ARTISAN AI - Assignment

The data at play is a .pkl file which contains a list of dictionaries which are not uniform in nature. The goal is to analyse this data and to make a ML model and present the analysis on the same.

This project report summarise the overall approach in 4 different parts:

a) Introductory Analysis

- The idea was to convert the data into a tabular data frame structure. Once we achieve that, we get a dataset which has 150 rows across 5 columns.
- The data is not balanced in nature and seems to be of campaign marketing analytics /raw data.
- As this is a campaign/email marketing data,we see 2 main text columns of body and subject
- We find the following in the EDA:
  - Combined processing of columns post data cleaning
  - Word clouds and pandas profiling
  - Unbalanced dataset using value count
  - N gram and Most common words for emails opened vs emails not opened

There are other things that can be looked at like using spacy instead of NLTK for better stemming, generating dummy data for balancing the classes, trying ngrams with greater than 4 length etc.

b) Machine Learning Model Analysis

- As this is a text related classification problem, there are multiple ways to solve it. We can apply a simple classification model to predict Opened vs Not opened or Link Clicked vs Not clicked etc.
- But I have solved this problem using the MultiOutput Classifier for MultiClass Classification using XgBoost and RF.
- In this method, we will be predicting both the results simultaneously i.e whether the email was opened and if yes was the link clicked etc.
- One drawback of this method is that, if email opened is false then the meeting link clicked should automatically become false
- I have tried the simple models using TF - IDF vectorization looking at the time constraint

But other things which can be implemented are cross validation, better neural networks, different classification approaches like MultiLevel Classification. Also parameter tuning can be attempted so as to get the best model.

c) Results

The idea was to evaluate the model based on F1 score. Again there were various metrics like AUC- ROC or Life Curve analysis to analyse the predictions, but keeping it simple, F1 score was chosen.

We get a decent F1 of 0.8 on email opened and 0.53 on link_clicked. I think the latter can be improved by some of the techniques mentioned above.

The code for the same is attached in the notebook within this repo