Classifying emotion intensity in tweets (Theme 5A)

José Miguel da Costa Simões   
*Faculty of Engineering of the University of Porto*  
Porto, Portugal  
up201704317@fe.up.pt

DJosé Rafael Fidalgo Fonseca Matias   
*Faculty of Engineering of the University of Porto*  
Porto, Portugal  
up201706413@fe.up.pt

Paulo André Areias Gomes Leal Pinto   
*Faculty of Engineering of the University of Porto*  
Porto, Portugal  
up201706415@fe.up.pt

*Abstract*— The aim of this study was to experiment with different NLP techniques and classification models, and evaluate their respective performance in the context of a natural language classification problem. Three different and well-known algorithms were explored as a result, and each was analyzed in terms of performance quality, as well as time spent to reach a given solution (or, in this case, to train the machine).

Keywords—Artificial Intelligence, Natural Language Processing, Naïve-Bayes, Decision Trees, Neural Networks

# Introduction

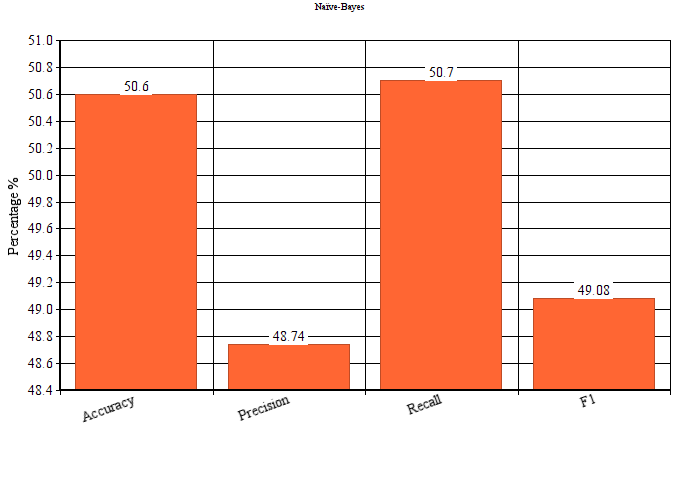
Algorithms used for Natural Language Processing are used to process text and overcome certain challenges, such as speech recognition, natural language understanding and natural language generation. This study falls on the second category, as there was a need to classify the emotion intensity in tweets. The goal of the study was to explore different NLP algorithms in this context, so as to learn more about how each one works.

# Description of the problem

## The Task

This task was promoted by CodaLab as a competition in 2018, where competitors had datasets for four different emotions – anger, fear, joy and sadness. Given a tweet and one of the four mentioned emotions E, competitors had to classify the tweet into one of four ordinal classes of intensity of E that would best represent the mental state of the tweeter. Four different, and isolated datasets were available, one for each emotion. It was considered to be in our best interest to explore other approaches to this task (besides the original one). Thus, the algorithms in question were used to classify the tweets as a whole. This resulted in two different contexts, one where the program would classify just the emotion intensity of the tweet (unconcerned about which of the emotions it was dealing with).

## The Theme

The goals set for this theme were to explore different components and techniques that make up natural language processing. Firstly, the data was to be explored, in order to make useful datasets out of the already-provided datasets. This step required the investigation and usage of different techniques for the generation of structured data. Additionally, it was required to use and implement different learning algorithms, while analyzing their performance. This could be achieved by studying the number of errors obtained during the learning period, the confusion matrix, statistical values like precision, recall, accuracy and F-measure, and time spent in training/testing.

# Approach

# Experimental evaluation

## Naïve-Bayes

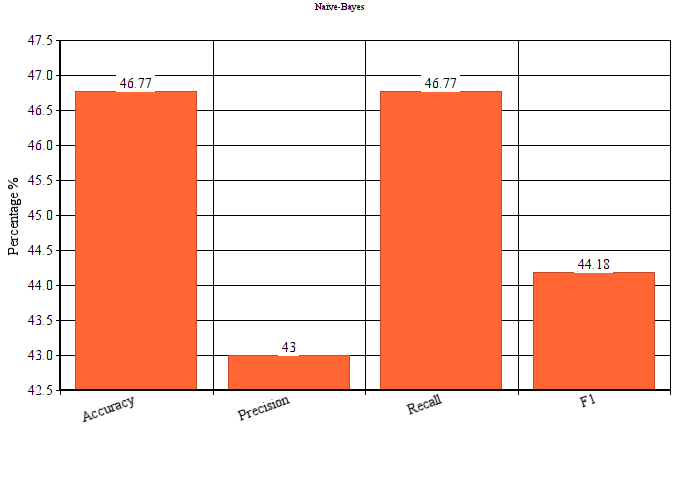
Naïve-Bayes is an algorithm that analyzes a given dataset, using the statistics it collects when making a prediction. It checks the frequency of each word and its classification in each case. After going through the entirety of the dataset, it stores the likelihood of a given tweet – or any other text, for that matter – falling under each possible classification **for each word** that was analyzed. When it makes a prediction, it checks said likelihood for each classification and multiplies the likelihood of all words for each classification. In the end, all that is left is for it to pick the highest probability out of all options.

It clearly shows the importance of using a text-processing technique, since we do not want words such as “a” or “the” to be given any importance (most of the times). Thus, stemming was applied to the parsed tweets. This way, not too much importance was given to said words and conjunctions. Words starting with “@’s” were also removed (note that these words are common in tweets to tag someone).

The results presented below show the average of the results of all emotions for each parameter. We can see that the Accuracy and Recall parameters were higher than Precision and F1. The Recall parameter shows how many tweets with an intensity I were actually predicted to be of said intensity. This parameter works in a similar way as accuracy, which explains the similar values.

Precision scored lower, which means that our algorithm classified certain intensities more often then it should, since this parameter portrays, given all cases of an intensity I, how many were actually of said intensity. As expected from the statistical analysis, this algorithm ended up having an intensity level which was most used to label the tweets. Most often it was 0 – the tweeter does not show the emotion in question. This can be explained by reading the dataset, which contains many examples of intensity level 0.

The following graph depicts the results obtained when the emotion in question was not taken into account. In those cases, the algorithm predicted only the emotional intensity of tweeter, no matter the emotion. This situation was explored, thinking improvements in the results would improve from larger amount of examples and, consequently, the dataset. However, this is not observable. In fact, after some reflection, it was concluded that the examples given did not contain a variety of classifications – datasets for different emotions all had more examples for intensity level 0 than others – which led to an actual decrease in performance.

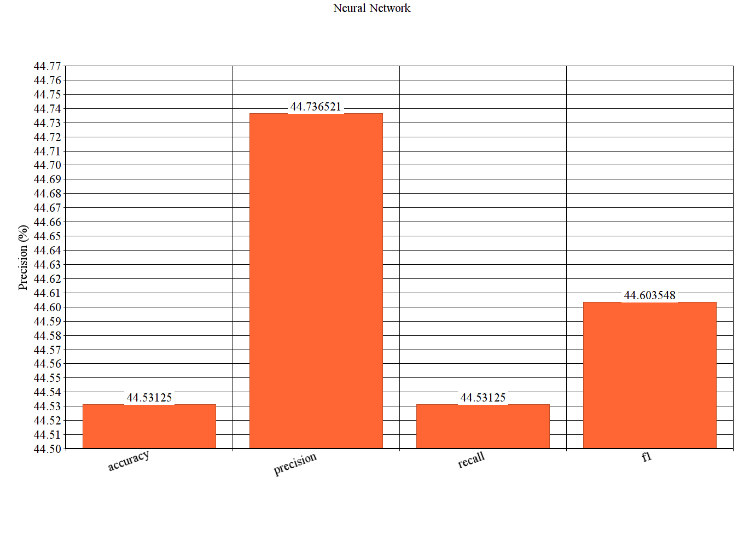


## Decision Trees

## Neural Networks

Neural Networks is inspired in the human brain, as they have hidden layers that represents the brain cells. Such systems "learn" to perform tasks by considering examples, generally without being programmed with task-specific rules. They don’t recognize the object that is being analyzed, but it generate identifying characteristics from the examples that they process.

We used neural networks to analyze the emotions and the intensity of the emotions in some given tweets. The results of the use of the algorithm Neural Networks is shown in the graphic. Although the precision is higher, it is not so different from the other stats. All of the emotions had similar results, so there is an illustrative graph of those results regarding the emotion “anger”:



The more iterations there were, the less was the training loss. The number of iterations was approximately 340 for the standard use of MLPClassifier (without changing its parameters), as is stopped when the training loss did not improve more than tol=0.000100 for 10 consecutive epochs, even though the iterations limit was set to 1000.

Adding hidden layers improved the stats until a certain point. With just 1 layer and 1 neuron, the number of iterations surpassed the 1000 limit and the training loss was still too high at that point.

“Iteration 1000, loss = 0.55477678”

With a lot of hidden layers, but only one neuron in each, the training loss was also very high, but the number of iterations didn't surpass 600.

“Iteration 574, loss = 1.40947317

Training loss did not improve more than tol=0.000100 for 10 consecutive epochs. Stopping.”

With the same number of neurons, but distributed only by two hidden layers, both the number of iterations and the training loss were considerably lower.

“Iteration 456, loss = 0.02291142

Training loss did not improve more than tol=0.000100 for 10 consecutive epochs. Stopping.”

# Conclusions

According to the results for each of the three explored algorithms, the "fear" emotion is the one that has higher results (accuracy, precision and recall). That was due to having more tweets for the program to train the "fear" emotion (2252 tweets) than the other three emotions. That proves that the more the program tests, regardless of the algorithm, it has better decision making the more train it gets. After the "fear", the emotion that had higher values was "anger" (1701 tweets). The difference between the stats of "anger", "sadness" and "joy" wasn't very distinctive, as the number of tweets used for training was not so different from each other (for "joy" there were 1616 and for "sadness" there were 1533 tweets). Apart from those results, we also tried to input single tweets for tests and check the program answers. We realized that there were good evaluations from the program, but some answers were not what we expected for the intensity of a certain feeling. As all the algorithms are supervised learning algorithms, the more we train them, the precision, accuracy and recall are improved.

Overall, taking into account that this task was part of an international competition, it is possible to say that we are satisfied with the results. The program is able to do what was asked for, even if at a lower rate of success than desired. This project allowed us to learn more about supervised learning and artificial intelligence in the context of natural language processing. Moreover, the develop was exciting and made us more interested on the subject

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*
2. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

**IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**