DUFLOS
Inès

Project Report

Introduction

The goal of this project is to classify sentences within the PubMed 200k Randomized Controlled Trial (RCT) dataset. The dataset, sourced from PubMed, consists of sentences categorized into various classes pertinent to medical research. Our investigation revolves around comparing the performance of three models: a baseline model utilizing bag-of-words and a model leveraging pre-trained biomedical word embeddings. By employing these models, we seek to enhance the understanding of text classification in the biomedical domain and ascertain the efficacy of incorporating domain-specific embeddings.

Methodology

Creation of the dataset:

To create the train and test dataset I had to load both files and separate the different elements in those files using the split function. I also had to remove the empty lines and the id numbers corresponding to every trial. Once it was done and I only had the labels and the texts left I put them in two data frames named train_df and test_df.

Preprocessing of data:

To do the preprocessing of the texts I used the scispacy tokenizer and lemmatizer and applied it the all the texts and put the preprocessed result in a new column of the datasets called 'preprocessed_text'.

Baseline model with bag of words:

Once I had my datasets and my preprocessed texts I did a TF-IDF with it and use it to do a classification using 3 different models, multinomial naïve bayes, logistic regression and random forest.

To visualize everything in a better way I printed the confusion matrix for the 3 models.

Model with pre-trained biomedical (word) embeddings:

This part was the most difficult one for me because downloading the pre-trained biomedical embeddings took way more time than I expected, indeed every time I wanted to download a biomedical embedding google collab was crashing. So, in a first time I only used ‘word2vec-google-news-300’ and then once I finally find a way to load my pre-trained biomedical word embeddings, I applied the same method I used with the ‘word2vec-google-news-300’ word embeddings but using my biomedical word embeddings.

The pre-trained model embeddings I used was the 200-dimensional word2vec from this website: https://github.com/piskvorky/gensim-data/issues/28.

Once I loaded my word embeddings model, I had to vectorize my tokens to be able to use my model to classify my texts. And I applied a logistic regression to classify my texts.

Evaluation, comparison of the models performance

I got the following results for my different models with the baseline models in blue and the model using pre-trained biomedical embeddings in green:

	Precision	Recall	F1 Score	Accuracy
Baseline Model				
Multinomial naïve bayes	0.522	0.522	0.522	0.68
Logistic regression	0.66	0.67	0.664	0.75
Random forest	0.614	0.632	0.623	0.73
Word Embeddings Model				
Pre-trained biomedical embeddings	0.441	0.485	0.428	0.485
Word2vec-google-news-300	0.475	0.501	0.449	0.501

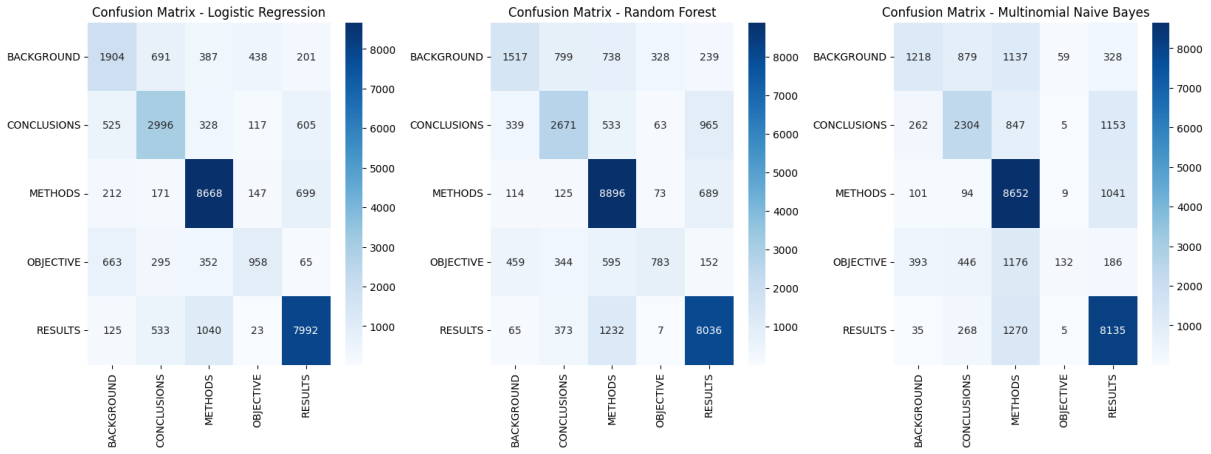
From the evaluation scores obtained for my classification from my baseline models that the Logistic Regression performs the best across all metrics (precision, recall, F1-score, and accuracy) and the Multinomial Naïve Bayes has the lowest performance among the baseline models. About Random Forest performs, it moderately well but does not outperform Logistic Regression in terms of accuracy and F1 score.

For the Word Embeddings Models we can observe that both word embedding models seem to underperform compared to the baseline models, especially in terms of precision, recall, and F1-score. However, surprisingly, the Word2vec-google-news-300 model performs a little better than pre-trained biomedical embeddings, but it still is not that great.

For this context the baseline models, especially Logistic Regression, seem to be more effective. The pre-trained model I chose might have not been ideal for the texts and I maybe could have had better performances by choosing the 400-dimensional word2vec pre-trained biomedical embeddings, but I did not use the 400-dimensional word2vec because it was too big for my computer to download it and load it in google collab.

Results, discussion of your results

I got the following confusion matrices from my classifications:



With those confusion matrices we can observe different things and problems that the performances metrics did not show. This is especially true for the word embeddings models. Even if the performance were not catastrophic, we can see that there is a real problem in the prediction of certain labels like the Background label, the Conclusions label and the Objective label. Those labels are mostly misclassified as the Methods label or the Results label which are the most present labels in the dataset, so that means that when the model does not know the label of a text it labels it as a Methods or Results because it is most likely to be true but that creates a lot of misclassifications.

For the baseline models we can see that there is misclassification but not as much as the embeddings models. However, we can still see this kind of misclassification with the Multinomial naïve bayes model. But, with the other 2 models, we can clearly see the diagonal created by the correct classifications of the models.

Conclusion

To conclude this project, in my case the most efficient model was the logistic regression.

The word embeddings models were not very efficient, but it might be because I chose and used biomedical model that did not have enough dimension or because it was not fitted for my data set or did not receive enough training.

Even if the logistic regression had the better results with my data set out of all the models I used, it still does not have excellent results, so it still has quite a lot of misclassifications.