusion matrix is reported in Table 4, while the score details in Table 5. The latter table shows that our system achieves the lowest results on negative tweets, both in terms of precision and of recall.

gs/pred	positive	negative	neutral
positive	946	101	525
negative	90	274	237
neutral	210	70	1360

Table 4: Confusion matrix for Twitter task

class	prec	recall	F-score
positive	0.7592	0.6018	0.6714
negative	0.6157	0.4559	0.5239
neutral	0.6409	0.8293	0.7230
average(pos and neg)			0.5976

Table 5: Detailed results for Twitter task

The *SMS test set* for the competition was composed of 2,094 SMS. Official results provided by the task organizers show that our approach yields an F-score of 0.5487 for SMS messages (8th out of 28 participants), while the best performing system obtained an F-score of 0.6846. The confusion matrix is reported in Table 6, while the score details in Table 7. Also in this case the recognition of negative messages achieves by far the poorest performance.

A comparison of the results on the two test sets shows that, as expected, our system performs better on tweets than on SMS. However, precision achieved by the system on neutral SMS is 0.12 points better on text messages than on tweets.

Interestingly, it appears from the results in Tables 5 and 7 (and from the distribution of the classes in the data sets) that there may be a correlation between the number of tweets/SMS for a particular class and the performance obtained for such class. We plan to further investigate this issue.

gs/pred	positive	negative	neutral
positive	320	44	128
negative	66	171	157
neutral	208	64	936

Table 6: Confusion matrix for SMS task

class	prec	recall	F-score
positive	0.5387	0.6504	0.5893
negative	0.6129	0.4340	0.5082
neutral	0.7666	0.7748	0.7707
average(pos and neg)			0.5487

Table 7: Detailed results for SMS task

5 Conclusions

In this paper, we presented *Tweetsted*, the system developed by FBK for the SemEval 2013 task on Sentiment Analysis. We trained a classifier performing a two-step binary classification, i.e. first neutral vs. subjective data, and then positive vs. negative ones. We implemented a set of features including contextual and semantic ones. We also integrated in our feature representation external knowledge from SentiWordNet, LIWC and the resource by Mohammad et al. (2009). On both test sets (i.e., Twitter messages and SMS) of the constrained modality of the challenge, we achieved a good performance, being among the top 30% of the competing systems. In the near future, we plan to perform an error analysis of the wrongly classified data to investigate possible classification issues, in particular the lower performance on negative tweets and SMS.

Acknowledgments

This work is supported by "eOnco - Pervasive knowledge and data management in cancer care" and "Trento RISE PerTe" projects.

References

Chih-Chung Chang and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

A. Esuli and F. Sebastiani. 2006. SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy.

Lorenzo Gatti and Marco Guerini. 2012. Assessing sentiment strength in words prior polarities. In *Proceedings of COLING 2012: Posters*, pages 361–370, Mumbai, India, December. The COLING 2012 Organizing Committee.

Saif Mohammad, Bonnie Dorr, and Cody Dunne. 2009. Generating High-Coverage Semantic Orientation Lexicons From Overtly Marked Words and a Thesaurus. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL 2013*, Atlanta, Georgia, June.

J. Pennebaker and M. Francis. 2001. Linguistic inquiry and word count: LIWC. Erlbaum Publishers.

Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Sara Rosenthal, Veselin Stoyanov, and Alan Ritter. 2013. SemEval-2013 task 2: Sentiment analysis in twitter. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '13, June.