Introducing The Semantics In Sentiment Analysis On Twitter Using WordNet

Ines Mostefai
Department of computer science
EEDIS Laboratory, University Djillali Liabes
Sidi Bel-Abbes, Algeria
mostefai.ines@univ-sba.dz

Zakaria Elberrichi
Department of computer science
EEDIS Laboratory, University Djillali Liabes
Sidi Bel-Abbes, Algeria
elberrichi@gmail.com

Abstract— **Social networks are an excellent source of information, and opinion extraction. The present work shows the introducing of the semantics for sentiment analysis on Twitter using the Machine Learning Approach and WordNet lexical database. The best performance was obtained using the SVM classifier for the machine learning approach with a very good F-measure of 90.75%.**

Sentiment analysis; Twitter; Machine Learning; WordNet; SVM; Naïve Bayes.

I. INTRODUCTION

The advent of online social networks is one of the most exciting events of this decade. Many social networking sites such as Twitter, Facebook and Tumblr have become very popular, particularly, with the events that world knows. These Social networks are rich of all kinds of data, and this leads the researchers to explore them for analysis. Social networks allow users to engage in textual social interactions such as sharing written comments and opinions. For example, if someone wants to purchase a product from a supplier, he can share this experience and his/her opinions about the product or the service with the other users. However, if someone wants to buy a product or a service, but he is uncertain or hesitant, he can do a research, and read the comments and the observations of other users before deciding. Thus, the enterprises that want to keep their market shares and increase their added value have interest to follow and monitor the public opinion.

This article addresses the problem of sentiment analysis on Twitter and presents a certain number of experiments conducted on this issue, using the machine learning approach, for which we have selected two classifiers among the most popular in supervised learning, particularly in the classification of sentiments [5] [9] [10], namely, Naïve Bayes and Support Vecteur Machine (SVM), in order to compare their performances in this domain. Then, we have introduced semantics using the semantic relations of the lexical database WordNet (eg Antonyms, Synonyms, hypernyms …etc.).

The rest of the present work is organized as follows. In the second section, we will present some the existing works that are mostly cited in the literature. In the third section, we will describe the corpus on which we did the experiments. Then in the fourth section, we describe the very important pretreatment procedure applied on this corpus. In the fifth section, we will discuss of an original experiment characterized by introducing the semantic aspect using the WordNet thesaurus, so we will describe the functions used. The findings of the experiments are well presented in the sixth section. Finally, we finish this work by a conclusion.

II. RELATED WORKS

In this section, we will present some works done in this area.

Reference [5] have made a binary classification (positive vs. negative) on a set of tweets prior labeled based on emoticons. To extract informative characteristics, they have used unigrams, bigrams, both unigram and bigram, and finally unigrams with their POS tag (part of speech). For learning they used the classifiers Naïve Bayes, SVM and MaxEntropie. Their findings show that the model constructed from MaxEntropie using both unigrams and bigrams gives better results. Around 3% more than the other models.

Unlike [5], [9] have taken into account three (03) classes of sentiments, namely, positive, negative and neutral. For the analysis, they have used two sets of data, one for the subjective data collected in the same way that [5] and the other for the objective data. Before, they did a linguistic statistical analysis on the corpus. From this analysis, they have concluded that the objectives texts contain more proper noun and common noun, while subjective texts often use more personal pronouns. To train the Naive Bayes classifier, they used the N-grams and POS tags. Unlike [5], their results reveal that adding the POS tag to bigrams improves the accuracy of Tweets sentiments classification.

Reference [3] have shown that the use of n-grams decreases the performance of the classifier due to the large number of infrequent words in Twitter. Therefore, they have proposed the use of micro-blogging functions such as retweets, hashtags, punctuation and emoticons. They have found that the use of these functions to train the SVM classifier increases the accuracy of the sentiments classification by 2.2% compared to the model formed only with the unigrams.

Reference [7] have built a set of training data based on the Hashtag function of Twitter to identify positive, negative and neutral tweets. For the sentiment classification, they have used the microblogging characteristics including emoticons, abbreviations and the presence of intensifier as capitalized words and repeated characters. Their results show that the best performance comes with the use of n-grams together with the characteristics of micro-blogging and lexical features, i.e., the use of prior polarity of words tagged. Unlike [9], the addition of the POS function decreased performance.

The originality introduced by [10] for sentiment analysis on Twitter, is the addition of semantic features for classification. They have proposed three (03) approaches to incorporate semantics, namely, by replacement, augmentation and interpolation. Their results reveal that the semantic model outperforms unigram models and POS for identifying the positive and negative sentiments. They have also found that the interpolation method gives the most accurate result among the three methods of semantic incorporation with an average F-measure of 75.95%.

III. Description of the Corpus

This section discusses the characteristics and peculiarities of the data set on which we applied our experiments, but before, we will define what a Tweet is, what the characteristics of Twitter are, and why most analysts prefer this platform.

Twitter is an open microblogging platform, which allows us to send real-time short messages called Tweets. These allow us to follow the news and the today's opinions of others. In 2013, Twitter counts over 500 million users around the world with an average of 58 million tweets sent per a day. Furthermore, Twitter has a search API, which allows us to collect a great amount of tweets for analysis. We can also specify the language and the topic of these tweets. Another advantage of this platform is, it is ubiquitous and accessible from different types of media devices including Smartphones. Here are some specific functions to twitter :

- **Tweet** is the message sent on Twitter and it is restricted to 140 characters.
- **ReTweet** (RT/VIA) is a message broadcasted by another user. If you find an interesting tweet, you can share it via using the Retweet function. There are two ways for indicating the origin of the tweet, whether through using the term RT followed the author identifier, or through the term VIA.
- **Mention** (@) is used to refer to another user.
- **Hashtags** (#) are tags or keywords that allow us to do a research on a specific topic.

The following figure shows an example of a ReTweet which is posted by @TwittyAlgeria, its origin is the user @devoirdesavoir and covers the topic of Ramadan in Algeria.

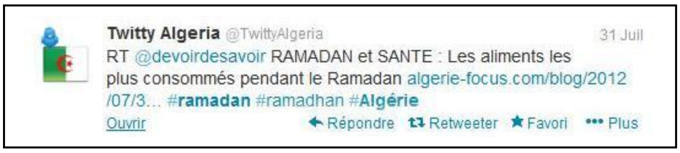


Figure 1. Example of a Tweet

The data set on which we conducted our experiment is that of [1], this set contains the original 5 127 tweets written in different languages and manually annotated according to three sentiment class, namely, positive, negative and neutral. Since our experiments focus only on the English language, a filtering operation is necessary. For this, we have used two libraries LangID, which is an autonomous tool for identifying languages and CLD (Compact language Detector) Google (included in the Google Chrome browser), which detects the language of a given text (text or HTML). The choice of the language of Tweet in the filtering operation is made on the basis of the best accuracy calculated by these two libraries. After the filtering operation, the total number of tweets has been reduced to 3,763 tweets with 1 345 tweets positive, 1 224 negative and 1 194 neutral, respectively, 36%, 32% and 32% of the overall total.

IV. Pretreatment

Unlike reviews and articles, tweets are usually written in an informal language because of their limited number of characters. Twitter users often use abbreviations, emoticons and slang to convey their messages and express their sentiments. Consequently, a preprocessing step is necessary to clean and prepare the data for the learning stage [13]. Our purpose is to process a tweet and rewrite it, as close as possible, in a formal language.

This section explains this processing, but before this, we have created two dictionaries based on some resources, one for abbreviations and the other for the usual textual emoticons. These dictionaries will be helpful in this pretreatment phase. Extracts of these dictionaries are as follows:

```
"abbreviation";"meaning"
"<3";"heart"
"a3";"anyplace anywhere anytime"
"ab";"about"
"adn";"any day now"
"afaik";"as far as i know"
"afk";"away from keyboard"
"aka";"also known as"
"app";"application"
"apps";"applications"
```

Figure 2. Abbreviations' Extract Dictionary

```
"emoticone";"status"
"(:";"happy"
"(<_>)";"sad"
"(=";"happy"
"(^_^)";"happy"
"(happy)";"happy"
"(sad)";"sad"
")-:";"sad"
"):";"sad"
"*-*";"happy"
```

Figure 3. Emoticons' Extract Dictionary

The pretreatment steps are as follows:

First, we have replaced the emoticons by their status as it is shown above in the extract of the emoticons' dictionary and the abbreviations have been replaced by their whole words. Then we normalized all tweets by converting them into lower case form and we have replaced the functions of Twitter by simple labels as it is shown in the following table:

TABLE I. LABELS USED FOR FUNCTIONS OF TWITTER

The function	The label
The user identifier	USER
Web links	URL
The hashtags	TAG

Concerning the contracted forms, we have used regular expression to transform them into their non-contracted forms. We have also rewrote words which contains the consecutive and repeated characters and chains more than twice in their correct forms. For this, we have used regular expressions and an English dictionary. In the same context, we have also eliminated the VIA and RT functions, the numbers, the words that begin with numbers (e.g. 9:00pm) and punctuation marks. Table 2 shows some examples of tweets before and after pretreatment.

At the end of this step, we have stored the pretreated tweets in a MySQL database to use them later in the learning stage.

TABLE II. THE EXAMPLES OF TWEETS BEFORE AND AFTER PRETREATMENT

Before	After
@mommyanasays LMFAOOO SMH! i wish i can retweet this...but **ur** tweets are not retweetable **ugh**!	**USER lmfao shaking my head** i wish i can retweet this but **your** tweets are not retweetable **disgusted**
Omg stop it got me over here ctfu!! RT **@ModernairVanity**: **#oneofmyfollowers** got a boo **that's** big enuff to smother his ass **lmao**	**oh my god** stop it got me over here ctfu **USER TAG** got a boo **that is** big enuff to smother his ass **laughing my ass off**
RT **@COSTARRICENSE** - Tormenta Tropical NICOLE - **http://t.co/3MUg33B**: Tormenta Tropical NICOLE - **http://t.co/3MUg33B** **http://bit.ly/9S6LOn**	**USER** tormenta tropical nicole **URL** tormenta tropical nicole **URL URL**

V. THE APPROACH BASED ON SUPERVISED LEARNING MACHINE

The Approach Based on Supervised Learning Machine uses automatic machine learning techniques to train classifiers on a set of training data. Then, apply the models generated by the classifiers on a set of test data in order to evaluate and compare their performances. For our experiments, we have selected two well known classifiers in the field of sentiment analysis, namely, Naive Bayes and Support Vector Machine (SVM). To realize our experiments, we have defined a set of elementary functions, in order to better see their impact on the classifiers; you will find their descriptions in the table below. We have also introduced the semantics using semantic relations of WordNet (Antonymy, synonymy...etc.).

TABLE III. A SET OF ELEMENTARY FUNCTIONS WITH THEIR DESCRIPTIONS

The function	Description
F1	The basic function, we take the pretreated tweets and we eliminate the functions of Twitter (i.e USER TAG and URL).
F2	Apply the lemmatization on the pretreated tweets.
F3	Apply the stemming on pretreated tweets; the chosen algorithm is the porter stemmer.
F4	Replace the negative form contained in tweets by their antonyms using WordNet (e.g. not happy =>unhappy.)
F5	Replace the words by their synonyms using WordNet.
F6	Replace the words by their hypernyms using WordNet.
F7	Replace the words by their hyponyms using WordNet.
F8	Add to the words their synonyms and their hypernyms together using WordNet.
F9	Add to the words their synonyms and their hyponyms using WordNet.
F10	Add to the words their hypernyms and hyponyms using WordNet.
F11	Add to the words their synonyms, hypernyms and hyponyms using WordNet.

We did several experiments combining the functions described above before the feature extraction in order to enrich the set of features, then we have applied the classifiers on the training set, recording each time the obtained results. Concerning the evaluation of models, we have calculated the accuracy, precision, recall and F-measure where you find their equations below and for validation, we have use the technique of n-fold cross-validation with n=10. The obtained results are presented in the following section.

$$\textbf{Accuracy} = |\text{ correct tweets }|/|\text{ all tweets }| \quad (1)$$

$$\textbf{Precision} = |\text{ correct tweets }|/|\text{ retrieved tweets }| \quad (2)$$

$$\textbf{Recall} = |\text{correct tweets}|/|\text{ relevant tweets }| \quad (3)$$

$$\textbf{F-measure} = 2.\text{Precision}.\text{Recall}/(\text{Precision}+\text{Recall}) \quad (4)$$

VI. THE RESULTS

A. Naïve Bayes classifier

The first function consists to take the pretreated tweets and eliminate the specific functions to Twitter. The first generated model using this function gives an accuracy of 84.14%.

By adding the function of lemmatization and stemming, the model performance decreases by 1.89% compared to the first model.

The transformation of the negation form contained in tweets by their antonyms using WordNet increases slightly the performance of the first model by 0.02%. For the rest of the experiments; we consider this model (No. 5 in the table) as a baseline. Replacing the words contained in the tweets by their synonyms using WordNet decreases the performances of the previous model by 1.01%. Replacing the words contained in the tweets by their hypernyms degrade the performance of the model by 4.10%. Replacing the words contained in the tweets by their hyponyms also degrade the performance of the baseline model by 2.77%.

In order to enrich the content of the tweets, we will add to the words their synonyms and hypernyms together. The generated model is less efficient than the baseline model by 1.70%. Repeating the same experiment, but using hyponyms instead of hypernyms, and the resulting model degrades more than the previous model. We record a decrease of accuracy of 2.26% compared to the baseline model. Similarly, combining the hypernyms and hyponyms with the words without using the synonymy relationship, decrease the performances by 1.75%.

Finally, we add all the three previous relationships to the words. This has a negative influence on the generated model of 2.29%.

TABLE IV. FINDINGS ACHIEVED BY THE NAIVE BAYES CLASSIFIER IN DIFFERENT CASES

N°	The Funct.	Nb Feat.	Acc. (%)	Prec. (%)	Rec. (%)	F-meas. (%)
01	F1	*7770*	*84.14*	*84.18*	*84.00*	***84,09***
02	F1+F2	*6701*	*82.78*	*82.84*	*82.65*	***82,74***
03	F1+F3	*6318*	*82.86*	*82.95*	*82.71*	***82,83***
04	F1+F2+F3	*6239*	*82.25*	*82.28*	*82.11*	***82,19***
05	F1+F4	*7783*	*84.16*	*84.18*	*84.03*	***84,10***
06	F1+F4+F5	*6519*	*83.15*	*83.23*	*83.03*	***83,13***
07	F1+F4+F6	*5534*	*80.06*	*80.11*	*79.87*	***79,99***
08	F1+F4+F7	*6524*	*81.39*	*81.61*	*81.34*	***81,47***
09	F1+F4+F8	*9515*	*82.46*	*82.68*	*82.46*	***82,57***
10	F1+F4+F9	*9727*	*81.90*	*82.21*	*81.89*	***82,05***
11	F1+F4+F10	*9690*	*82.41*	*82.67*	*82.40*	***82,53***
12	F1+F4+F11	*10465*	*81,87*	*82,21*	*81,88*	***82,04***

B. SVM

Comparing the first generated models of Naïve Bayes to that of SVM, the SVM generated models surpass the first ones by approximately 5.40%.

As in the case of the Naive Bayes, the lemmatization and stemming decrease the performance of the model. So no need to lemmatize or stem for tweets.

The use of antonyms for the negation forms increases slightly the accuracy of the model. In what follows, we will consider this model as a baseline for comparison.

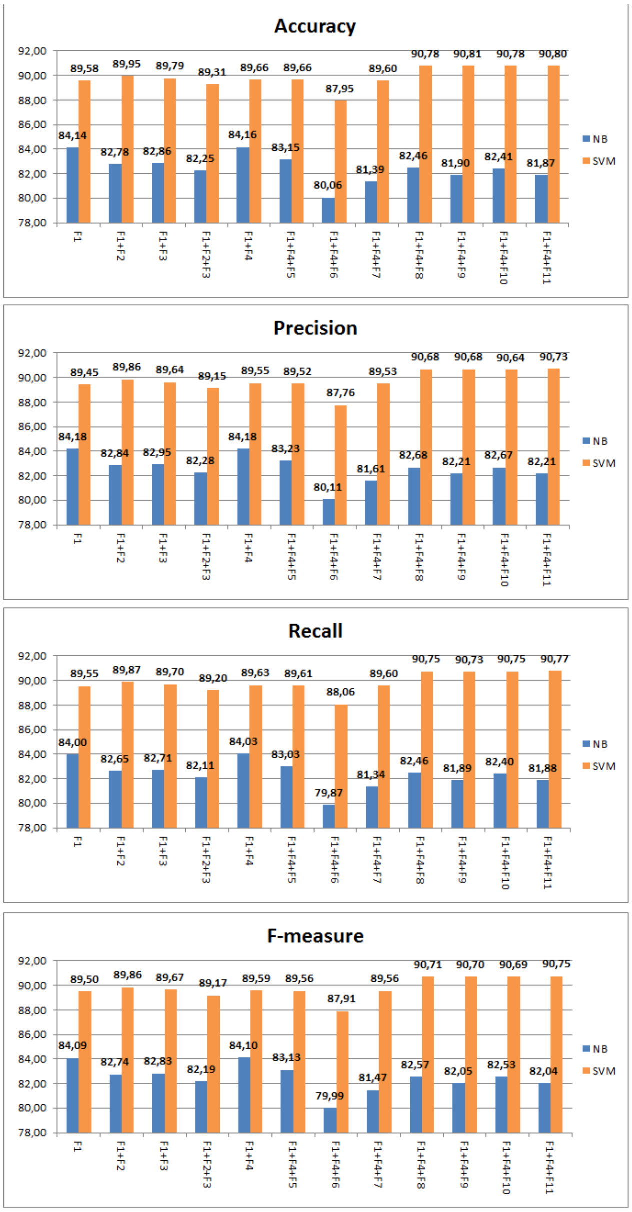
The use of antonyms for the negation forms increases slightly the accuracy of the model. In what follows, we will consider this model as a baseline for comparison.

The use of synonyms does not affect the accuracy of the model. In contrast, the use of hypernyms degrades the model. In the case of hyponyms, the model degrades but slightly with 0.06%. So, as the case of Naive Bayes the synonymy relationship is better than other relationships, even though its gain is not that interesting.

Unlike to the Naive Bayes classifier, the use of synonyms together with hypernyms (model No. 9) increases the performance of the model by 1.12%. Similarly, when we use hyponyms we have an increase of 1.15%.Comparing the two relations hypernymy and hyponymy, we find that the use of the latter relationship is better. Using the hypernymy relationship together with hyponymy relationship without introducing synonymy relationship will negatively affects the performance of the previous models. The use of all these relationships together positively affects the performance of the model with an increase of 1.14% compared to the baseline model.

TABLE V. FINDINGS ACHIEVED BY THE SVM CLASSIFIER IN DIFFERENT CASES

N°	The Funct.	Nb Feat.	Acc. (%)	Prec. (%)	Rec. (%)	F-meas. (%)
01	F1	*7770*	*89.58*	*89.45*	*89.55*	***89,50***
02	F1+F2	*6701*	*89.95*	*89.86*	*89.87*	***89,86***
03	F1+F3	*6318*	*89.79*	*89.64*	*89.70*	***89,67***
04	F1+F2+F3	*6239*	*89.31*	*89.15*	*89.20*	***89,17***
05	F1+F4	*7783*	*89.66*	*89.55*	*89.63*	***89,59***
06	F1+F4+F5	*6519*	*89.66*	*89.52*	*89.61*	***89,56***
07	F1+F4+F6	*5534*	*87.95*	*87.76*	*88.06*	***87,91***
08	F1+F4+F7	*6524*	*89.60*	*89.53*	*89.60*	***89,56***
09	F1+F4+F8	*9515*	*90.78*	*90.68*	*90.75*	***90,71***
10	F1+F4+F9	*9727*	*90.81*	*90.68*	*90.73*	***90,70***
11	F1+F4+F10	*9690*	*90.78*	*90.64*	*90.75*	***90,69***
12	F1+F4+F11	*10465*	*90,80*	*90,73*	*90,77*	***90,75***



The graphics below show the variation of each measure of evaluation by model and classifier.

Figure 4. Comparison between NB and SVM according the different evaluation measures

VII. Conclusion

Firstly, we find that the SVM classifier is better for the classification of sentiment in Twitter than the Naive Bayes classifier. The incorporation of semantic characteristics had a better influence for the SVM classifier than the Naive Bayes classifier.

The application of lemmatization and stemming on the tweets degrades the performance for both classifiers. We think, we can safely conclude, you don't need to perform them in the special case of the tweets.

We found that using antonymy relationship for the negation form increases the performance of the two classifiers.

The best accuracy recorded for the Naive Bayes classifier is 84.16%, when we used the antonymy relationship for negation form. In the case of the SVM, the best accuracy is obtained by the model N°10 with 90.81%, where we have used the antonyms, synonyms and hyponyms together.

Comparing these semantic relationships, we have found that you can use only the relations synonyms and antonyms, and forget the others semantic relations.

References

[1] A. Agarwal, X. Boyi, V. Ilia, R. Owen, and P. Rebecca, "Sentiment Analysis of Twitter Data" , in Proceedings of the Workshop on Language in Social Media, Association for Computational Linguistics, 2011. pp. 30–38.

[2] S. Baccianella, E. Andrea, and S. Fabrizio, "SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining" , Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), 2010, pp. 2200–2204.

[3] L. Barbosa, and J. Feng, "Robust sentiment detection on Twitter from biased and noisy Data" , Proceedings of the 23rd International Conference on Computational Linguistics: Posters, COLING '10, ,AT&T Labs-Research, 2010, pp. 36–44.

[4] A.N. Finn, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs" , Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts', 2011, pp. 93–98.

[5] A. Go, B. Richa, and H. Lei, "Twitter sentiment classification using distant supervision" , In Processing, 2009, pp.1–6.

[6] V.N. Khuc, C. Shivade, R. Ramnath, and J. Ramanathan, "Towards Building Large-Scale Distributed Systems for Twitter Sentiment Analysis", Proceedings of the 27th Annual ACM Symposium on Applied Computing SAC '12, 2012, pp. 459–464.

[7] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter Sentiment Analysis: The Good the Bad and the OMG!" , Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, 2011, pp. 538–541.

[8] A. Pak, theses "Automatic, Adaptive and Applicative Sentiment Analysis" , University of Paris-Sud, Lab LIMSI-CNRS, France, 2012, http://www.theses.fr/2012PA112101/abes/VD2_Pak_Alexander_13062012.pdf.

[9] A. Pak, and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining", Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), 2010, pp. 1320–1326.

[10] H. Saif, Y. He, and H. Alani, "Semantic sentiment analysis of twitter", ISWC'12 Proceedings of the 11th international conference on The Semantic Web - Volume Part I, 2012, pp. 508–524.

[11] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge, "Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph", Proceedings of the First Workshop on Unsupervised Learning in NLP EMNLP '11, 2011, pp. 53–63.

[12] M. Subhabrata, M. Akshat, A.R. Balamurali, and B. Pushpak, "TwiSent: A Multistage System for Analyzing Sentiment in Twitter", Proceedings of the 21st ACM international conference on Information and knowledge management CIKM '12, 2012, pp 2531–2534.

[13] Z. Elberrichi, M. Taibi, A.Belaggoun," Multilingual medical documents classification based on mesh domain ontology". International Journal of Computer Science Issues, (2012). 9(2), 150.