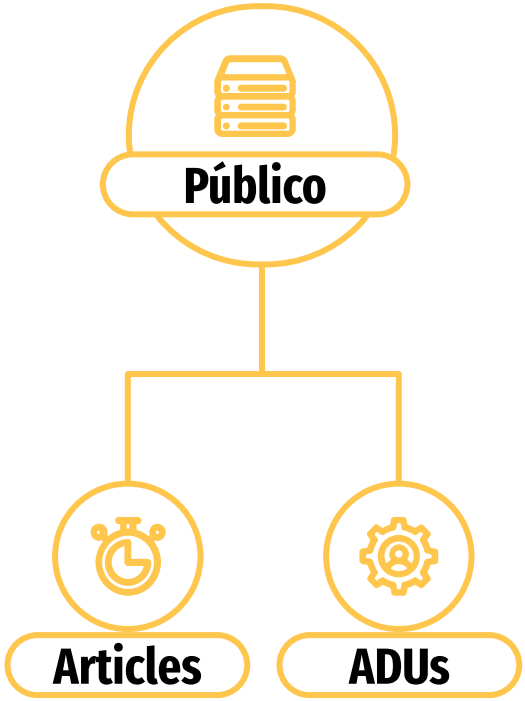
PLN

Inês Quarteu (up201806279@edu.fe.up.pt)

Rúben Almeida (up201704618@edu.fe.up.pt)

FEUP 08-06-2022



Roadmap

1. Previously on PLN...
2. Problem Definition
3. Four Pipelines
4. Before Training
5. Model Testing
6. Domain Adaptation
7. The “Best” Model
8. The “Rational” Model

Previously on PLN...

373
Opinion Articles from the newspaper Público

16743
ADUs annotated by 4 different annotators

0.661
Accuracy (Macro)

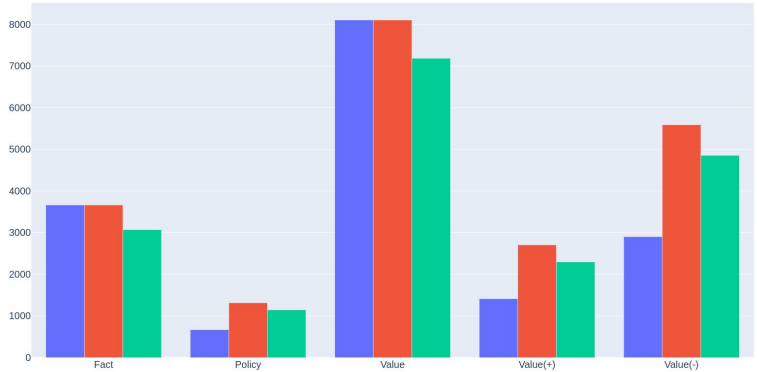
0.656
F1 Score

Data Augmentation
Back Translation Method

Pre Processing
Produced improvements

Disagreements
Removed 1384 ADUs

SVM with TF-IDF
The Best Combination



Problem Definition

Two Tasks

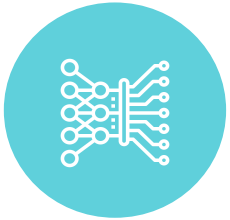
Sentence Classification

Predict the correct label for Argumentative Discourse Units

Domain Adaptation

As Bonus, perform a pre trained focusing on Domain Adaptation

Usage of Transformers



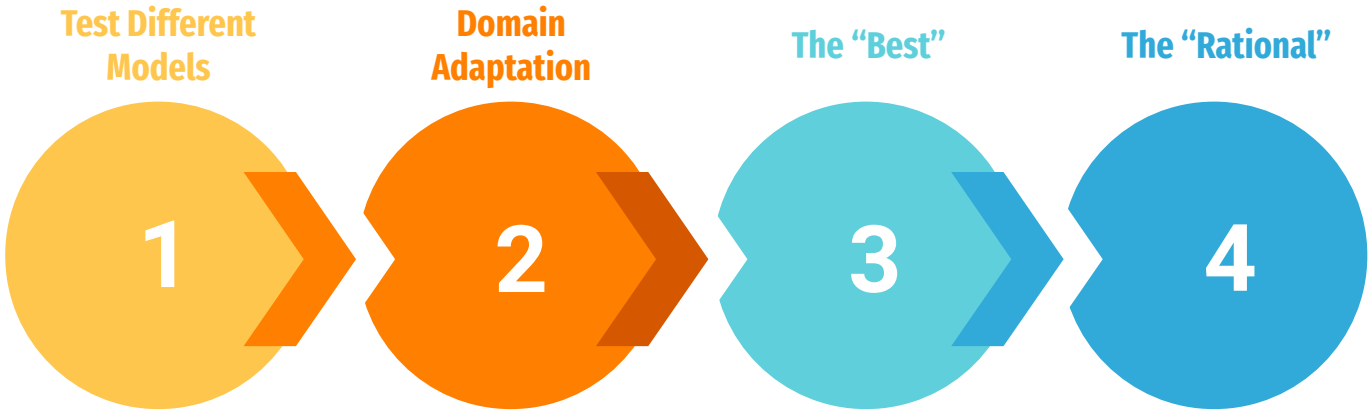
Deep Learning Strategies

Take advantage of Hugging Face models to apply transformed based deep learning techniques to address the classification task

Usage of Encoders

Sequence to Feature Vector Representation technique.
Find the correct BERT, Alberta, Roberta, Electra to address the challenge

Four Attempts - Four Pipelines



Before Training

Data Augmentation

- Use Spacy Lemmatization
- Use Translation.
- Use Google Translate
- Use Helsinki-NLP models available in Hugging Face
- Augment with Back Translation but also with English and Spanish
- **Only with Back Translation**
- All minority classes
- **The Policy Class**
- **None**

Loss Function

- Use the default Trainer
- **Overload Hugginface Trainer Cross Entropy without custom weights**
- **Cross Entropy with custom weights**

Optimizer

- Use the default option (AdamW with 5e-5 learning rate)
- Use AdamW with 1e-3 learning rate
- Ada Factor with lr_scheduler for adaptive learning rate
- FP16 Acceleration

Data Splitting

- Stratified 70-30
- 50 for validation 50 for testing

Model Testing

Multi Lingual

0.49
bert-base-multilingual-cased

0.43
bert-large-multilingual-cased

0.51
amberoad/bert-multilingual-passage-reranking-msmarco -> 0.51

0.51
cardiffnlp/twitter-roberta-base-sentiment

Portuguese

0.60
neuralmind/bert-base-portuguese-cased

0.47
neuralmind/bert-large-portuguese-cased

0.49
dlb/electra-base-portuguese-uncased-brwac

Same Training

3 epochs Default Trainer Parameters.
Stratified Split.
Results Evaluated in Accuracy test set

Base vs Large Models

Base models performed always better than large ones

Case vs Uncased

Reject Case folding reveal minor gains.
Case, when available, performed better

BERTimbau

The most robust model as the cased base version of Portugues BERT

Domain Adaptation

Based on Word Masking

Unsupervised training based on masking different tokens at each step

Adapt the Network to Our Corpus

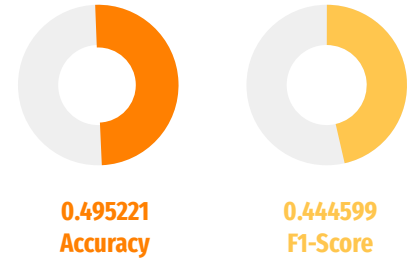
Encoders are trained in generic data. Fine tune them to our particular vocabulary.

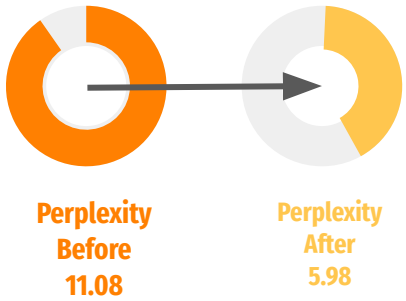
Two Steps Training

Start by freezing the layer. Train only the last layers, then unfreeze it for few passes

Reduce the Perplexity of the Model

Is this model use to deal with this vocabulary? The less perplex the better





The “Best” Model

No Data Augmentation

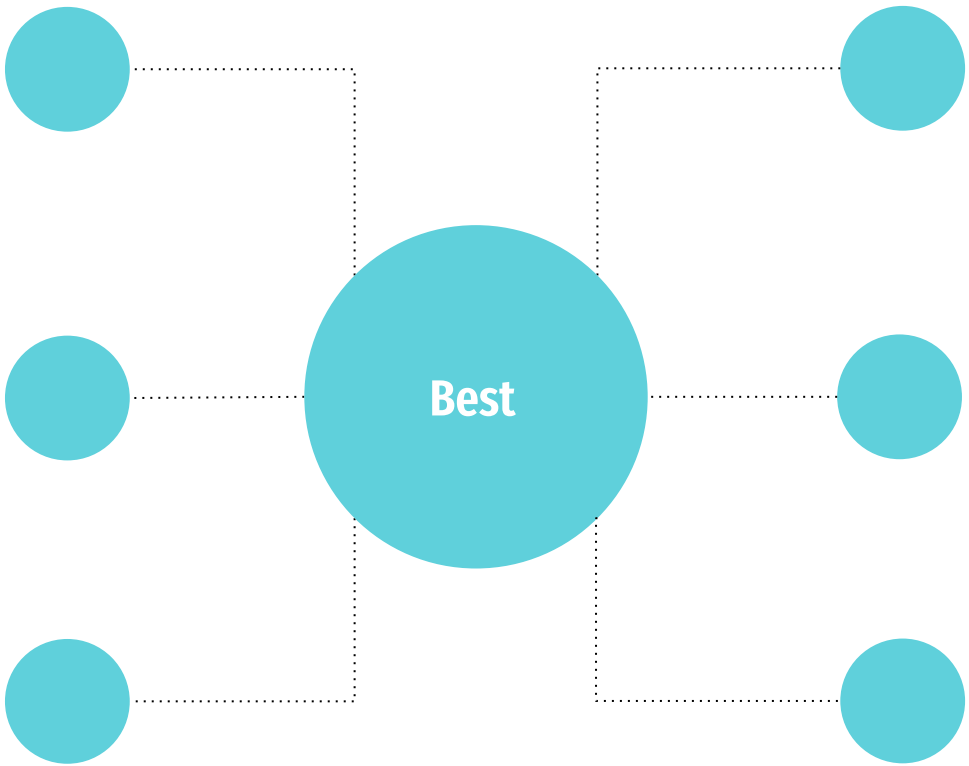
Model Performs Better Without Augmentation

Cross Entropy without Custom Weights

Overloading Trainer to use Pytorch CrossEntropy without Weights boosted results

Single Unfreeze Training

Don’t mind about freezing layers, simply train the model



Better Performance Metrics

Quickly Overfits

The analyses of the Loss function shows that the model is simpleing overfitting to the train input

Does it learn anything?

Let's check the Confusion Matrix

The “Rational” Model

Data Augmentation

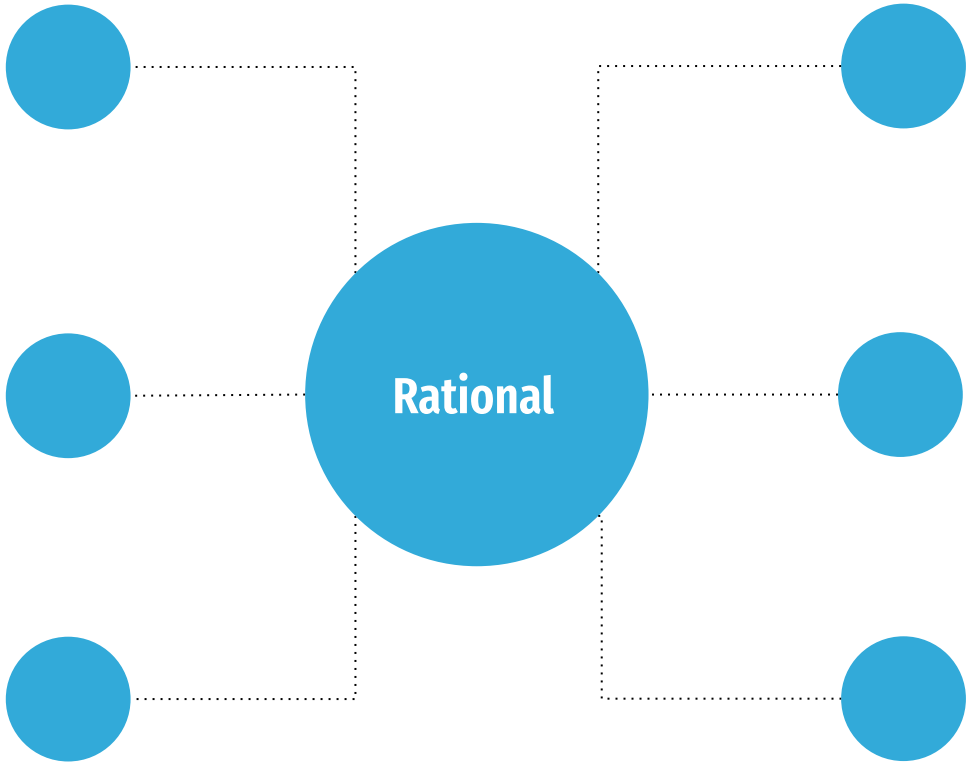
Augment with back translation in the Policy Class

Cross Entropy With Custom Weights

Overload the Trainer Class, in order to customize the weights. Use compute_class_weight from sklearn followed by a softmax to estimate those weights

Two Steps Training

Train the last 5 layers for 5 epochs, then unfreeze the whole network and train two more epochs



Performance Metrics Decay

Resilient to Overfitting

The loss function shows a more adequate curve for a deep learning problem

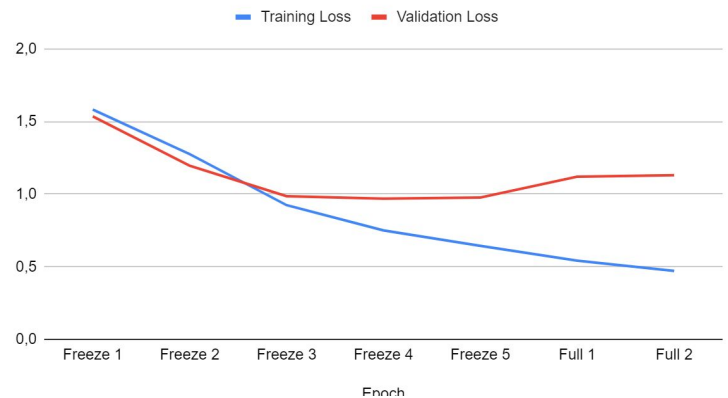
Does it learn anything?

Let's check the Confusion Matrix

Loss Function Comparison

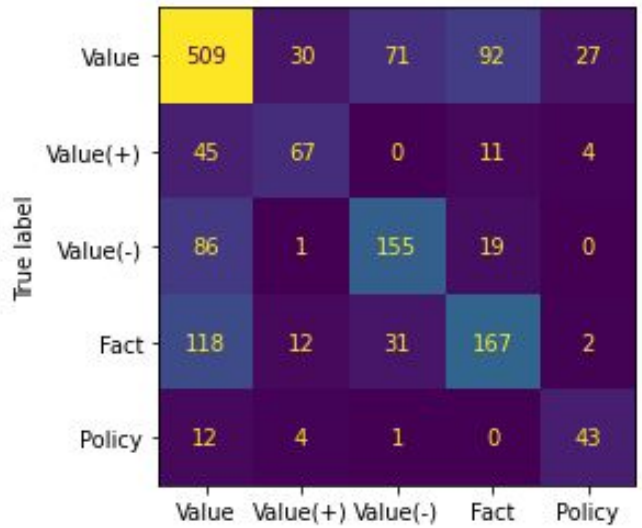
The “Best”

The “Rational”

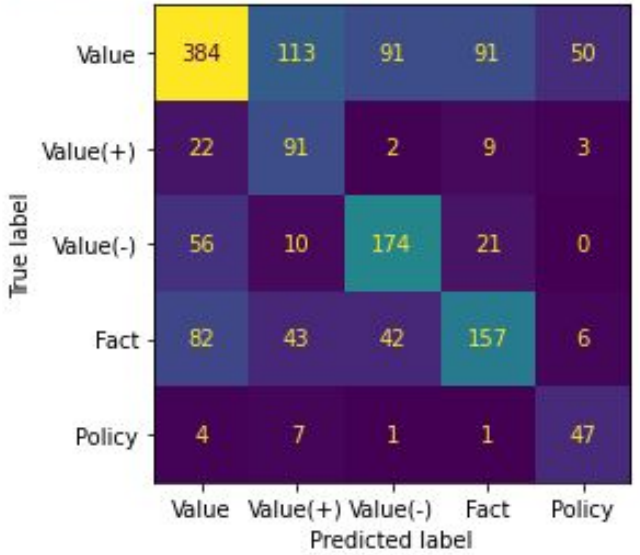


Confusion Matrix Comparison

The "Best"



The "Rational"



Conclusion

- We covered several transformer models and methods, assessing the requirements.
- The results were disappointing. We were unable to find the magical solution for result boosting.
- Transform models perform better than the traditional machine learning baselines.
- Our first project was much more complete in analyse and time available, we were able to score a good result.
- The success of the first project creates comparative analyses that originate the main disappointment for this second approach.