Andreas Styliaras
Britt Westerhof
Thor Tiefenthal
Paworapas Kakhai

GROUP 2

TOPIC ANALYSIS WITH BERT AND ROBERTA MODELS

PROJECT POSTER

For topic modeling, we used a Google Play Store app dataset with more than 2 million instances [3]. This dataset includes categories of various apps that are available on the Play Store. Three of these categories are 'Sports', 'Entertainment', and 'Books & References', which work well with the 'movie', 'sports', and 'book' topics in the test set. In order to make sure the model did not predict any unrelated topic for the test set, apps with other categories were excluded from the training data. Since the model cannot recognize words, we changed each topic from a named string to an integer ranging from 0 to 2 for both the training and test data. Two of the models we used to train our data were the pre-trained transformer models BERT [2] and RoBERTa[4][5]. These models will take 2000 samples of training data which consists of the app name with corresponding category and attempt to recognize patterns in this data. This data was further split into 200 samples of validation data and 1800 of training data to tune the hyperparameters. After this we tested our models on 10 samples of test data and compared the results from the predicted topics to the actual topics of the test data. With the word cloud tool, we can analyze the most often mentioned word among the three different categories: Sport, Book& References, and Entertainment (Movie).

BERT Classification Report

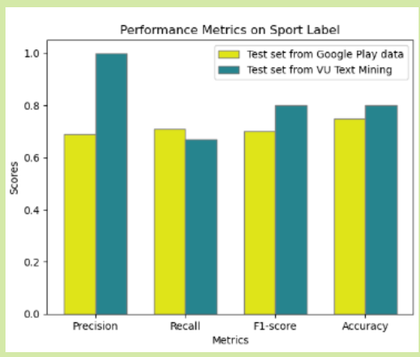
Category	Precision	Recall	F1-Score	Support
Sports	0.75	1.00	0.86	3
Movie	0.00	0.00	0.00	4
Book	0.50	1.00	0.67	3
Accuracy			0.6	10

As seen in the classification report, the BERT model does not seem to perform that well on the test set, achieving an accuracy of only 0.60. This could be explained by the fact that the test set only consists of 10 instances of data. Another reason might be because the training data and test data are from two different datasets, meaning they do not fully align even when they are divided into similar topics. This is especially visible for the movie topic, since the model did not manage to correctly predict, or even find, any of the 4 samples for the movie topic. The equivalent to the movie topic in the test set is actually the entertainment category in the Google Play Store dataset, so the model might have a difficult time picking up on the similarities between these topics. Since the total of samples for the movie test set is only 4, it could also just be explained by random chance that the model did not correctly identify anything for this topic. The higher performance for both the sports and the book topics might be related to the Google Play Store dataset actually having a 'Book & Reference' and 'Sports' category. It could be easier for the model to correctly predict the topic for the test set when the topics are equal between the test and training sets. Lastly, the difference in performance between the three topics could also be because the 'movie' topic in the training set only has 292 instances, while the sports and book topics have 688 and 820 instances respectively. This distribution can potentially result in the model predicting one of the more likely topics more often, rather than predicting the topic with less instances.

After training the RoBERTa model on the Google Play Dataset against the test dataset from the sentiment-topic-test, we can observe that the precisions for the labels on Sports and Books are very high while for move it is reasonable. This means the false positive value for both labels is relatively low and this also results in the model's accuracy. The factor that could contribute to this is the RoBERTa model performance on 1800 instances of training data against 10 instances of testing data.

RoBERTa classification Report

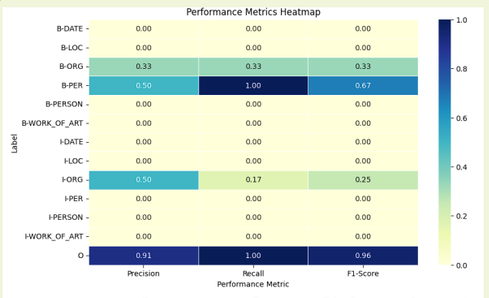
Category	Precision	Recall	F1-Score
Sports	1.0	0.67	0.8
Movie	0.67	1.0	0.8
Book	1.0	0.67	0.8
Accuracy	0.8		



When comparing the models' performance between using the test set from the sentiment-topic-test dataset and the test set from Google Play data on Sport Label, we can see that the test set from the sentiment-topic-test performs better. This is because there are only 10 instances of testing data while there are 1600 instances of training data. Thus, the prediction is more accurate. The test and train dataset from Google Play performs worse which results in precision and accuracy: 0.69 and 0.75 respectively. While the accuracy and precision for the sentiment-topic-test are 0.8 and 1.0. This is because there is more testing data and the models deviate while predicting the labels. Thus, the performances of the two models are significantly different.

NERC WITH SVM MODEL

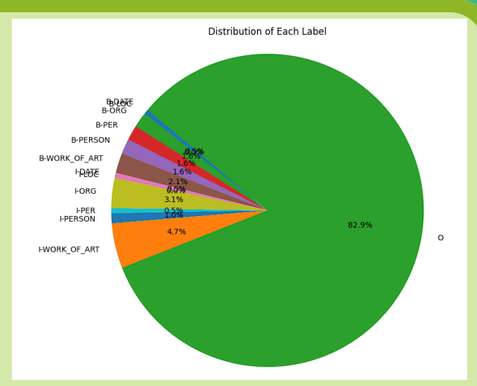
For the task of NERC we decided to use an SVM from the sklearn library. We used the SVC version which was shown to us during the Lab sessions and was discussed in the lectures [6]. One reason for choosing this method was due to resource constraints, as SVM are relatively easy and efficient to train.
We extracted various features from the data for the SVC to use. We wanted to have lots of features for the SVC to separate in the feature space [7]. We represented each token as a vector and trained and tested the classifier. The data we used to train the model was the CONLL 2003 dataset. This dataset was presented to us in the labs and lectures. We decided to use this dataset because it is a popular, annotated data set that is widely used for named entity recognition. We combined this data with the test set that was provided to analyse our technique.

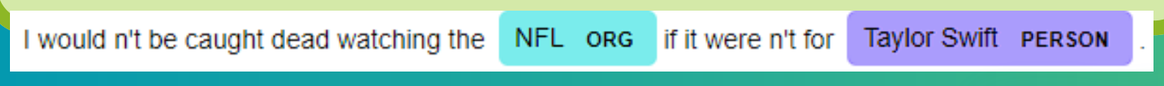


Metric	Precision	Recall	F1-Score	Support
accuracy	nan	nan	0.850000	193
macro avg	0.170000	0.190000	0.170000	193
weighted avg	0.790000	0.850000	0.820000	193

In our results we achieved a weighted Precision of 0.79, a weighted Recall of 0.85 and a weighted F1-score of 0.82. The best performance was obtained on the 'O' label. With a Precision, Recall and F1-score of 0.91,1.00 and 0.96 respectively. Reasons for this may be due to the imbalanced distribution despite being similar in training and testing. Thus predicting every label as 'O' will already give a good performance score. We had several low scoring labels. In all these cases the number of occurrences in the test data were either below 3, not available in the training data, or were due to an annotation error.

Some limitations we had was that the data set we trained on had few NER labels than on our test set. Thus classifying these extra labels such as 'B-WORK_OF_ART' wasn't possible for our classifier since it was not trained with this label. Also, the feature selection may not have been rich enough. Thus if we had more time we could represent better features for our data such as n-grams or using embeddings. In the future we could also train the classifier on a data set that more closely represents the test set that we are analyzing. Overall the classifier performed well on the 'O' class but scored poorly on the other classes. This may be due to the class imbalance in the test set or due to limitations of the linear classifier or the choice of our features not good enough feature selection.





VADER SENTIMENT ANALYSIS

For the task of sentiment analysis we decided to use the lexicon based VADER sentiment analyzer. This method was used, as the test set, containing ten sentences, resembles tweets or posts on social media. VADER sentiment analysis is good for social media text, as VADER is trained on a vast amount of social media data and therefore it understands the nuances, slang, and abbreviations commonly used in social media posts, tweets, comments, etc. [1]. This allows it to accurately interpret sentiments expressed in such language. It also includes a lot of current slang that may be used to express how a person is feeling in social media text in its lexicon. Unlike many machine learning models, VADER does not require training on specific datasets. This makes it convenient to use for sentiment analysis tasks. In this case, VADER was implemented by making use of the Natural Language Toolkit library in Python.

After applying VADER on the test set and even lemmatizing the tokens of the sentences in the test set, the prediction of VADER did not agree with most manually annotated sentiments of the test set. For "negative" class, precision, recall and f1-score are 0%, meaning that none of the instances predicted as negative were actually negative and none of the actual negative instances were correctly predicted. For the "positive" sentiment 20% of instances predicted as positive were actually positive and 33% of actual positive instances were correctly predicted as positive. For "neutral", 33% of actual neutral instances were correctly predicted as neutral and 33% of instances predicted as neutral were actually neutral.

VADER classification Report

Sentiment	Precision	Recall	F1-Score
Negative	0.00	0.00	0.00
Neutral	0.33	0.33	0.33
Positive	0.20	0.33	0.25
Accuracy	0.20		

This results in a weak overall Accuracy of 20%.

For example taking a look at the sentence: "I just finished reading pride and prejudice which had me HOOOKED from the beginning,". the VADER output was [neg: 0.19, pos: 0.14, neu: 0.66, compound: -0.23],, indicating a negative sentiment, although the goal sentiment is positive. In this sentence, the word "prejudice" (-2.3) holds a strong negative sentiment score in the lexicon. the word "pride" holds a positive rating (1.4), with the rest of the sentence being mostly neutral. Overall this sentence, according to VADER, has a slight negative sentiment score, as the negative sentiment of prejudice dominates the slightly less positive sentiment score of pride it is evident why VADER classified this sentence as negative, although it is clear to human annotators that this is an overall positive statement to a book,
The following sentence: "The film Everything Everywhere All At Once follows Evelyn Wang, a woman drowning under the stress of her family's failing laundromat." was classified as negative although the gold label is neutral. The VADER output was [neg: 0.25, pos: 0.0, neu: 0.74, compound: -0.73]. This is due to the negative words "stress", "failing" and "drown" which all hold a strong negative sentiment contributing to the high negative compound score (-0.73) assigned by VADER to this. Although this sentence objectively lists the topics in this movie which should be considered neutral, VADER struggles understanding the context and focuses on individual words therefore it is plausible why it applied a negative sentiment to this sentence.

CONCLUSION

The RoBERTa model performs well on the higher amount of training dataset. After training the model a couple of times, it seems that the model learns to predict the labels more accurate with less false positive values. However, when testing the model with a bigger set of instances, the RoBERTA model still predicts the labels inaccurately as mentioned above in the RoBERTa model section.
BERT on the other hand, performs generally worse than RoBERTa. The BERT model obtains an accuracy of 0.6 while RoBERTa manages to achieve 0.8. However, BERT has a higher recall for the Sports and Book categories, which means that BERT managed to do a better job at retrieving the instances for these topics. Since the number of instances in the test set is only 10, it could also just be attributed to the random chance factor. That being said, RoBERTa performing better than BERT is to be expected overall. RoBERTa was pre-trained on a lot more data than BERT and should consequently be able to pick up on patterns and context better than BERT.
Lastly, the test set having so few instances makes it more difficult to properly assess the performance and quality of the models. For future work it might help if we use a test set with more instances so we can properly evaluate the models. In addition to this, we could try a variety of different training parameter combinations to check if these affect the performance of the models at all. However, due to time and hardware constraints we were unable to test out these different parameters.
Furthermore, for the sentiment analysis, VADER's lexicon-based approach might struggle with understanding the context of a sentence fully. It might misinterpret nuanced language where sentiment is expressed indirectly. In addition, using a different ML approach (for example Naive Bayes) for this specific task of sentiment analysis could prove to be a better choice and perform better.

REFERENCES

[1] Lab3-assignment (https://t-redactyl.io/blog/2017/04/using-vader-to-handle-sentiment-analysis-with-social-media-text.html)

[2] Sun, Chi, Xipeng Qiu, Yige Xu, and Xuanjing Huang. "How to fine-tune bert for text classification?." In China National Conference on Chinese Computational Linguistics, pp. 194-206. Springer, Cham, 2019

[3] https://www.kaggle.com/datasets/gauthamp10/google-playstore-apps

[4] Lab6-assignment-topic-classification (https://github.com/cltl/ba-text-mining/tree/master/lab_sessions/lab6)

[5]RoBERTa-huggingface (https://huggingface.co/docs/transformers/en/model_doc/roberta)

[6] Lab4-assignment-ner (https://github.com/cltl/ba-text-mining/blob/master/lab_sessions/lab4/Lab4a.1-NERC-introduction.ipynb)

[7] Lecture 3 - Natural Language Processing & Machine learning

DIVISION OF WORK

Thor Tiefenthal (2696111): 1) NERC coding, 2) NERC analysis, 3) NERC poster preparation

Andreas Styliaras (2698459): 1) VADER coding, 2) Sentiment analysis, 3) VADER poster preparation

Britt Westerhof (2700722): 1) BERT coding, 2) BERT topic analysis & conclusion comparision to RoBERTa, 3) BERT poster preparation

Paworapas Kakhai(2709158): 1) RoBERTa coding, 2) RoBERTa model analysis & conclusion, 3) RoBERTa poster preparation