Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets[*]

Pablo Gamallo
CITIUS
Univ. de Santiago de Compostela
pablo.gamallo@usc.es

Marcos Garcia
Cilenis Language Technology, S.L.
marcos.garcia@cilenis.com

Abstract

This article describes a strategy based on a naive-bayes classifier for detecting the polarity of English tweets. The experiments have shown that the best performance is achieved by using a binary classifier between just two sharp polarity categories: positive and negative. In addition, in order to detect tweets with and without polarity, the system makes use of a very basic rule that searchs for polarity words within the analysed tweets/texts. When the classifier is provided with a polarity lexicon and multiwords it achieves 63% F-score.

1 Introduction

Sentiment Analysis consists in finding the opinion (e.g. positive, negative, or neutral) from text documents such as movie reviews or product reviews. Opinions about movies, products, etc. can be found in web blogs, social networks, discussion forums, and so on. Companies can improve their products and services on the basis of the reviews and comments of their costumers. Recently, many works have stressed the microblogging service Twitter. As Twitter can be seen as a large source of short texts (tweets) containing user opinions, most of these works make sentiment analysis by identifying user attitudes and opinions toward a particular topic or product (Go et al., 2009). The task of making sentiment analysis from tweets is a hard challenge. On the one hand, as in any sentiment analysis framework, we have to deal with human subjectivity. Even humans often disagree on the categorization of the positive or negative sentiment that is supposed to be expressed on a given text (Villena-Román et al., 2013). On the other hand, tweets are too short text to be linguistically analyzed, and it makes the task of finding relevant information (e.g. opinions) much harder.

The SemEval-2014 task "Sentiment Analysis in Twitter" is an evaluation competition that includes a specific task directly related to sentiment analyisis. In particular, subtask B, called "Message Polarity Classification", consists in classifying whether a given message is of positive, negative, or neutral sentiment. For messages conveying both a positive and negative sentiment, the stronger sentiment should be chosen. The results of our system in this task are situated in the average out of 51 evaluated systems.

In this article, we describe the learning strategies we developed so as to perform this task, all of them based on bayesian classification.

2 Naive Bayes Classifier

Most of the algorithms for sentiment analysis are based on a classifier trained using a collection of annotated text data. Before training, data is preprocessed so as to extract the main features. Some classification methods have been proposed: Naive Bayes, Support Vector Machines, K-Nearest Neighbors, etc. However, and according to (Go et al., 2009), it is not clear which of these classification strategies is the more appropriate to perform sentiment analysis.

We decided to use a classification strategy based on Naive Bayes (NB) because it is a simple and intuitive method whose performance is similar to other approaches. NB combines efficiency (optimal time performance) with reasonable accuracy. The main theoretical drawback of NB methods is that it assumes conditional independence among the linguistic features. If the main features are the tokens extracted from texts, it is evident that they

[*] This work has been supported by the projects: HPC-PLN: Ref:EM13/041 (Program *Emergentes*, Xunta de Galicia), Celtic: Ref:2012-CE138 and Plastic: Ref:2013-CE298 (Program Feder-Innterconecta)

Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 171–175,
Dublin, Ireland, August 23-24, 2014.

cannot be considered as independent, since words co-occuring in a text are somehow linked by different types of syntactic and semantic dependencies. However, even if NB produces an oversimplified model, its classification decisions are surprinsingly accurate (Manning et al., 2008).

2.1 Strategy

Two different naive bayes classifiers have been built, according to two different strategies:

Baseline This is a naive bayes classifier that learns from the original training corpus how to classify the three categories found in the corpus: Positive, Negative, and Neutral. So, no modification has been introduced in the training corpus.

Binary The second classifier was trained on a simplified training corpus and makes use of a polarity lexicon. The corpus was simplified since only positive and negative tweets were considered. Neutral tweets were not taken into account. As a result, a basic binary (or boolean) classifier which only identifies both Positive and Negative tweets was trained. In order to detect tweets without polarity (or Neutral), the following basic rule is used: if the tweet contains at least one word that is also found in the polarity lexicon, then the tweet has some degree of polarity. Othewise, the tweet has no polarity at all and is classified as Neutral. The binary classifier is actually suited to specify the basic polarity between positive and negative, reaching a precision of more than 80% in a corpus with just these two categories.

3 Preprocessing

As we will describe in the next section, the main features of the model are lemmas extracted using lemmatization. Given that the language of microblogging requires a special treatment, we propose a pre-processing task to correct and normalize the tweets before lemmatizing them.

The main preprocessing tasks we considered are the following:

- removing urls, references to usernames, and hashtags
- reduction of replicated characters (e.g. *loooooveeee* → *love*)
- identifying emoticons and interjections and replacing them with polarity or sentiment expressions (e.g. :-) → *good*)

4 Features

The features considered by the classifier are lemmas, multiwords, polarity lexicons, and valence shifters.

4.1 Lemmas (UL)

To characterise the main features underlying the classifier, we make use of unigrams of lemmas instead of tokens to minimize the problems derived from the sparse distribution of words. Moreover, only lemmas belonging to lexical categories are selected as features, namely nouns, verbs, adjectives, and adverbs. So, grammatical words, such as determiners, conjunctions, and prepositions are removed from the model.

To configure the feature representation, the frequency of each selected lemma in a tweet is stored.

4.2 Multiwords (MW)

There is no agreement on which is the best option for sentiment analysis (unigrams, bigrams, ...). In (Pak and Paroubek, 2010), the best performance is achieved with bigrams, while (Go et al., 2009) show that the better results are reached with unigrams. An alternative option is to make use of a selected set of n-grams (or multiwords) identified by means of regular patterns of PoS tags. Multiword expressions identified by means of PoS tags patterns can be conceived as linguistically motivated terms, since most of them are pairs of words linked by syntactic dependencies.

So, in addition to unigrams of lemmas, we also consider multiwords extracted by an algorithm based on patterns of PoS tags. In particular, we used the following set of patterns:

- NOUN-ADJ
- NOUN-NOUN
- ADJ-NOUN
- NOUN-PRP-NOUN
- VERB-NOUN
- VERB-PRP-NOUN

The instances of bigrams and trigrams extracted with these patterns ared added to the unigrams to build the language model. Multiword extraction was performed using our tool *GaleXtra*[1], released under GPL license and described in (Mario Barcala and Eva Domínguez and Pablo Gamallo and Marisol López and Eduardo Moscoso and Guillermo Rojo and Paula Santalla and Susana Sotelo, 2007).

4.3 Polarity Lexicon (LEX)

We have built a polarity lexicon with both *Positive* and *Negative* entries from different sources:

- AFINN-111[2] contains $2,477$ word forms, which were lemmatized and converted into $1,520$, positive and negative lemmas.
- Hedonometer[3] contains about $10,000$ frequent words extracted from tweets which were classified as expressing some degree of hapiness (Dodds et al., 2011). We selected the 300 most positive lemmas from the initial list.
- Hu&Liu list (Liu et al., 2005) contains over $6,800$ words out of which 5 positive and negative lemmas were selected $5,695$.
- Sentiwordnet-3.0 (Baccianella et al., 2010) contains more than $100,000$ entries. We selected a subset of $6,600$ positive and negative lemmas with the highest polarity values.
- Finally, we have built a polarity lexicon with $10,850$ entries by merging the previous ones.

The final polarity lexicon is used in two different ways: on the one hand, it is used to identify neutral tweets, since a tweet is considered as being neutral if it does not contain any lemma appearing in the polarity lexicon. On the other hand, we have built artificial tweets as follows: each entry of the lexicon is converted into an artificial tweet with just one lemma inheriting the polarity (positive or negative) from the lexicon. The frequency of the word in each new tweet is the average frequency of lemmas in the training corpus. These artificial tweets will be taken into account for training the classifiers.

4.4 Valence Shifters (VS)

We take into account negative words that can shift the polarity of specific lemmas in a tweet. In the presented work, we will make use of only those valence shifters that reverse the sentiment of words, namely *negations*. The strategy to identify the scope of negations relies on the PoS tags of the negative word as well as of those words appearing to its right in the sequence. The algorithm is as follows:

Whenever a negative word is found, its PoS tag is considered and, according to its syntactic properties, we search for a polarity word (noun, verb, or adjective) within a window of 2 words after the negation. If a polarity word is found and is syntactically linked to the negative word, then its polarity is reversed. For instance, if the negation word is the adverb "not", the system only reverses the polarity of verbs or adjectives appearing to its right. Nouns are not syntactically linked to this adverb. By contrast, if the negation is the determiner "no" or "none", only the polarity of nouns can be reversed. Our strategy to deal with negation scope is not so basic as those described in (Yang, 2008) and (Anta et al., 2013), which are just based on a rigid window after the negation word: 1 and 3 words, respectively.

5 Experiments and Evaluation

5.1 Training corpus

In our preliminary experiments we have used the training dataset of tweets provided by SemEval-2014 organization (tweeti-b.dist.tsv). This set contains $6,408$ tweets, which were tagged with the following polarity values: Positive, Negative, Neutral, Objective, and Neutral-or-Objective. In order to fill the requirements of the task, we transformed Neutral, Objective, and Natural-or-Objective into a single tag: Neutral. In addition, we also used a selection of annotated tweets (namely $5,050$ positive and negative ones), which were compiled from an external source (Narr et al., 2012). Using the terminology provided by the organizers of SemEval-2014, we call "constrained" the systems trained with only the dataset provided by the organization and "unconstrained" the systems trained with both datasets.

5.2 Evaluated classifiers

We have implemented and evaluated several classifiers by making use of the two strategies de-

[1] http://gramatica.usc.es/\~gamallo/gale-extra/index.htm

[2] http://arxiv.org/abs/1103.2903

[3] http://www.hedonometer.org/

scribed in section 2, combined with the features defined in 4. We also distinguished those classifiers trained with only tweeti-b.dist.tsv (constrained systems) from those trained with both internal and external datasets (unconstrained). As a result, we implemented the following classifiers:

CONSTRAINED-BASELINE: This system was implemented on the basis of the "Baseline" strategy and the following two features: unigrams of lemmas (UL) and valence shifters (VS).

CONSTRAINED-BASELINE-LEX: This system was implemented on the basis of the "Baseline" strategy and the following three features: unigrams of lemmas (UL), polarity lexicon (LEX), and valence shifters (VS).

CONSTRAINED-BINARY-LEX: This system was implemented on the basis of the "Baseline" strategy and the following three features: unigrams of lemmas (UL), polarity lexicon (LEX), and valence shifters (VS).

CONSTRAINED-BINARY-LEX-MW: This system was implemented on the basis of the "Binary" strategy and the following features: unigrams of lemmas (UL), multiwords (MW), polarity lexicon (LEX), and valence shifters (VS).

UNCONSTRAINED-BINARY-LEX: This system was implemented on the basis of the "Binary" strategy and the following features: unigrams of lemmas (UL), polarity lexicon (LEX), and valence shifters (VS).

UNCONSTRAINED-BINARY-LEX-MW: This system was implemented on the basis of the "Binary" strategy and the following features: unigrams of lemmas (UL), multiwords (MW), polarity lexicon (LEX), and valence shifters (VS).

All the classifers have been implemented with Perl language. They rely on the naive-bayes algorithm and incorporate the preprocessing tasks defined in section 3.

5.3 Evaluation

To evaluate the classification performance of these classifiers, we used as test corpus another dataset provided by the organization: tweeti-b.devel.tsv. The results are shown in table 1, where the names of the evaluated systems are in the first column and F-Score in the second one.

System	F-score
CONSTR-BASE	.49
CONSTR-BASE-LEX	.56
CONSTR-BIN-LEX	.57
CONSTR-BIN-LEX-MW	**.61**
UNCONSTR-BIN-LEX	.58
UNCONSTR-BIN-LEX-MW	**.63**

Table 1: Results of our six systems

.

The results show that there is an improvement in performance when the classifiers are implemented with the Binary strategy, when they use a polarity lexicon, and when multiwords are considered as features. The two systems submmited to Semeval competition were those obtained the best scores: CONSTR-BIN-LEX-MW and UNCONSTR-BIN-LEX-MW. The scores obtained by these two systems in the competition are very similar to those obtained in the experiments depicted in Table 1. More precisely, in the Tweets2014 test corpus, the constrained system reached 0.62 F-score while the unconstrained version achieved 0.63. Our best system was ranked as 26th from 53 systems. A Spanish version of this system (Gamallo et al., 2013) also participated in the TASS-2013 competition (Villena-Román et al., 2013), where it was ranked as the 3th best system out of 13 participants.

6 Conclusions

We have presented a family of naive-bayes classifiers for detecting the polarity of English tweets. The experiments have shown that the best performance is achieved by using a binary classifier trained to detect just two categories: positive and negative. In order to detect tweets with and without polarity we used a very basic strategy based on searching for polarity lemmas within the text/tweet. If the tweet does not contain at least one lemma also found in an external polarity lexicon, then the tweet has not any polarity and, thereby, is tagged with the Neutral value. The use of both a polarity lexicon and multiwords also improves the results in a significant way. Our system is being used by Cilenis S.L, a company specialised in natural language technology, and being applied to four languages: English, Spanish, Portuguese, and Galician.

References

Antonio Fernández Anta, Luis Núñez Chiroque, Philippe Morere, and Agustín Santos. 2013. Sentiment Analysis and Topic Detection of Spanish Tweets: A Comparative Study of NLP Techniques. *Procesamiento del Lenguaje Natural*, 50:45–52.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Human Language Technology Conference - North American chapter of the Association for Computational Linguistics*, pages 2200–2204.

Peter Sheridan Dodds, Kameron Decker Harris, Isabel M. Kloumann, Catherine A. Bliss, and Christopher M. Danforth. 2011. Temporal patterns of happiness and information in a global social network: Hedonometrics and Twitter. *PLoS ONE*, 6(12):e26752.

Pablo Gamallo, Marcos Garcia, and Santiago Fernández-Lanza. 2013. TASS: A Naive-Bayes strategy for sentiment analysis on Spanish tweets. In *Workshop on Sentiment Analysis at SEPLN (TASS2013)*, pages 126–132, Madrid, Spain.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. In *CS224N Technical report*. Standford.

Bing Liu, Minqing Hu, and Junsheng Cheng. 2005. Opinion observer: Analyzing and comparing opinions on the web. In *14th International World Wide Web conference (WWW-2005)*, pages 342–351, New York, NY, USA.

Chris Manning, Prabhakar Raghadvan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, MA, USA.

Mario Barcala and Eva Domínguez and Pablo Gamallo and Marisol López and Eduardo Moscoso and Guillermo Rojo and Paula Santalla and Susana Sotelo. 2007. A Corpus and Lexical Resources for Multi-word Terminology Extraction in the Field of Economy. In *3rd Language & Technology Conference (LeTC'2007)*, pages 355–359, Poznan, Poland.

Sascha Narr, Michael Hulfenhaus, and Sahin Albayrak. 2012. Language-Independent Twitter Sentiment Analysis. In *Knowledge Discovery and Machine Learning (KDML), LWA*, pages 12–14, Dortmund, Germany.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In *LREC-2010*, Valletta, Malta.

Julio Villena-Román, Sara Lana, Eugeinio Martínez-Cámara, and Juan Carlos González-Cristóbal. 2013. TASS - Workshop on Sentiment Analysis at SEPLN. *Procesamiento del Lenguaje Natural*, 50:37–44.

Kiduk Yang. 2008. WIDIT in TREC 2008 blog track: Leveraging Multiple Sources of Opinion Evidence. In *The Seventeenth Text Retrieval Conference (TREC-2008)*, Gaithersburg, Maryland, USA.