Text Classification

Classifying Propositional Content in annotated Discourse Units

PLN 2021/22

Caio Nogueira, up201806218@edu.fe.up.pt

Telmo Botelho, up201806821@edu.fe.up.pt

21/04/2022

Problem Definition

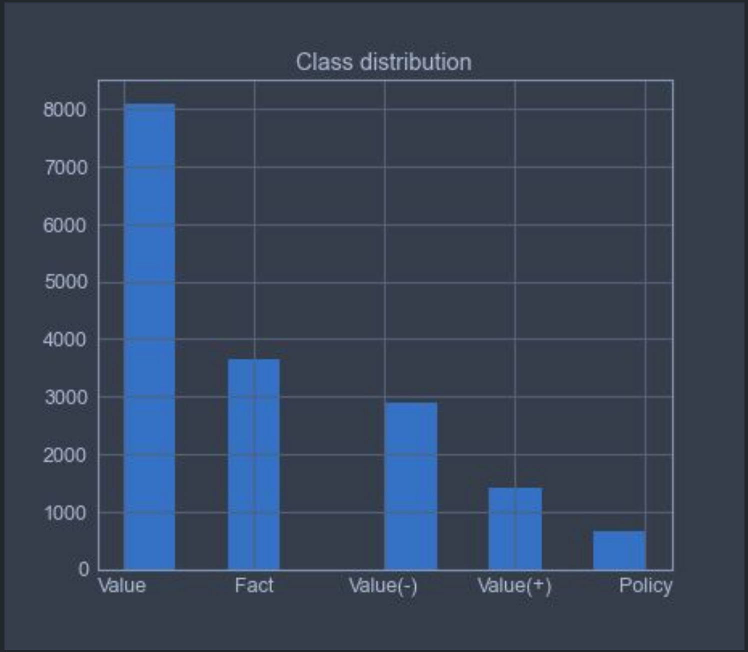
An annotation project carried out in opinion articles from the Público newspaper:

- Classifying text spans into propositions of **Value, Policy, Fact,** and within propositions of value, distinguish those with **positive** or **negative** connotation.
- Text spans taken from articles which contain topic, author and other associated metadata.

Our job consists of developing a model for this multi-class classification problem with the best possible.

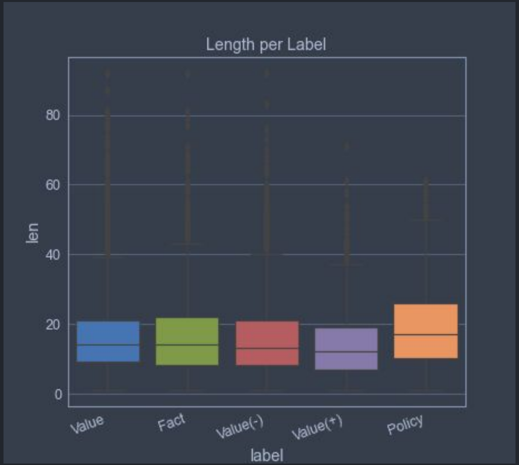
The Dataset

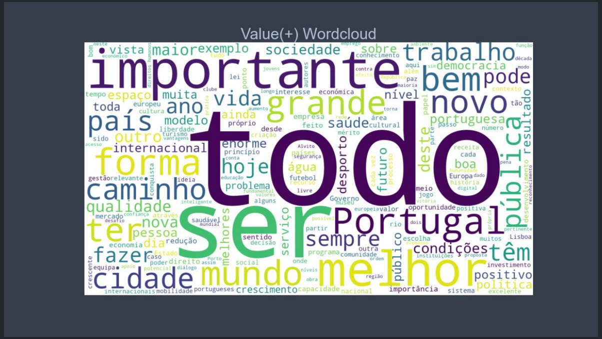
- 373 different articles;
- 16743 annotations;
- 12302 unique text spans annotated;
- 8 different topics;
- Imbalanced labels - 1:12 ration between majority and minority classes.
- 0 missing/inconsistent values - no **data cleaning** was necessary.

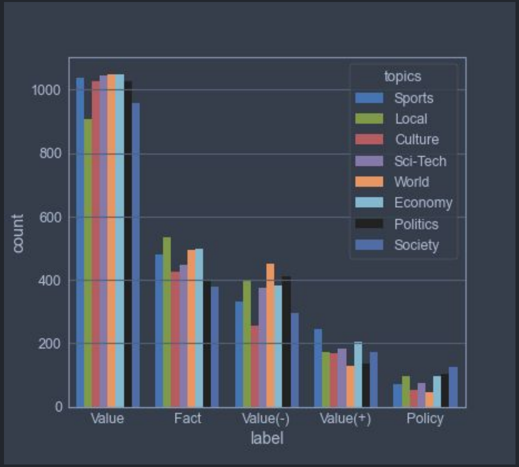


Data Exploration

- Most of the articles are from 2019;
- The average text span contains 16 tokens (std= 10.7) before any text normalization;
- *Policy* text spans are slightly larger than the others;





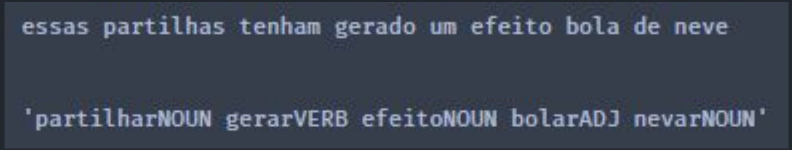


Data Representation

Before training the models, we applied several **text normalization** techniques (e.g., stemming, lemmatization, ascii folding, lowercasing). Moreover, we dealt with the “overlapping“ annotations: text spans with more than 1 annotation were assigned a label by majority voting.

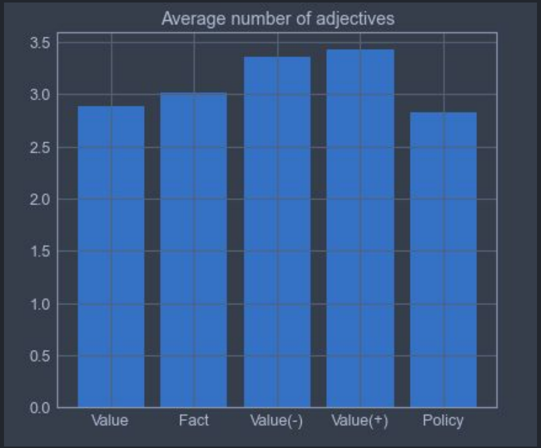
In our experimental setup, we used 2 different text representations:

- Tf-idf feature vectors with POS-tags and topics;
- Pre-trained representations.



```
essas partilhas tenham gerado um efeito bola de neve


'partilharNOUN gerarVERB efeitoNOUN bolarADJ nevarNOUN'
```



Data Representation

Besides tf-idf feature vectors, we also performed text classification based on a word2vec model trained on the articles dataset, as well as some pre-trained models from NILC:

- FastText skip-gram model with 100 dimensions;
- FastText CBOW model with 100 dimensions;
- Word2Vec skip-gram model with 100 dimensions;
- Word2Vec CBOW model with 100 dimensions;

The results obtained from these models were similar (accuracy = 48%).

Experimental setup

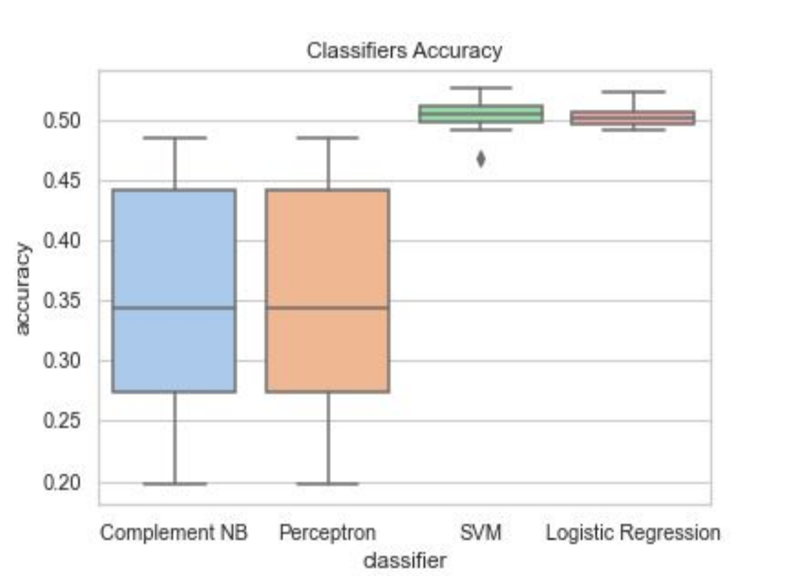
The **pipeline** created included several steps :

- Transforming textual data into a tf-idf feature vector for each element of the corpus;
- Resampling techniques (Random Oversampling, Random Undersampling, SMOTE);
- Calculating class weights;
- Cross-Validation (using 5 folds);
- Scaling the data ($\boldsymbol{\mu} = 0$; $\boldsymbol{\sigma} = 1$);
- Hyperparameter Tuning (using GridSearchCV and HalvingGridSearchCV) applied on the chosen algorithms: Logistic Regression, Linear SVM, Perceptron, Decision Trees, Random Forest, Gradient Boosted Trees;
- Analysis and comparison of results obtained (classification report, confusion matrix, accuracy, precision, recall, f1 score).

Results comparison

The following box plot contains information regarding accuracy scores during cross-validation. These models were trained using tf-idf feature vectors (which achieved the best results).

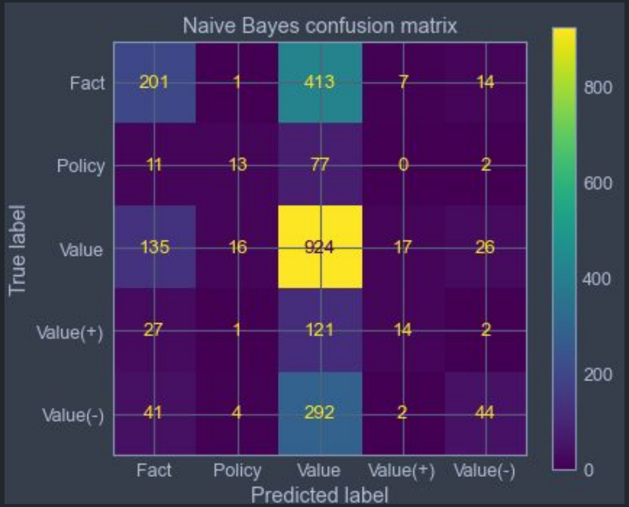
- The metric used for comparison was accuracy;
- The best algorithm and probably the one to choose for this problem is the **linear SVM** algorithm (accuracy = 53%)**.**
- SVM and Logistic Regression with **L2 regularization** achieved the best results.



RESULTS DISCUSSION

After exploring different algorithms, as well as **data preparation**, **feature engineering** and **resampling** techniques, we reached the following conclusions regarding the problem.

- Conventional resampling techniques led to poor results with textual data;
- SMOTE doesn't work well in a high dimensional feature space (such as this text classification problem).
- Some models could not be completely explored, as Grid Search would take too long;
- Sequence labeling makes a small but noticeable difference on the results (up to 4 percentual points).



CONCLUSIONS

Conclusions:

- Working with textual data can be challenging due to the high amount of dimensions;
- After comparing the results obtained with other 5-star text classification problems, we believe that the results were satisfactory.

Future Work:

- Results could probably be improved with a text augmentation phase. Some approaches included **back translation**, using a **global synonym list**.