Polarity analysis of non figurative tweets : Tw-StAR participation on DEFT 2017

Hala Mulki[1] Hatem Haddad [2] Mourad Gridach[3]
(1) Department of Computer Engineering, Selcuk University, Konya, Turkey
(2) Department of Computer and Decision Engineering, Universite Libre de Bruxelles, Belgium
(3) High Institute of Technology, Ibn Zohr University, Agadir, Morocco
halamulki@selcuk.edu.tr, Hatem.Haddad@ulb.ac.be, m.gridach@uiz.ac.ma

RÉSUMÉ

Dans cet article, nous présentons notre contribution dans l'atelier "Analyse d'opinion et langage figuratif dans des tweets en français DEFT 2017". Nous avons abordé la tâche 1 intitulée «Classification des tweets non figuratifs selon leur polarité ». Nous proposons trois stratégies de classification des sentiments utilisant : un modèle d'apprentissage supervisé, un modèle basé sur la lexique et un modèle de type réseau de neurones. Pour le premier modèle, une nouvelle stratégie est introduite où les entités nommées (ENs) ont participé à la tâche d'analyse de sentiments. Pour les deux premiers modèles, les fonctionnalités de bag-of-N-grams ont été adoptées alors que pour le troisième modèle, les caractéristiques ont été extraites automatiquement à partir des Tweets sous la forme de vecteurs de documents. L'évaluation officielle des trois modèles montre que la meilleure performance est atteinte par le modèle d'apprentissage supervisé. Néanmoins, les résultats obtenus par le modèle de type réseau de neurones sont considérés comme prometteurs et peuvent être encore améliorés si des vecteurs d'apprentissage préalablement formés sont utilisés pour initialiser les caractéristiques du modèle.

ABSTRACT

In this paper, we present our contribution in DEFT 2017 international workshop. We have tackled task 1 entitled "Polarity analysis of non figurative tweets ". We propose three sentiment classification models implemented using lexicon-based, supervised, and document embedding-based methods. For the first model, a novel strategy is introduced where Named Entities (NEs) have been involved in the Sentiment Analysis task. The first two models adopted bag-of-N-grams features while for the third model, features have been extracted automatically from the data itself in the form of document vectors. The official evaluation of the three models indicated that the best performance was achieved by the supervised learning-based model. Nevertheless, the results obtained by the document embedding-based model are considered promising and can be further improved if pretrianed French word vectors are used to initialize the model's features.

MOTS-CLÉS : Analyse des sentiments, apprentissage supervisé, modèle basé sur le lexique , entités nommées.

KEYWORDS: Sentiment analysis, supervised learning, lexicon-based model, document embeddings, named entities.

1 Introduction

Social media is literally shaping decision making processes in many aspects of our daily lives. Exploring online opinions is therefore becoming the focus of many analytical studies. Twitter is one of the most popular micro-blogging systems that enables a real-time tracking of opinions towards ongoing events (Mohammad *et al.*, 2016). Hence, it provides the needed feedback information for analytical studies in several domains such as politics and targeted advertising. Sentiment Analysis (SA) plays an essential role in performing such studies as it can extract the sentiments out of the opinions and classify them into the polarities they represent (Tang *et al.*, 2015). Twitter is widely accessed by non-English speakers where more than 34% of tweets are posted in languages other than English [1]. Yet, few research has been devoted for non-English languages (Korayem *et al.*, 2016). The French language has been ranked among the top seven languages mostly used on Twitter [2]. Therefore, providing an SA model for French is considered a crucial step towards developing non-English oriented SA systems.

Here, we describe our participation in Task 1 of DEFT 2017 (Benamara *et al.*, 2017) under the team's name "Tw-StAR". The task requires classifying the sentiment of single French tweets into one of the classes : positive, negative, mixed or objective. We used three classification strategies : supervised learning based model, lexicon-based model equipped with an enriched lexicon expanded using Named Entities (NEs) extracted from the training dataset and document embeddings-based model. Various types of features have been extracted including N-grams and document vectors. The presented models have been trained and tuned then evaluated using the test data.

The remainder of the paper is organized as follows : In Section 2, we describe the preprocessing step. In Section 3, we identify the extracted feature sets. Section 4 explains presented models. Results are reviewed and discussed in Section 5 while Section 6 concludes the study and future work.

2 Data Preprocessing

In this step, we have first cleaned the tweets from the unsentimental content such as URLs, username, dates, hashtags, retweet symbols, punctuation and emotions to get the French text with stopwords kept as some of them may carry sentimental information (Saif *et al.*, 2016). Thus, a tweet such "Sarkozy : "Ce n'est pas Hollande que le PS voulait, c'est DSK !" http ://bit.ly/yqVXPm" becomes "Sarkozy Ce n'est pas Hollande que le PS voulait c'est DSK " after preprocessing. Lastly, for the lexicon-based model, we have subjected each tweet to tokenization in terms of obtaining its unigrams, bigrams and a combination of both to assist the lookup process in the lexicon.

3 Features Extraction

Bag-of-N-grams features have been adopted to be used in both supervised and lexicon-based models. N-grams represent a sequence of adjoining N items collected from a given corpus (Tripathy *et al.*, 2016). Extracting N-grams can be thought of as exploring a large piece of text through a window

1. http://semiocast.com
2. http://on.mash.to/IW558d

of a fixed size (Pagolu *et al.*, 2016). Features selection has been performed using NLTK module FreqDist which gives a list of the distinct words ordered by their frequency of appearance in the corpus (Bird, 2006). A specific number of features was defined (equals to 11,527 for the combination of unigrams+bigrams+trigrams) in order to be selected from the FreqDist's list. For a certain N-grams scheme, a tweet's feature vector is constructed via examining the presence/absence of the N-grams features among the tweet's tokens. Consequently, the feature vector's values are identified as True (presence) or False (absence). The feature extraction pipeline is illustrated in Figure 1.

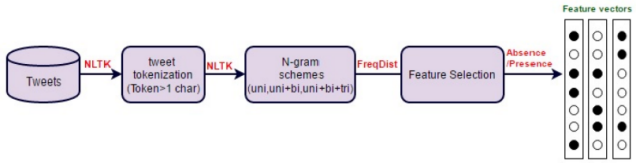

FIGURE 1 – Feature extraction pipeline for the supervised learning-based model

As for the deep-learning based model, no features extraction was performed since this model has the ability of automatically learning the continuous and real-valued text features from the data itself (Tang *et al.*, 2015).

4 The Proposed Models

4.1 Supervised Learning-based Model

Supervised learning requires a labeled corpus to train the classifier on the text polarity prediction (Biltawi *et al.*, 2016). In our case, a polarity labeled dataset of (3906) French tweets provided by DEFT 2017 has been used such that 3125 tweets were dedicated to train the model while 781 tweets were used to tune it. The learning process has been carried out by inferring that a combination of specific features of a tweet yields a specific class. We have used Naive Bayes (NB) from Scikit-Learn. Additionally, linear SVM from LIBSVM was employed for its robustness. Regarding the used features, as higher-order N-grams performed better compared to unigrams (Saif *et al.*, 2016), we have adopted N-grams schemes ranging from unigrams up to trigrams.

4.2 Lexicon-based Model

In this strategy, neither labeled data nor training step are required to train the classifier. The polarity of a word or a sentence is determined using a sentiment lexicon that can be either pre-built, manually-built or translated. The sentiment lexicon contains subjective words along with their polarities (positive or negative). For each polarity, a sentiment weight is assigned using one of these weighting algorithms :

— Straight Forward Sum (SFS) method : adopts the constant weight strategy to assign weights to the lexicon's entries, where negative words have the weight of -1 while positive ones have the weight of 1. The polarity of a given text is thus calculated by accumulating the weights of

negative and positive terms. Then, the total polarity is determined by the sign of the resulted value.

— Double Polarity (DP) method : assigns both a positive and a negative weight for each term in the lexicon. For example, if a positive term in the lexicon has a weight of 0.8, then its negative weight will be : -1+0.8 = -0.2. Similarly, a negative term of -0.6 weight has a 0.4 positive weight. Polarity is calculated by summing all the positive weights and all the negative weights in the input text. Consequently, the final polarity is determined according to the greater absolute value of the resulted sum.

As major events are often related to specific persons, locations or organizations so called Named Entities (NEs), we have supposed that NEs' polarity is an important information with which the sentiment lexicon can be enriched and hence the SA performance can be improved. To combine NEs in this model, they were first extracted from the training set using an NER system developed by (Gridach, 2016) then their polarity was determined using a scoring algorithm. Later, positive/negative NEs were added to the lexicon. Here, we have adopted French Expanded Emotion Lexicon (FEEL) lexicon which includes 14,127 entry accompanied with their positive/negative polarities. This lexicon was constructed by translating the English lexicon NRC-EmoLex then expanded by obtaining synonyms for each lexicon's entry (Abdaoui *et al.*, 2016). In addition, we have further expanded the lexicon into 14,950 entries by adding the sentimental NEs (see Table 1)

Lexicon	**Positive**	**Negative**
FEEL	5704	8423
Extracted NEs	145	678
Total	5849	9101

TABLE 1 – Polarity distribution in the FEEL lexicon and in the extracted NEs

To determine a tweets' positive/negative/objective polarity, we have used SFS method where the polarity calculation procedure involved looking for entries that match the tweet's unigrams or bigrams in the lexicon. To handle the "mixed" polarity, we have altered the SFS algorithm such that if a non-zero equal number of positive and negative words are encountered within a tweet, then the tweet is considered of a mixed polarity. Thus, for instance, a tweet such "meilleur patissier yesss super j en salive déjà" which means"yes, the best pies maker, they are already mouth watering", the polarity is calculated by summing the polarity values of its tokens "meilleur+patissier+super+salive= 1+0+1+0=+2 >0 ". Hence, the tweet is classified positive.

4.3 Document Embeddings-based Model

Bag-of-N-grams features have a very little sense about the words' semantics; they also suffer from the curse of high dimensionality and data sparsity. Document embeddings-based approaches have recently emerged to provide a high sentiment classification performance without any effort spent on handcrafted features (Mikolov *et al.*, 2013). This is achieved through using multi-layer nonlinear neural networks (NN) to learn text representations (embeddings) from the data itself in an unsupervised manner (Tang *et al.*, 2015). Each NN layer transforms the representation at one level into a representation at a higher and more abstract level such that a piece of text is mapped into a continuous and real-valued space (LeCun *et al.*, 2015).

The learned representations can be then used as features for the sentiment classification task (Le &

Mikolov, 2014). Text embeddings are divided into two types :

— Word embeddings : Each word in the corpus is mapped to a real-valued low-dimensional vector in the embedding space using one of the word mapping algorithms such as word2vec (Mikolov *et al.*, 2013) and GloVe (Pennington *et al.*, 2014).
— Document embeddings : Denote continuous representations for larger blocks of text such as sentences, paragraphs or whole documents constructed out from the the linear combination of the vectors of words contained in it using a document mapping algorithm such as doc2vec (Mikolov *et al.*, 2013; Abburi *et al.*, 2016).

In this study, we have used Doc2Vec model provided by Gensim (Rehurek & Sojka, 2011) to generate the document feature vectors. Word vectors were trained from scratch and combined linearly to formulate the document vectors features. Using these features, the model have been then trained via SGD and LR classification algorithms.

5 Results and Discussion

The provided dataset consists of three parts :TRAIN (3125 tweets) for training models, DEV (781 tweets) for tuning models, and TEST (976tweets) for the official evaluation. Data preprocessing involved using regular expressions recognition and substitution provided by the re Python module. For the supervised-based model, N-grams feature schemes (unigrams+bigrams+trigrams) have been generated via NLTK [3]. On the other hand, documents vectors were produced using doc2vec via Gensim [4] to form the features of the document embeddings-based model. Having the data preprocessed and the features extracted, we have trained the supervised-based model using SVM (from LIBSVM [5]) and NB [6]. The document embeddings-based model have been trained using Stochastic Gradient Descent(SGD) [7] and Logistic Regression LR [8]. Regarding the lexicon-based model, NEs have been first extracted by an NER system based on deep neural networks where it combines a Bidirectional Long Short-Term Memory (BLSTM) with a Convolutional Neural Networks (CRF) on the top of the BLSTM with word vectors initialized using pretrained word embeddings. Additionally, it employs a character-level representation to represent each word by its characters which addresses out-of-vocabulary (OOV) issues (Gridach, 2016). We have developed a scoring algorithm to assign the proper polarity to each NE. This algorithm compares NES against the annotated tweets then assigns an aggregated signed score to each NE according to how many times an NE was encountered in positive, negative tweets while NEs found in objective/mixed tweets are ignored. Consequently, only positive/negative NEs are added to the lexicon. Later, each tweet's tokens (unigrams, unigrams+bigrams) are looked up in the enriched FEEL lexicon and due to the sentiment score calculated via the modified SFS method, the tweet's final polarity is defined.

The three models have been evaluated by applying them on the tweets of the TEST dataset. Official evaluation referred to the outperformance of the supervised-based model with SVM algorithm over both document embeddings and lexicon-based models where it achieved a macro (average) F-score equals to 49.1% compared to 40.6% and 21.6% scored by document embeddings-based and lexicon-

3. http://www.nltk.org/
4. https://radimrehurek.com/gensim/models/doc2vec.html
5. https://www.csie.ntu.edu.tw/Ëœcjlin/libsvm/
6. http://scikit-learn.org/stable/modules/naive_bayes.html
7. http://scikit-learn.org/stable/modules/sgd.html
8. http://scikit-learn.org/stable/modules/LogisticRegression.html

based models respectively. Table 2 lists the detailed results achieved by the three models for the TEST dataset where AVG F-score, AVG R and AVG P denote the macro F-score, macro recall and macro precision measures respectively.

Model	Algorithm	AVG F-score	AVG R	AVG P
Supervised	SVM	**0.491**	**0.482**	**0.513**
Document embeddings	SGD	0.406	0.417	0.408
Lexicon-based	-	0.216	0.304	0.271

TABLE 2 – Evaluation results with TEST dataset

Considering the results in Table 2, the outperformance of the supervised-based model can be attributed to the utilization of SVM algorithm as this algorithm can efficiently handle feature vectors of high dimensions through its overfitting protection property (Patil *et al.*, 2014). For the second-ranked system, the low dimensionality of feature vectors used by the document embeddings-based achieved quite acceptable results. However, further improvement can be obtained if the used document vectors were replaced by pretrained ones trained on a large external corpora since they provide a better representation of the corpus content (Lau & Baldwin, 2016). Finally, the poor results obtained by the lexicon-based model may be related to the low coverage of the used lexicon (Liu, 2012).

6 Conclusion

We have investigated sentiment classification of French tweets via three classification models of various features and different learning strategies. The official evaluation revealed that the supervised learning-based model has the best performance as it outperformed the other two models in all evaluation measures. However, the worst performance was achieved by the lexicon-based model. This can be addressed by using a merged lexica that can provide a sufficient coverage of the corpus or by combining specific categories of NEs such as persons, locations etc. As for the document embeddings-based model, the yielded results were satisfying for document vectors trained from scratch. In this context, further development can be obtained in the future if pretrained word vectors were used to initialize the document vectors needed for the sentiment classification task.

Références

ABBURI H., AKKIREDDY E. S. A., GANGASHETTY S. V. & MAMIDI R. (2016). Multimodal sentiment analysis of telugu songs. In *Proceedings of the 4th Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2016)*, p. 48–52.

ABDAOUI A., AZÉ J., BRINGAY S. & PONCELET P. (2016). Feel : a french expanded emotion lexicon. *Language Resources and Evaluation*, p. 1–23.

BENAMARA F., GROUIN C., KAROUI J., MORICEAU V. & ROBBA I. (2017). Analyse d'opinion et langage figuratif dans des tweets : présentation et résultats du défi fouille de textes deft2017. In *Actes de l'atelier DEFT de la conférence TALN 2017*.

BILTAWI M., ETAIWI W., TEDMORI S., HUDAIB A. & AWAJAN A. (2016). Sentiment classification techniques for arabic language : A survey. In *Information and Communication Systems (ICICS), 2016 7th International Conference on*, p. 339–346 : IEEE.

BIRD S. (2006). Nltk : the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, p. 69–72 : Association for Computational Linguistics.

GRIDACH M. (2016). Character-aware neural networks for arabic named entity recognition for social media. *WSSANLP 2016*, p.23̃.

KORAYEM M., ALJADDA K. & CRANDALL D. (2016). Sentiment/subjectivity analysis survey for languages other than english. *Social Network Analysis and Mining*, **6**(1), 75.

LAU J. H. & BALDWIN T. (2016). An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv :1607.05368*.

LE Q. & MIKOLOV T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, p. 1188–1196.

LECUN Y., BENGIO Y. & HINTON G. (2015). Deep learning. *Nature*, **521**(7553), 436–444.

LIU B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, **5**(1), 1–167.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.

MOHAMMAD S. M., SALAMEH M. & KIRITCHENKO S. (2016). How translation alters sentiment. *J. Artif. Intell. Res.(JAIR)*, **55**, 95–130.

PAGOLU V. S., CHALLA K. N. R., PANDA G. & MAJHI B. (2016). Sentiment analysis of twitter data for predicting stock market movements. *arXiv preprint arXiv :1610.09225*.

PATIL G., GALANDE V., KEKAN V. & DANGE K. (2014). Sentiment analysis using support vector machine. *International Journal of Innovative Research in Computer and Communication Engineering*, **2**(1), 2607–2612.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). Glove : Global vectors for word representation. In *EMNLP*, volume 14, p. 1532–1543.

REHUREK R. & SOJKA P. (2011). Gensim–python framework for vector space mo delling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*.

SAIF H., HE Y., FERNANDEZ M. & ALANI H. (2016). Contextual semantics for sentiment analysis of twitter. *Information Processing & Management*, **52**(1), 5–19.

TANG D., QIN B. & LIU T. (2015). Deep learning for sentiment analysis : successful approaches and future challenges. *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery*, **5**(6), 292–303.

TRIPATHY A., AGRAWAL A. & RATH S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, **57**, 117–126.