Manchester Metropolitan University

Pota, M and Esposito, M and Palomino, MA and Masala, GL (2018) A subword-based deep learning approach for sentiment analysis of political tweets. In: 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), 16 May 2018 - 18 May 2018, Krakow, Poland.

A Subword-based Deep Learning Approach for Sentiment Analysis of Political Tweets

Marco Pota
Institute for High Performance Computing and Networking
National Research Council of Italy
Naples, Italy
marco.pota@icar.cnr.it

Massimo Esposito
Institute for High Performance Computing and Networking
National Research Council of Italy
Naples, Italy
massimo.esposito@icar.cnr.it

Marco A. Palomino
Big Data Group
Univeristy of Plymouth
Plymouth, UK
marco.palomino@plymouth.ac.uk

Giovanni L. Masala
Big Data Group
Univeristy of Plymouth
Plymouth, UK
giovanni.masala@plymouth.ac.uk

Abstract—The successful use of online material in political campaigns over the past two decades has motivated the inclusion of social media platforms—such as Twitter—as an integral part of the political apparatus. Political analysts are increasingly turning to Twitter as an indicator of public opinion. We are interested in learning how positive and negative opinions propagate through Twitter and how important events influence public opinion. In this paper, we present a neural network-based approach to analyse the sentiment expressed on political tweets. First, our approach represents the text by dense vectors comprising subword information to better detect word similarities by exploiting both morphology and semantics. Then, a Convolutional Neural Network is trained to learn how to classify tweets depending on sentiment, based on an available labelled dataset. Finally, the model is applied to perform the sentiment analysis of a collection of tweets retrieved during the days prior to the latest UK General Election. Results are promising and show that the neural network approach represents an improvement over lexicon-based approaches for positive/negative sentence classification.

Keywords—sentiment analysis, social networks, Twitter, neural networks, deep learning, lexicon-based approach

I. INTRODUCTION

Due to its ease of use and ample reach, social media is rapidly setting trends and agendas in topics that range from technology and the entertainment industry to public health and politics. People are increasingly searching for opinions expressed on social media to make decisions based on the information they find [1]. Among the different social media platforms, we are particularly interested in *Twitter*, the online news and social networking service where users post and interact with messages known as *tweets* [2].

Political analysts are increasingly turning to Twitter to look for expressions of political opinion [2]. The successful use of online resources in political campaigns over the past two decades has motived the inclusion of Twitter as an integral part of the political apparatus [3-4]. Indeed, after the growth of candidate websites in 1996, Jesse Ventura's e-mail campaign in 1998, the online fund-raising for John McCain's campaign in 2000, the use of blogs for Howard Dean's campaign in 2004, and the pivotal role of social media in the Obama 'Yes We Can' campaign in 2008 [5], Twitter has established itself as a key communication channel in the political arena.

A fair amount of recent research on political posts retrieved from Twitter has focused on predicting future outcomes [6]—i.e., making quantitative predictions intended to outperform traditional methods of gathering information, such as surveys and opinion polls—or on proving that the predictive power of Twitter has been exaggerated [7]. However, little work has been done in terms of measuring the impact of political comments posted on Twitter. Our goal is, precisely, to study how positive and negative opinions propagate on social media and how important events influence public opinion. To this extent, we focus on analysing the sentiment of political tweets, using text classifiers to distinguish positively oriented tweets from negative ones. Our work lies within the area of *sentiment analysis*—the process of determining the emotional tone behind words, which is used to gain an understanding of attitudes, opinions and emotions expressed in online posts [8]. Our concern is to produce a suitable classifier to ascertain the *polarity* of tweets—i.e., whether the tweet is *positive*, *negative* or *neutral*.

Recent research suggests that neural network approaches have the potential to perform better than lexicon-based approaches to interpret the meaning of text posted online. Thus, we have developed a neural network to determine the sentiment expressed on tweets. Our approach comprises a pre-trained text representation of subword-based embeddings that are able to detect word similarities exploiting both morphology and semantics. We have also produced a classification model based on a deep neural architecture, which is pre-trained on an available labelled dataset to learn how to classify tweets depending on sentiment.

We have evaluated this approach on a collection of political tweets, by comparing it with a lexicon-based classification approach, here considered as baseline, that employs a list of English positive and negative *opinion words*, compiled originally by Hu and Liu in 2004 [9]—the list is updated regularly.

We do not attempt to predict the outcome of political events. We are simply interested in learning how we can use sentiment analysis to improve our understanding of political audiences. We have gathered a sample of tweets related to the 2017 UK General Election, and we have established that

a strong correlation exists between the neutral tweets that we collected, and the relative volume of searches made on Google over time, according to *Google Trends* [10]. Such correlation validates the use of our sentiment analysis approach to timely monitor political campaigns.

The remainder of this paper is organized as follows: Section II describes the deep learning method that we employed to determine the polarity of tweets, Section III introduces the dataset and baseline method for our work. Section IV presents our analysis of results. Finally, Section V states our conclusions and outlines opportunities for future work.

II. DEEP LEARNING APPROACH

The model proposed here for tweet classification is made of a pre-processing stage, followed by a feed-forward Convolutional Neural Network (CNN), comprising input layer, embedding layer, convolutional layer, pooling layer, fully connected layer and output layer. The architecture and the techniques used for training its parameters are described below. Architecture arrangements and hyperparameters are settled by taking into account previous work, where optimal settings are found for sentence classification and question classification [11-13]. Moreover, particular settings are chosen, matching the peculiarities of the tweet classification problem.

Tweet pre-processing consists in cleaning each textual fragment, by removing non-alphanumeric characters, since "#" and "@" are very frequent in tweets, but we hypothesized that those characters and punctuation are not influent for tweet classification, and substituting apostrophes and spaces, to obtain a sequence of single-space separated tokens. Then, each tweet is tokenized into L tokens, without lemmatization. A vocabulary is built of size V, comprising all the training and test words.

The input layer of the CNN represents each tweet as a matrix $\mathbf{X}$, composed by sparse vectors $\mathbf{x}_j$ with dimension V representing the words. The maximum tweet length $Lmax$ is the other dimension of $\mathbf{X}$ (last rows of shorter tweets are filled with zeros).

The input layer is transformed into the embedding layer, by representing each token as a dense vector of lower dimensions De. This is performed by multiplying $\mathbf{x}_j$ for the embedding matrix $\mathbf{W}_{emb}$, to obtain the dense representation $\mathbf{X}_{emb}$, made of $Lmax$ rows and De columns, as shown in the following:

$$\mathbf{X}_{emb} = \mathbf{X}\mathbf{W}_{emb} \tag{1}$$

In this work, the $\mathbf{W}_{emb}$ matrix is initialized by using pre-trained word vectors (tokens not present in the pre-trained vocabulary are initialized as null). The pre-trained embeddings are based on FastText model [14], with 300 embedding dimensions for each token, trained by the skip-gram approach [15] on the English Wikipedia corpus. The

skip-gram approach allows learning the word representation by predicting other words present in the same context. Moreover, the FastText extension of this approach takes into account also subword information, thus helping a better representation of words with respect to their morphology. For instance, opposite words, such as *believable* and *unbelievable* as well as *useful* and *useless,* are better distinguished based on their character n-grams.

The embedding layer is transformed into the convolutional layer by applying on $\mathbf{X}_{emb}$ a number Nf of 300 filters, which are matrices $\mathbf{W}_{conv}^{i}$ with Fs rows and De columns, where Fs is the filter size. Padding is avoided, and stride 1 is adopted. Here, filters of size 1, 2 and 3 are used in equal number. Therefore, for each tweet and for each filter, a vector $\mathbf{x}_{conv}^{i}$ of size $Lmax - Fs + 1$ is obtained:

$$\mathbf{x}_{conv}^{i} = \mathbf{X}_{emb} \otimes \mathbf{W}_{conv}^{i} \tag{2}$$

Filter sizes are chosen by taking into account the nature of tweets to be classified. In fact, sometimes isolated tokens are inserted, which drives to the choice of size 1 filters. Moreover, filters of size 2 and 3 are chosen to analyse the sequence of tokens, in order to consider sequences like "not good" or "not very well" as a whole fragment, and classify the tweet accordingly. Longer filter sizes are not considered, due to the shortness of average tweet length.

Then, for each filter i, a randomly initialized bias term b_{conv}^{i} is added to each j-th component $x_{conv}^{i,j}$, and a Rectified Linear Unit (*ReLU*) activation function is applied, to obtain the tweet representation in the convolutional layer:

$$x_{act}^{i,j} = ReLU\left(x_{conv}^{i,j} + b_{conv}^{i}\right) \tag{3}$$

In the pooling layer, the maximum element is taken among the elements of each filter from the previous layer:

$$p^{i} = \max\left\{x_{act}^{i,j}, j = 1,...,L\max - Fs + 1\right\} \tag{4}$$

All these values are concatenated into a vector $\mathbf{p}$ of size Nf.

In the following layer, a number of neurons equal to the number of classes (3 in this case) are fully connected to all nodes of the previous layer, i.e., the class activation values $\mathbf{y}$ are obtained by multiplying $\mathbf{p}$ by a weight matrix $\mathbf{W}_{fc}$, and adding a bias vector $\mathbf{b}_{fc}$ (both $\mathbf{W}_{fc}$, and $\mathbf{b}_{fc}$ are initialized with random values):

$$\mathbf{y} = \mathbf{p}\mathbf{W}_{fc} + \mathbf{b}_{fc} \tag{5}$$

A dropout probability of $d = 0.5$ is fixed to keep only some connections during the training phase, while $d = 1$ during predictions.

In the output layer, also made of 3 nodes, the maximum value of the previous layer is chosen, corresponding to the predicted class.

In order to train the neural model, a dataset is used, made of tweets already classified according to 3 classes: *Positive*, *Negative*, and *Neutral*. It was taken from SemEval-2017 Workshop task, and comprises 20633 tweets for training and 12284 for testing [16].

The learning procedure involves iterative adaptation of the network weights and the fine-tuning of word embeddings with respect to the specific classification task. In particular, this fine-tuning allows better distinguishing word representations of syntactically equivalent words (e.g. *good* and *bad*) but with opposite meanings with respect to their polarity. A backpropagation approach is used, that is based on Stochastic Gradient Descent algorithm, with batch size 100, Adadelta updating rule, learning rate 0.1, and 2000 epochs to ensure convergence. We obtained classification accuracy of 71.94% on the training set, and 63.59% on the test set.

The trained neural classification model is saved, to distinguish different sentiments on the dataset of political tweets.

III. DATASET AND BASELINE METHOD

In this paper, we consider the dataset related to the UK General Election used in [17] to make a comparison of the proposed method based on CNN with a baseline method [17]. A Java-based application interacting with the Twitter API [18] retrieved the public tweets between the 18th April 2017 to the 14th June 2017. Table 1 shows the dataset composition of the tweets collected with the main hashtag used.

TABLE I.   DATASET COMPOSITION

Number of records	40,997
Hashtags / words used	#UKGeneralElection #ge17 #GeneralElection2017 #Election2017 #ge2017

We consider as baseline approach the method proposed in Plummer et al [17] where authors used the list of opinion words (positive and negative) compiled by Hu and Liu originally in 2004 [9]. Such list is updated regularly for research purposes. At present, the list comprises 6,800 words, approximately. To determine the overall sentiment of a tweet, the baseline method loops through all the words in the tweet text and compare them to the sentiment words in the list. E.g. if a sentiment word in the text of the tweet is in the positive list, the overall sentiment value of the tweet increments a counter. On the base of the counters score for each tweet it is possible to define a sentiment positive, negative, neutral (and also balanced if there is a parity on the score positive vs negative words).

IV. ANALYSIS

The first measure that we performed is related to the similarity of the results of the sentiment assigned on the specific tweet by the proposed method with respect to the outcome of the baseline method [17] on the dataset of the Table 1. The value obtained for the similarity (number of match between the two methods) is 61.01% on this dataset. The neutrals in the proposed method correspond to neutrals plus balanced sentiments used in the baseline. In addition, in Table 2 the details related to the similarity for each single sentiment are shown.

TABLE II.   COMPARISON OF THE METHODS

Sentiment	Similarity of the CNN among the baseline method
Positive	29%
Negative	30%
Neutral	93%
Mean of value	61%

According to Connelly et al. [19] different approaches produce different scores in the positive and negative computations but the information that the methods produce are in our case very important. In particular we have a high correspondence between the CNN and the baseline on the neutral sentiments, as represented also in the trends of Figure 1.

This is an important outcome considering that, according with the authors in [17], a peak of neutral sentiment is correlated also to a peak of interest over time in the search using Google Trend [10]. Vice versa it is possible to consider Google Trend as a way to find interesting peaks of interest for specific topics, triggering a possible automatic system able to analyze the tweets sentiment for focused important date, instead to have a continuative analysis of all tweets.
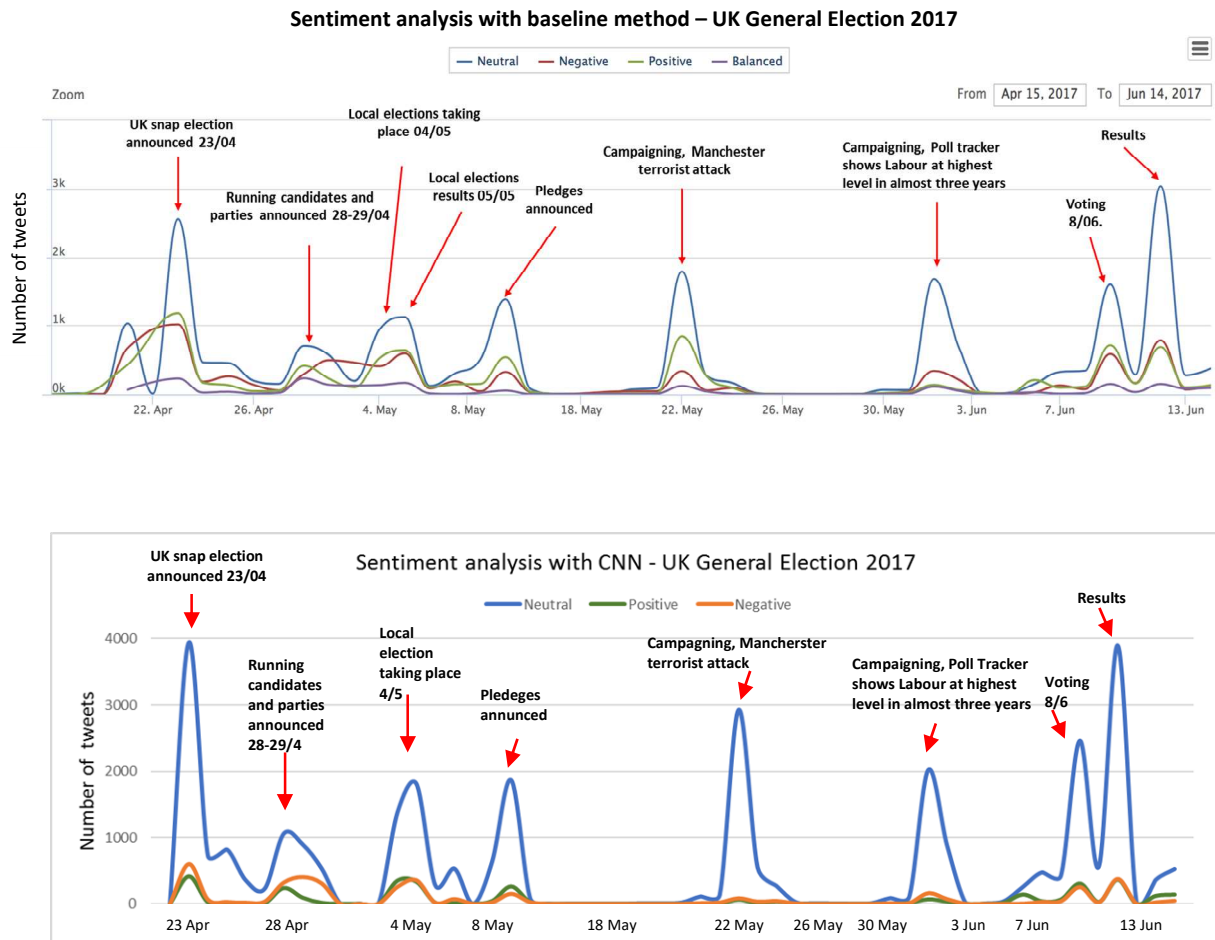
Fig. 1. Twitter Sentiment Analysis using a baseline method on the top and a CNN approach on the bottom

We also used word clouds [20] to visualize the words included in the tweets that we collected and explain their relevance to the events considered. The idea was filtering the tweets down to the words that appear with highest frequency. Prior to the creation of the word clouds, we removed the stop words from the tweets—i.e., we removed words that are extremely common and semantically non-selective, such as "the", "is", "at" and "on". We employed the stop-word list built by Salton and Buckley for the experimental SMART information retrieval system [21], which contains 571 words. By removing these words, we guarantee that the word clouds are not cluttered with non-relevant text. The actual rendering of the word clouds was done with the help of WordClouds.com [22], a freely-available online tool.

We looked into the two main events of the UK General Election campaigning period considering the 22nd of May 2017, when the campaigning goes ahead with different topics and there is the Manchester terrorist attack, and the 2nd of June 2017, when the poll tracker shows Labour at highest level in almost three years. Figures 2 and 3 show the word clouds for the positive, negative and neutral sentiments for the proposed method based on CNN.
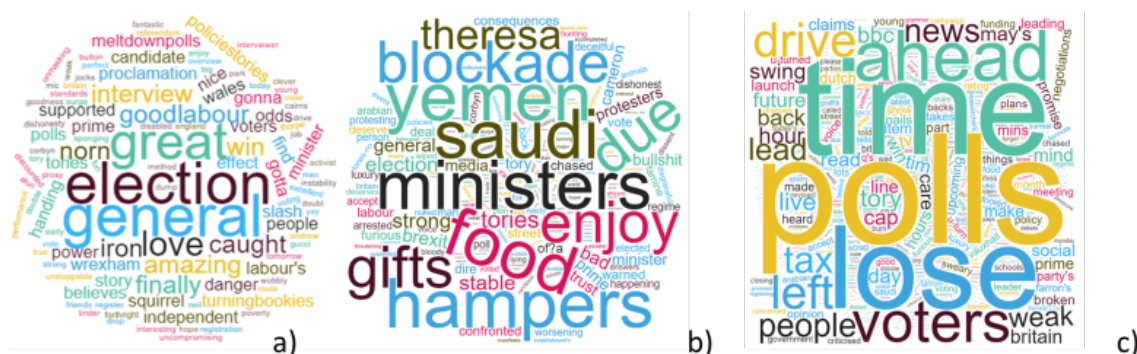
Fig. 2. 22nd May 2017, word clouds based on the sentiment extracted from the tweets with the proposed method: a) positive, b) negative, c) neutral, when the campaigning go ahead with different topics and there is the Manchester terrorist attack.
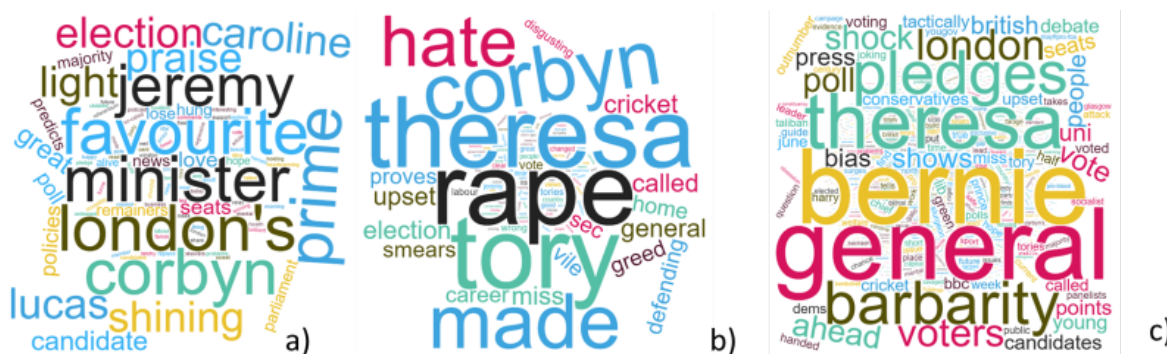

Fig. 3. 2nd June 2017 word clouds based on the sentiment extracted from the tweets with the proposed method: a) positive, b) negative, c) neutral, when the poll tracker shows Labour at highest level in almost three years.

Figure 2: *The 22nd May 2017 the campaigning goes ahead with different topics and there is the Manchester terrorist attack:*

Considering the positive sentiment of a), it is possible to note that there is a general hope in the outcomes of the election represented by words like *great*, *amazing*, *win*, *good labour* but the general phrases considered positive from the CNN, trained in the meaning of the whole phrase, don't contain strong positive words or references to the campaigning themes. Instead, the negative sentiment expressed in b) clearly catch the keywords related to the terrorist attack like *yemen*, *saudi*, *blockade* but also a general negative sentiment related to the *ministers*, the Premier *Theresa* May and the conservatory (*Tories*), in accordance with the results of the baseline method on the negative sentiment [17]. In c) are represented instead many words extracted by neutral comments on the *news*, *bbc*, the last *polls* available, repeating generic comments on *time*, go *ahead*, or by balanced comments on the campaigning themes on *lose voters*, *tax*, *tory* that the CNN consider as neutrals on the base of our trained model.

Figure 3*: The 2nd June 2017 poll tracker shows Labour at highest level in almost three years:*

In the general positive sentiment in a), we can note several links to the Labour party like *Corbyn*, *Jeremy*, and a general happiness related to the poll trackers prediction like *great*, *poll*, *praise*, *favourite*. In the word clouds of negative sentiment *Theresa* May and *tory* are the most important words, it is possible to see also some concerns about the partial outcomes of the *Corbyn* campaigning. In c) several news from US and Middle East are influencing the tweets, in particular, considering the Iran sanctions, with comments, shock, and barbarity words. In particular these tweets are influenced by the news published in the Intercept report of Zaid Jilani, and the important related comments from the American politician *Bernie* Sanders. In general, the CNN agrees with the baseline method for the general topics that dominate the positive and negative sentiments.

V. Conclusion

The sentiment analysis on Twitter is very important because produces interesting explanations of the orientations of voters' thoughts during the electoral campaigning. An interesting outcome is that the topics used by the Prime Minister Theresa May in support of the conservatory parties produced general negative sentiments with respect to themes expressed in the Jeremey Corbyn's manifesto of the Labour, which instead encountered positive sentiments. This is a possible explanation of the relative decrease of the consensus of the conservatories with respect to the increase of the Labour consensus, which was monitored during the poll tracker during the campaigning until the final vote.

The neutral tweets are an interesting index of the popularity of a topic, allowing to refine the search on contents related to specific days of interest. The approach of the CNN produces important outcomes and it is a viable solution to analyse the sentiment of the tweets. However, considering the comparison with a baseline method, which is based on a dictionary of positive/negative words, we note a small number of similarities for positive and negative classes. It is possible thinking to a possible refining the CNN results on the base of specific dictionary of words combining the approaches.

References

[1] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," Foundations and Trends in Information Retrieval, vol. 2(1-2), pp. 1–135, 2008.

[2] Twitter. Twitter. "It's what's happening," 2017 [cited 2017; Available from: https://twitter.com/].

[3] B. J. Jansen, et al., "Twitter power: Tweets as electronic word of mouth," Journal of the Association for Information Science and Technology, vol. 60(11), pp. 2169–2188, 2009.

[4] V. Gueorguieva, "Voters, MySpace, and YouTube," Social Science Computer Review, vol. 26(3), pp. 288–300.

[5] K. Wallsten, ""Yes we can": How online viewership, blog discussion, campaign statements, and mainstream media coverage produced a viral video phenomenon," Journal of Information Technology & Politics, vol. 7(2-3), pp. 163–81, 2010.

[6] S. Asur and B.A. Huberman, "Predicting the future with social media," in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT) 2010. Toronto, Canada: IEEE, 2010.

[7] P. T. Metaxas, E. Mustafaraj, and D. Gayo-Avello, "How (not) to predict elections," in 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), IEEE, 2011.

[8] B. Liu, "Sentiment analysis and opinion mining," Synthesis lectures on human language technologies, vol. 5(1), pp. 1–167, 2012.

[9] M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA: ACM, 2004.

[10] Google Inc. "Google Trends," 2017 [cited 2017; Available from: https://trends.google.co.uk/].

[11] M. Pota, M. Esposito, and G. De Pietro, "Convolutional Neural Networks for Question Classification in Italian Language," in Proceedings of The 16th International Conference on Intelligent Software Methodologies, Tools, and Techniques (SOMET_17), Kitakyushu, Japan, pp. 604–615, 2017. https://doi.org/10.3233/978-1-61499-800-6-604

[12] Y. Zhang, and B. C. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," in Proceedings of the 8th International Joint Conference on Natural Language Processing, Taipei, Taiwan, pp. 253–263, 2017.

[13] Y. Kim, "Convolutional neural networks for sentence classification," arXiv preprint, 2014, arXiv:1408.5882.

[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," Transactions of the Association for Computational Linguistics, vol. 5, pp. 135–146, 2017.

[15] T. Mikolov, K. Chen, G. Corrado, J. Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv preprint, 2013, arXiv:1301.3781

[16] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 task 4: Sentiment analysis in Twitter," in Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 502–518, 2017.

[17] M. Plummer, M. Palomino, and G. Masala, "Analysing the Sentiment Expressed by Political Audiences on Twitter: The case of the 2017 UK general election," in Proceedings of the 2017 International Conference on Computational Science and Computational Intelligence (CSCI'17), Las Vegas, USA, 2017.

[18] Twitter, "I. Twitter Developer Documentation - API Overview," 2017 [cited 2017; Available from: https://dev.twitter.com/overview/api].

[19] A. Connelly, V. Kuri, and M. Palomino, "Lack of consensus among sentiment analysis tools: A suitability study for SME firms," in Proceedings of the 8th Language & Technology Conference (LTC'17), Poznan, Poland, pp. 54–58, 2017.

[20] F. Heimerl, et al., "Word cloud explorer: Text analytics based on word clouds," in 47th Hawaii International Conference on System Sciences (HICSS), IEEE, 2014.

[21] S. F. Dierk, "The SMART retrieval system: Experiments in automatic document processing," IEEE Transactions on Professional Communication, vol. PC-15(1), G. Salton (ed.), Englewood Cliffs, NJ: Prentice-Hall, 1972.

[22] Zygomatic. WordClouds.com. 2017 [cited 2017; Available from: http://www.wordclouds.com/].