Sentiment Analysis of US political tweets and application to French political ones

Machine Learning for Natural Language Processing 2020

Clotilde Miura
ENSAE Paris
clotilde.miura@ensae.fr

Cédric Allain
ENSAE Paris
cedric.allain@ensae.fr

Abstract

Nowadays, social media yield a growing influence on politics. Indeed, politicians consider Tweeter as a good indicator to assess their popularity. In this context, NLP and particularly Sentiment Analysis rises a growing interest. We found that our sentiment model, based on BERT embedding, was able to correctly predict 91% of the "sentiments" of tweets of the first GOP debate in August 2015 on the test set[1] and that it could be transposed on French political tweets after translating them in order to assess the popularity of the 2017 French Presidential election.

1 Problem Framing

Tweets about politicians are often biased and their study can be used to refine campaign strategies by assessing a candidate popularity among a certain group of users (regional for example). Our hypothesis is that the sentiment analysis of tweets can be a good proxy of the public opinion during election campaigns as a complement of polls results. Our initial purpose was to train a sentiment model on the French political tweets published during the last Presidential election campaign of 2017. But the lack of good labeled data for French sentiment analysis led us to use the tweets of the first GOP (Grand Old Party) debate between the Republican candidates preceding the Presidential election of 2016. The idea was then to try to transpose it to French political tweets, after translating them, and to evaluate the popularity of the 5 principal candidates of 2017.

[1] Jupyter notebook, data and trained model are available at the following GitHub page: https://github.com/CedricAllainEnsae/sentiment-analysis-french-political-tweets

2 Experiments Protocol

Data The dataset used gathers **13781** tweets about Republican candidates published on August 7th 2015, ie, the day of the GOP debate. They are classified among 3 classes: negative, neutral and positive, with a level of confidence about the sentiment labeled ranging from 0 to 1. After discarding tweets with a confidence level inferior to 0.6, and splitting the dataset in train and test set, we had **9156** tweets for the training and **3053** for the evaluation with unbalanced distribution. Indeed, **61%** of the tweets are negative, **23%** neutral and **16%** positive.

Model used We decided to use a pretrained english version of BERT model [2] (*Biderectional Encoder Representations from Transformers*) (Devlin et al., 2018) and to fine tune it to our specific task of sentiment analysis, as transfer learning usually leads to better performances. BERT uses Transformer architecture with multiple head attentions and is trained as a Mask Language model. As it works at the subword level, there is no 'Out of Vocabulary' problem.

Implementation First, we had to preprocess our tweets. We used Bert Tokenizer which splits the tweets into several tokens at a subword level and convert them to id, for a total vocabulary of **30522** tokens. We also applied some specific functions to remove hashtags, etc., previously used in the labs. BERT maximum length for sequences is 512 tokens. As the maximum length of our tweets was 167, we padded all the sequences with 0 to reach the same length and used attention mask of 1 for tokens with no padding and 0 otherwise. Then we built a sentiment model by adding on top of Bert pretrained embedding layers a dense layer with 3

[2] 'https://huggingface.co/transformers/pretrained_models.html'

outputs for 3 classes, randomly initialized. Then, we retrained all the network.

Model training One main advantage of BERT over others language models is that it's highly parallelized. Therefore we could use Colab GPU to speed the training of the model. As specified in the initial paper(Devlin et al., 2018), only a very few number of epochs is necessary to train the model, so we only performed 5 epochs.

3 Results

Quantitative evaluation The model reaches **95%** of accuracy on the train set. But as the data is unbalanced, we chose to compute a report of several metrics on the test set. The following table presents our results.

sentiment	precision	recall	f1-score
0	0.95	0.93	0.94
1	0.78	0.87	0.82
2	0.91	0.85	0.88
average	0.91	0.90	0.91

Table 1: *Performances on the test set*
0: negative, 1: neutral, 2: positive

We have a global accuracy of **91%** on the test set. The performances are very good on the the majority class, the negative one and also on the positive tweets even if they amount for the smaller proportion. Precision is less good on the neutral tweets probably because they are globally more difficult to predict even for a human.

Qualitative evaluation We tried to analyze some tweets the classifier missclassified. We found that those tweets, even for a human were very difficult to classify. Sometimes we even agreed more with the prediction of the classifier than with the label.

Application on French tweets We scraped French tweets mentioning the 5 principal candidates of the French Presidential election from early January 2017 to the date of the first day of the election, April 23rd. After translating them into english, we used our model to predict the sentiment of those tweets in order to assess the popularity of each candidate: Emmanuel Macron, François Fillon, Marine Le Pen, Jean-Luc Mélenchon and Benoit Hamon. Popularity can then be estimated by the percentage of positive tweets or by a score averaging the proportions of positive, negative and neutral tweets. Thus, for each candidate, his or her popularity score is calculated as follows:

$$\sum_i w_i \times p_i \times (p_i - \bar{p}_i)$$

where i is the label (negative, neutral, positive), w_i is the weight associated for the label (respectively -1, 0.5 and 2), p_i and $\bar{p}_i$ are the proportion of tweets predicted of the label respectively among the candidate's tweets and among all predictions. This popularity score gives us a ranking that is close to that of the first round.

4 Discussion/Conclusion

We observed that transfert learning based on BERT embedding applied on GOP tweets gives us quite good results, even though some political tweets are particularly difficult to categorize, even for a human. We also observed that the sentiment classification is done in an absolute manner, for the GOP tweets as well for the tweets about french candidates: a negative tweet mentioning a candidate will be classified as such, even though it is positive towards another candidate. A solution to that, and a potential future work, would be a more complex model that would also take in input the person for whom the sentiment analysis is done. Hence, each tweet, along with his sentiment classification, will be tagged with the person it is about and if the tweet is in favour or against this person.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.