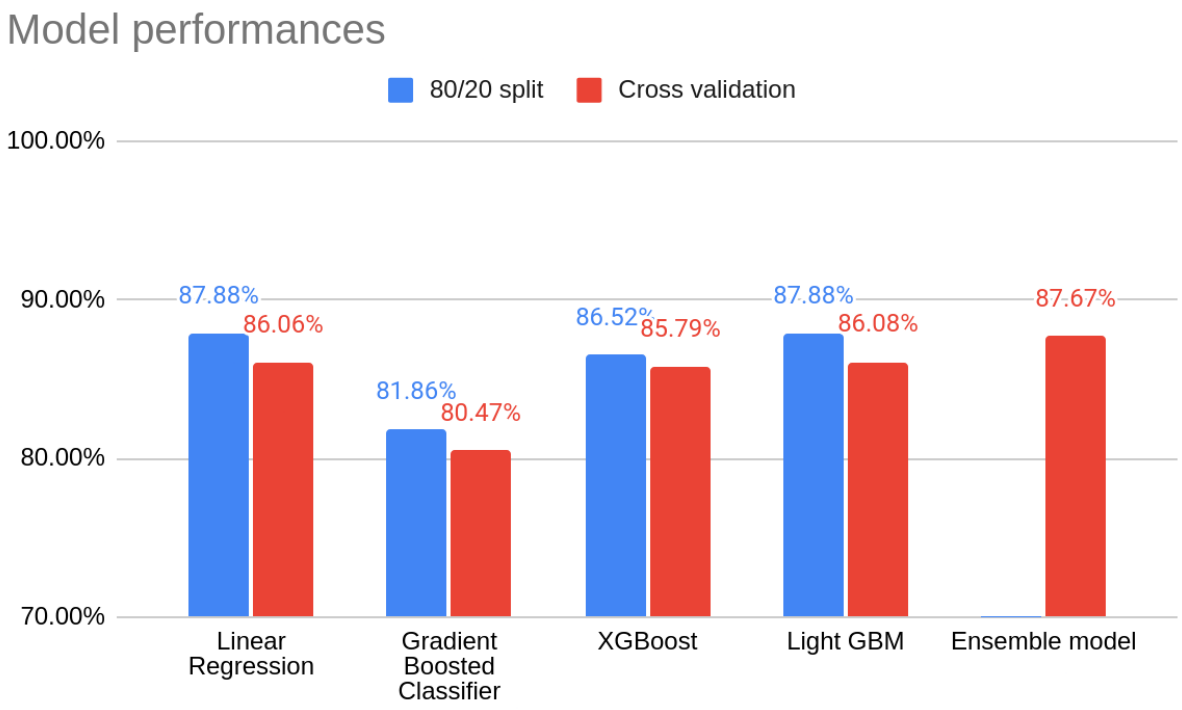
Approach to this problem

For this problem, I approached it as a word vectorization classification problem. To transform the dataset I used a term frequency vectorizer https://en.wikipedia.org/wiki/Tf%E2%80%93idf to transform the linguistic data into a numeric representation based on which words occurred in the text. To this end, I evaluated the performance of statistical machine learning models, namely Linear Regression, Gradient Boosted Classifier, XGBoost and Light GBM. Random Forest, support vector machines and Extra Trees were also considered, however, due to processing power limitations they were excluded.

To evaluated performance, I started off with an 80/20 train test split to get an initial estimate of the performance and then moved on to 5 fold cross-validation https://en.wikipedia.org/wiki/Cross-validation_(statistics) to evaluate the performance of each of the machine learning models on every part of the training data.

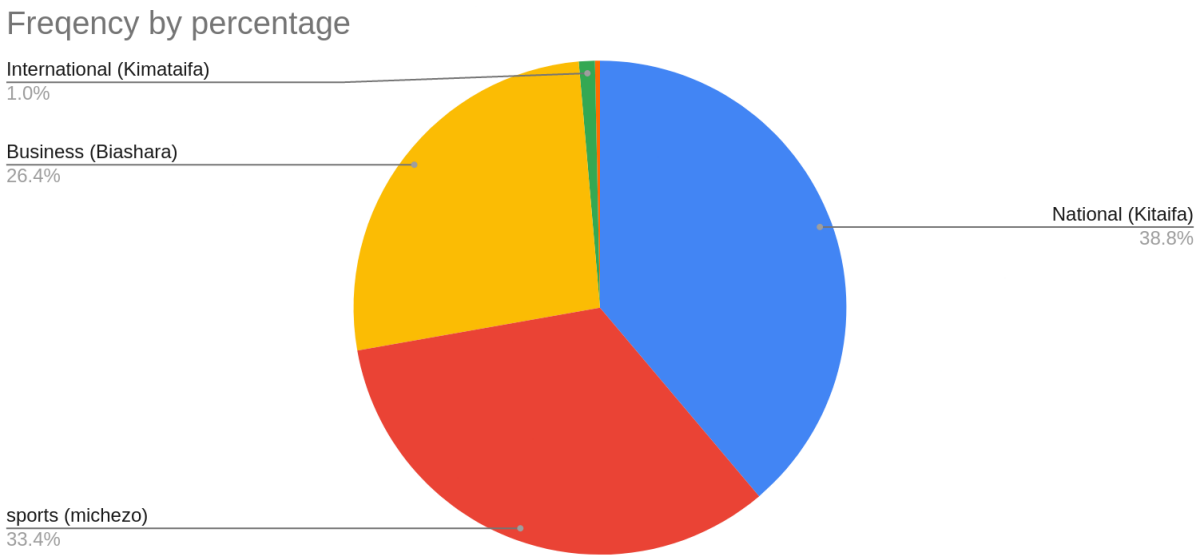
Using the prediction probabilities used in the evaluation of the models when applying 5 fold cross-validation we then combined the models into a single ensemble model with a stacking estimator implemented via random forest.

After comparing the performance of each of the models it was shown that the ensemble model marginally outperformed the other models, which is why it was selected as the final algorithm. Then to get the performance, the entire training set was fed into each of the machine learning models and then the ensemble model predicted the label of the test set.



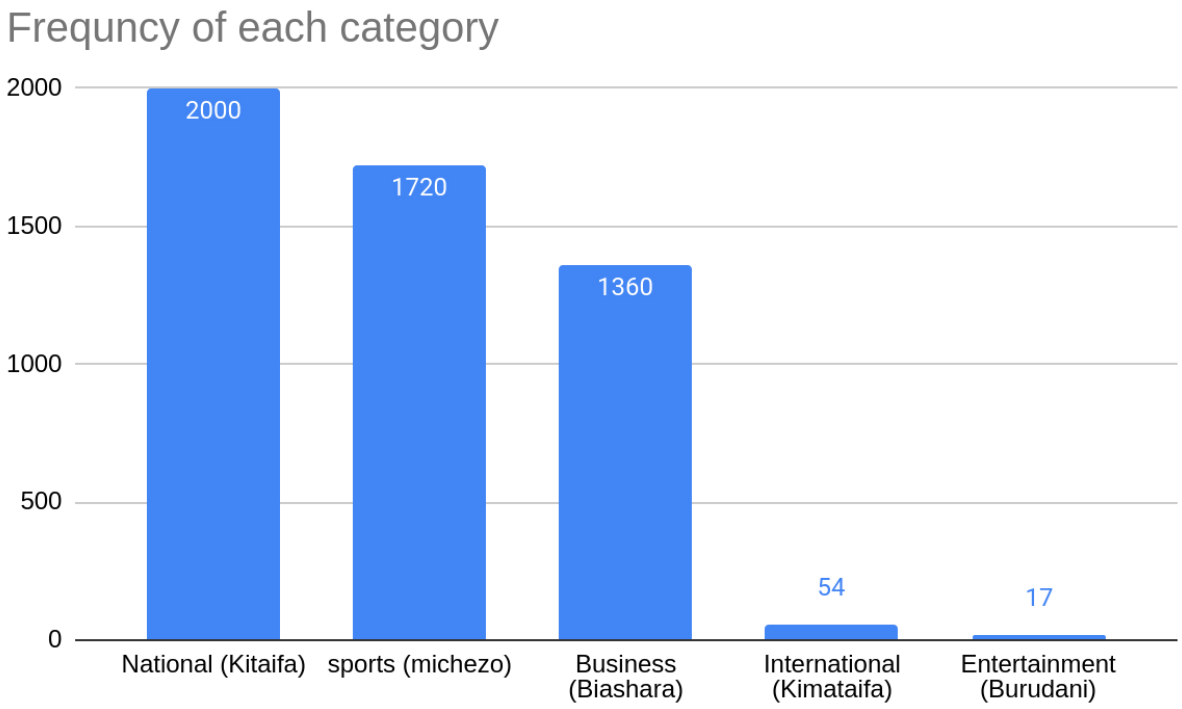
Recommendations

It would make more sense to only keep 3 categories to show the readers namely Kitaifa, Michezo and Biashara. It is my understanding that users typically would want to sort the information based on how relevant the topics are. This would help if the categories roughly had the same amount of categories. In this case, 99% of the dataset fell into one of the 3 categories and as shown in the figure below, the 3 categories are of similar size. If the data on https://www.habarileo.co.tz/ is representative of the data the client wants to use, it would make more sense to ignore the small categories.



Relevant and interesting visualization and describe it to the client

Below is a figure of how frequently each category was used in the dataset. As you can see, 3 of the categories are used considerably more frequently than the others, to the point where it should be questioned whether the last two categories should be included. As you can see there are between 1,300 and 2,000 samples of each of the 3 main categories, while there are less than 100 of the two smaller categories. So we have to ask the question is it worth retaining these lower categories if the articles will be so few in comparison to the others.



Future work

- I do believe building convolutional or transformer models could have yielded better results. With either of these approaches, it would have been possible to train on other datasets that are similar. The data would then range from either sister languages or data with different categories since the internal model would gain an understanding of the structure of similar languages. I am fairly confident that given enough time and resources that would given the best results (and depending on the resources), the model would probably be around 95% accurate.
- It would have been interesting to see the performance of other autoML tools and compare the results with other systems (and to actually just run all the initial models to see how they perform and the performance increase in the ensemble model)
- I would have looked into resources available for the language, such as parts of speech tagging.
- I would have spent a bit more time writing up and cleaned the code better.
- It might have been prudent to play with feature selection and hyperparameter tuning