UMCC_DLSI: Sentiment Analysis in Twitter using Polirity Lexicons and Tweet Similarity

Pedro Aniel Sánchez-Mirabal,
Yarelis Ruano Torres,
Suilen Hernández Alvarado
University of Matanzas / Cuba
pedroasm@umcc.cu
yara@umcc.cu
suilen.alvarado@umcc.cu

Yoan Gutiérrez,
Andrés Montoyo,
Rafael Muñoz
University of Alicante/Spain
ygutierrez@dlsi.ua.es
montoyo@dlsi.ua.es
rafael@dlsi.ua.es

Abstract

This paper describes a system submitted to *SemEval-2014 Task 4B: Sentiment Analysis in Twitter*, by the team **UMCC_DLSI_Sem** integrated by researchers of the University of Matanzas, Cuba and the University of Alicante, Spain. The system adopts a cascade classification process that uses two classifiers, K-NN using the lexical Levenshtein metric and a Dagging model trained over attributes extracted from annotated corpora and sentiment lexicons. Phrases that fit the distance thresholds were automatically classified by the KNN model, the others, were evaluated with the Dagging model. This system achieved over 52.4% of correctly classified instances in the Twitter message-level subtask.

1 Introduction

Nowadays, one of the most important sources of data to extract useful and heterogeneous knowledge is Textual Information. Daily, millions of Tweets, SMS and blog comments increase the huge volume of information available for researchers. Texts can provide factual information, such as: descriptions, lists of characteristics, or even instructions to opinion-based information, which would include reviews, emotions, or feelings (Gutiérrez et al., 2013). These facts have motivated that dealing with the identification and extraction of opinions and sentiments in texts requires special attention. Applications of Sentiment Analysis are now more common than ever in fields like politics and business. More than 50 systems participating in this task, clearly indicate the increase of interest in the scientific community.

Twitter messages can be found among of the most used corpora nowadays for Sentiment Analysis (SA). This kind of messages involves an evident informality which has been addressed in different ways. For example, there are some works like (Gutiérrez et al., 2013) that apply normalisation textual tools to reduce the informality of the twitter messages. Authors such as (Go et al., 2009), (Gutiérrez et al., 2013), (Fernández et al., 2013) and others are focused on the application of preprocessing processes and feature reduction to be able to standardise twitter messages and reduce different types of elements like hashtags, user nicks, urls, etc.

In terms of those techniques that can be used for SA, we can cite (Pang et al., 2002) who built a lexicon with associated polarity value, starting with a set of classified seed adjectives and using conjunctions (and) disjunctions (or, but) to deduce the orientation of new words in a corpus. This research was based on machine learning techniques to address Sentiment Classification. Other interesting research is (Turney, 2002), which classifies words according to their polarity based on the idea that terms with similar orientation tend to co-occur in documents. There are a large quantity of approaches to deal with SA, and basically most of them are based on word bags and/or annotated corpora as knowledge base. Based on this information the SA systems are able to apply different types of evaluation techniques such as machine learning or statistic formulas to predict the correct classification. As part of machine learning approaches we would like to mention those works such as (Go et al., 2009), (Mohammad et al., 2013) and others that were based on feature vectors and which cover a wide range settings of SA. As a starting point, we based this work on the (Mohammad et al., 2013) approach, adding

Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 727–731,
Dublin, Ireland, August 23-24, 2014.

new features extracted from the sentiment repositories Sentiment 140 [1] and NRC-Hashtag Sentiment (Mohammad and Turney, 2013).

The remainder of this paper is structured as follows: section 2 describes in detail the approach presented. In section 3 we explain the experiments we carried out. Finally in section 4 conclusions and future works are expounded.

2 System Description

In this section we present our system in detail which is able to classify the polarity of tweets as positive, negative, or neutral.

The system is structured in two main stages. The first stage consists of classifying a given tweet. For that, we first recovered all the tweets from the training corpus that have a similarity value greater than a fixed threshold T. The second stage consists of classifying using the K-NN rule (Coomans and Massart, 1982), considering as K all tweets recovered. The process begins with $T = 0.9$ decreasing it until $T = 0.6$. In section 3 we will explain how these values were determined.

As similarity metric we use the Levenshtein (Levenshtein, 1966) lexical distance. In case that we cannot find any tweet fulfilling the condition, the tweet polarity is assigned using a second classifier trained using **Dagging** which combines several **Logistic** classifiers set by **WEKA** as default.

2.1 Preprocessing

The first step in our system is to pre-process all tweets. The following operations were applied in the given order.

- Replacing emoticons: Each emoticon is replaced by a word according to a lexicon of emoticons. The meanings of the emoticons were taken from `http://en.wikipedia.org/wiki/List_of_emoticons`.

- Replacing acronyms: Each acronym is replaced by its meaning. The meanings of the acronyms were taken from `http://www.acronymfinder.com/`.

- Cleaning text: Remove not alphanumeric characters from the tweet.

- Replacing abbreviations: Each abbreviation is replaced by its respective words. The abbreviations were taken from `http://en.wikipedia.org/wiki/Abbreviation`.

- Lemmatising: Each word is replaced by its lemma. We use Freeling 3.0 (Padró and Stanilovsky, 2012) for this purpose. We only retain lemmas corresponding to adjectives, adverbs, interjections, nouns and verbs.

- Expanding contractions: Each contraction is replaced by its respective word. The contractions were taken from `http://www.softschools.com/language_arts/grammar/contractions/contractions_list/`.

- Deleting punctuation marks.

- Deleting stop words. The stop words were taken from `http://www.ranks.nl/stopwords`.

2.2 Recovering tweets from similarity

As it was explained before, in a first step we tried to classify tweets using the K-NN rule. To recover the K similar tweets we used the Levenshtein metric (Levenshtein, 1966). This measure allows to compute the similarity of two strings of symbols counting the minimum number of deletions, substitutions and insertions necessary to transform one string into another. In our case, each word in the string is considered as a symbol. In the future we plan to improve this metric using Levenshtein at word level and then at sentence level. This metric is known as DLED (Double Levenshteins Edit Distance) and will be taken from (Fernández et al., 2012).

2.3 Features for Dagging classifier

We represented each tweet as a vector of features based in (Mohammad et al., 2013) plus other new ones. Also we used the lexicons **Sentiment 140** and **NRC-Hashtag Sentiment** as it was defined by Mohammad.

Also two new lexicons, named **NRC Emotion Lexicon 1.0** and **NRC Emotion Lexicon 2.0** were derived from the **NRC Emotion Lexicon** (Mohammad and Turney, 2013). In the first case we associated to each word just the values in the columns *positive* and *negative* of **NRC Emotion Lexicon**, thus, no sentiment score was computed.

[1] http://cs.stanford.edu/people/alecmgo/trainingandtestdata.zip

For the second lexicon, the *positive* score was calculated as the sum of the values for the classifications *positive*, *anticipation*, *joy*, *surprise* and *trust*. On the other hand, the negative score was computed as the sum of the values for the classifications *negative*, *anger*, *disgust*, *fear*, *sadness* and *trust*.

In each case we computed the following attributes:

- Pos: Sum of the positive scores of each token in the tweet over the number of tokens in the tweet.
- Neg: Sum of the negative scores of each token in the tweet over the number of tokens in the tweet.
- PercentPos: $\frac{100*Pos}{Pos+Neg}$
- MissNGram: Percent of tokens in the tweet that were not found in the lexicon.

For the **Sentiment 140** and **NRC-Hashtag Sentiment** lexicons we also computed the feature:

- SSE: Sum of the sentiment score of each token in the tweet over the number of tokens in the tweet.

Based on the information involved into Sentiment 140 and NRC-Hashtag Sentiment lexicons, unigrams, bigrams and pairs were tokenised involving any non-contiguous combination of the previous n-grams. With respect to the pairs extraction were considered the following possibilities: unigram-unigram, unigram-bigram and bigram-bigram. Similar to (Mohammad et al., 2013) different set of attributes were generated for each type of token. As result an initial set of 50 attributes were obtained.

In the case of the new lexicons (NRC Emotion Lexicon 1.0 and NRC Emotion Lexicon 2.0), only unigrams were considered. Moreover, the feature **SSE** was not computed. So, another 8 features were taken into account with respect to these lexicons.

Finally we computed:

- NCL: Percent of tokens in capital letters.
- NoE: Number of emoticons in the tweet.
- NoA: Number of acronyms in the tweet.

In general the system works with a total of 61 attributes.

2.4 Classifier Design

As training set, we joined the preprocessed tweets from both the *train* and *development* sets provided by the Task9B of Semeval-2014. The Dagging classifier was trained using this set with the following parameters **-F 15 -S 1 -W weka.classifiers.functions.Logistic – -R 1.0E-8 -M -1** using a 10 fold cross-validation as evaluation method.

3 Experiments

The experiments were evaluated over the training dataset provided by Task 9: Sentiment Analysis in Twitter, subtask B. Based on the explanation provided in section 2 according to the initialisation of the threshold T to ensure that the K similar tweets are in fact similar enough, we carried out an experiment for different values of T. These experiments refer an analysis to know how the variation of T affects the classification results.

T	% CCI
0.9	86.7
0.8	83.3
0.7	74.1
0.6	67.2
0.5	61.1
0.4	55.0
0.3	56.0

Table 1: Results of the K-NN classifier using Levenshtein metric.

T	% CCI
0.9	81.2
0.8	83.3
0.7	74.1
0.6	66.7
0.5	63.1
0.4	60.6
0.3	54.2

Table 2: Results of the K-NN classifier using Matching Coefficient metric.

The first stage of the system was applied to compute the number of instances which have at least one instance with a similarity value greater than T. We computed the percent of instances correctly classified ($\%CCI$). Table 1 shows the behaviour of the system when T changes. Table 2 shows the results of the K-NN classifier using

System	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm
Best result	74.8	70.3	72.1	71.0	58.2
Average result	63.5	55.6	59.8	60.6	45.4
UMCC-DLSI-Sem	**53.1**	**50.0**	**52.0**	**55.4**	**42.8**
Worse result	29.3	24.6	34.2	33.0	29.0

Table 3: Results in the SemEval-2014 Task 4B.

Matching Coefficient metric (`http://www.coli.uni-saarland.de/courses/LT1/2011/slides/stringmetrics.pdf`). This metric counts the quantity of matched symbols (words in this case) between two sentences.

Furthermore, we repeated this experiment using the Matching Coefficient similarity metric to better tunning the algorithm and to evaluate if the results behave in a similar way when T changes. In both cases, we use the implementation provided in the **SimMetrics** library.

As those results shows, when T decrease the accuracy decrease too. In practice, for the values of T lower than 0.6 the results are worse than 61.4% using the Dagging classifier in the 10 fold cross-validation. For that reason, as was mentioned in 2, we only tried to apply the first stage for values of $T \geq 0.6$.

We evaluated our system in the challenge Task 4B: Sentiment Analysis in Twitter, using the provided training and test data of this challenge. Based on the classifier obtained in the training process we tested our system over the test dataset achieving values of $\%CCI$ up to 55.4. Table 3 show detailed results for each of the 5 different sources.

4 Conclusions and Future Works

Our system was based on an approach that follows two stages to classify the polarity of tweets. Regardless the fact that our system behaves worse than the average, we consider that the approach is suitable to deal with SA, since our results are close to the average. As future works we will study other approaches in order to encourage further developments of this proposal. Several issues could be adjusted, for example, other distances should be tested and evaluated such as DLED (Double Levenshteins Edit Distance) (Fernández et al., 2012). Also, features that encode information about the presence of negation and opposition words could be very useful.

Acknowledgements

This research work has been partially funded by the University of Alicante, Generalitat Valenciana, Spanish Government and the European Commission through the projects, "Tratamiento inteligente de la informacin para la ayuda a la toma de decisiones" (GRE12-44), ATTOS (TIN2012-38536-C03-03), LEGOLANG (TIN2012-31224), SAM (FP7-611312), FIRST (FP7-287607) and ACOMP/2013/067.

References

D. Coomans and D.L. Massart. 1982. Alternative k-nearest neighbour rules in supervised pattern recognition : Part 1. k-nearest neighbour classification by using alternative voting rules. *Analytica Chimica Acta*, 136(0):15–27.

Antonio Fernández, Yoan Gutiérrez, Héctor Dávila, Alexander Chávez, Andy González, Rainel Estrada, Yenier Castañeda, Sonia Vázquez, Andrés Montoyo, and Rafael Muñoz. 2012. Umcc_dlsi: Multidimensional lexical-semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 608–616, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Javi Fernández, Yoan Gutiérrez, José M Gómez, Patricio Martınez-Barco, Andrés Montoyo, and Rafael Munoz. 2013. Sentiment analysis of spanish tweets using a ranking algorithm and skipgrams. *Proc. of the TASS workshop at SEPLN 2013. IV Congreso Español de Informática*, pages 17–20.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *Processing*, pages 1–6.

Yoan Gutiérrez, Andy González, Roger Pérez, José I. Abreu, Antonio Fernández Orquín, Alejandro Mosquera, Andrés Montoyo, Rafael Muñoz, and Franc Camara. 2013. Umcc_dlsi-(sa): Using a ranking algorithm and informal features to solve sentiment analysis in twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International*

Workshop on Semantic Evaluation (SemEval 2013), pages 443–449, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Vladimir Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Cybernetics and Control Theory*, 10(8):707–710. Original in *Doklady Akademii Nauk SSSR* 163(4): 845–848 (1965).

Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a word-emotion association lexicon. 29(3):436–465.

Saif M. Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *CoRR*, abs/1308.6242.

Lluís Padró and Evgeny Stanilovsky. 2012. Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey, May. ELRA.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peter D. Turney. 2002. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA. Association for Computational Linguistics.