FANTASY FOOTBALL AI - Challenging Betting Models Using Feature Engineering and Sentiment Analysis

ACM40960 - Projects in Maths Modelling | Deepankar Vyas | 23200527

Motivation

- Develop a model capable of predicting outcome of football matches
- Challenging pre-existing betting models by training our model on **engineered features** (based on domain knowledge) and **sentiment analysis** of pre-match reports
- Combine the earlier works done in this field either exclusively using statistical features or using exclusively text analytics and conclude whether a combination of features provide better results

Dataset Preparation

Prepared the dataset using Web - Scraping . Base Features and Betting Odds were taken from Football-Data.co.uk [6] , Team Ratings were taken from FIFA Index [5] and Pre - Match Reports were scraped from - WhoScored.com [8] . Everything was collated to prepare a Master Dataset.

Methodology

- Using **R** as the programming language, the model is implemented. The entire model is simulated in **RStudio**

Feature Engineering

- **HGKPP , AGKPP** - Home and Away teams' past 5 matches Goals.
- **HSTKPP , ASTKPP** - Home and Away teams' past 5 matches Shots on Targets.
- **HCKPP , ACKPP** - Home and Away teams' past 5 matches Corners.

$$\mu_j^i = \left(\sum_{p=j-k}^{j-1} \mu_p^i\right)/k$$

$$\mu^i \in \{\text{Corners, Shots on Target, Goals}\}, k = 5$$

- **HForm , AForm** - Home and Away teams' Form

When Team A beats Team B

$$F_j^A = F_{(j-1)}^A + \gamma F_{(j-1)}^B$$
$$F_j^B = F_{(j-1)}^B - \gamma F_{(j-1)}^B$$

In case of a Draw

$$F_j^A = F_{(j-1)}^A - \gamma(F_{(j-1)}^A - F_{(j-1)}^B)$$
$$F_j^B = F_{(j-1)}^B - \gamma(F_{(j-1)}^B - F_{(j-1)}^A)$$

$$\gamma = 0.33$$

- **HSt , ASt, HStWeighted, AStWeighted** - Home and Away teams' Streak and Weighted Streak of the past 5 matches.

$$Streak(\delta_j) = \left(\sum_{p=j-k}^{j-1} \text{resp}_p\right)/3k$$

$$\text{resp}_p \in \{0, 1, 3\}$$

$$Weighted\_Streak(\omega_j) = \sum_{p=j-k}^{j-1} \frac{2(p-(j-k-1)\text{resp}_p)}{3k(k+1)}$$

- **HTGD , ATGD**- Home and Away teams' past 5 matches Goal Difference

$$GD_k = \sum_{j=1}^{k-1} \text{GS}_j - \sum_{j=1}^{k-1} \text{GC}_j$$

$$\text{GS} = \text{Goals Scored}, \text{GC} = \text{Goals Conceded}$$

- Base Features (Goals, Shots on Targets and Corners) are used to engineer the above features for the Home and Away teams [1,3].
- Other than the engineered features, we also considered the **Team Ratings** of each team's Attack, Midfield, Defense and also Overall Rating.
- The **Day of the Week** when the match was played and the **Season** were included as Psychological factors [7].
- **Home_Score , Away_Score** - Sentiment Analysis on a match's pre-match report to generate Home and Away team's sentiment score, referenced in Figure 1. [2]

Figure 1 : Wordcloud

EDA and Data-Preprocessing

- Data was cleaned by removing the NA values and unwanted columns
- Most of the engineered features violated normality assumption, therefore, new features were created which were the difference between the values of Home and Away features , called **Differential Features.** Same is illustrated in Figure 2 and Figure 3.

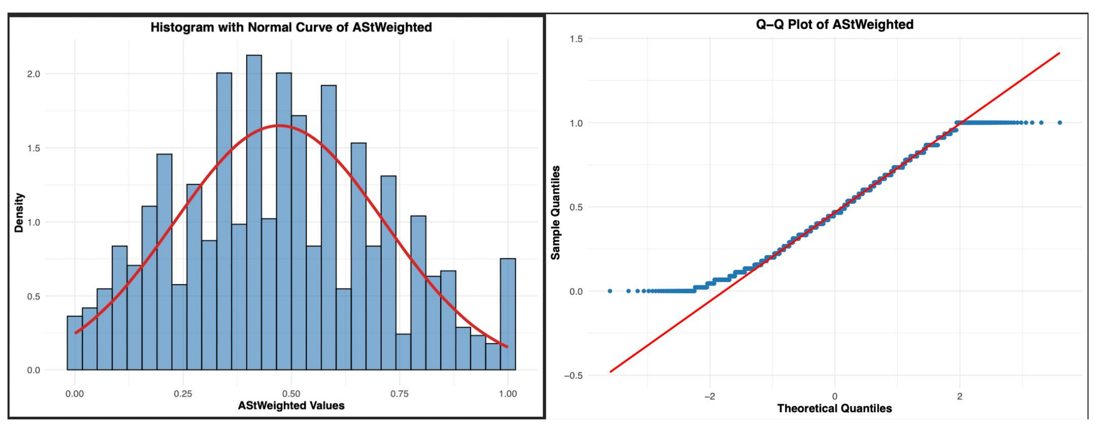

Figure 2 : Home and Away Features

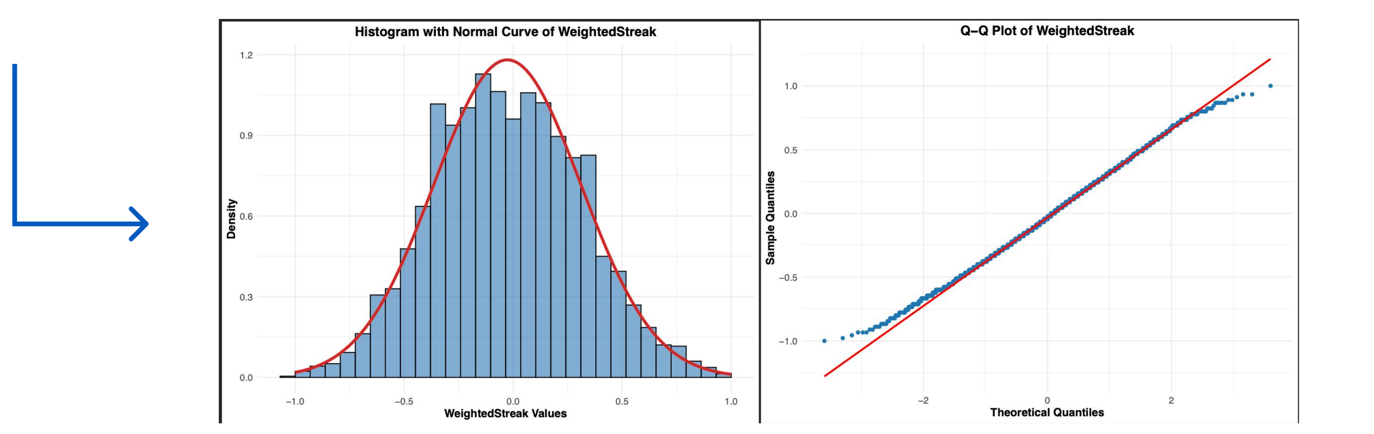

Figure 3 : Differential Features

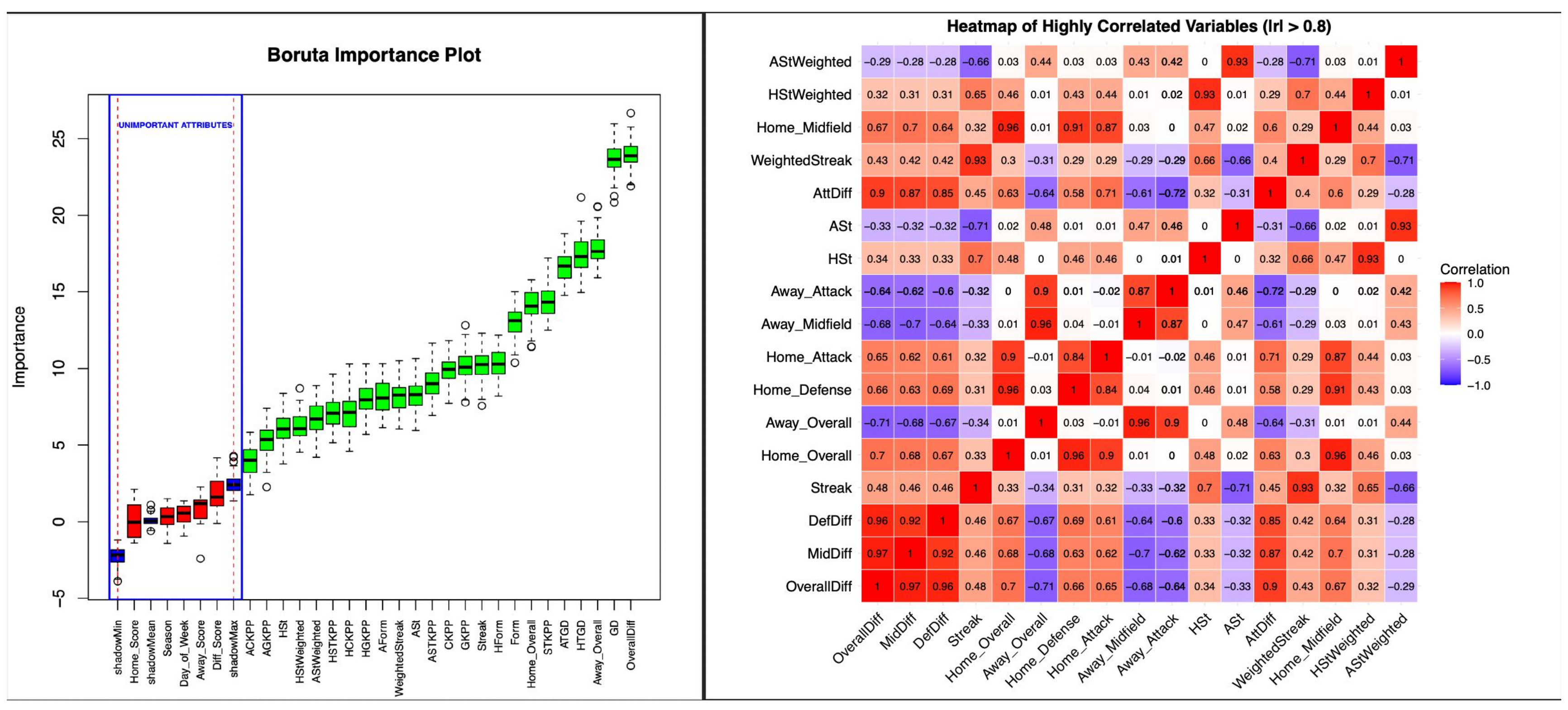

Figure 4 : Correlated and Unimportant Features

- As shown in Figure 4 depicting the **Correlation Matrix** and **Boruta Feature Selection**, it became clear that the variables are highly correlated to their differential features and features such as - **Day_Of_Week, Season** are not important. These features were removed and our dataset was divided into 4 datasets :-

(a) **Class A** :- Home and Away features of the teams
(b) **Class A NLP** :- Home and Away features of the teams with Sentiment Scores
(c) **Class B** :- Differential features of the teams
(d) **Class B NLP** :- Differential features of the teams with Sentiment Scores

Model Training and Tuning

5 models were trained for each dataset, with their respective hyperparameter grid, shown in Figure 5. Tuning was done using 5-fold cross validation with 3 repetitions. **RPS (Ranked Probability Score)**[4] was used to decide the optimal hyperparameter.

$$\text{RPS} = \frac{1}{r-1}\sum_{i=1}^{r-1}\left(\sum_{j=1}^{i}(p_j - e_j)\right)^2$$

- **Logistic Regression**
- **SVM with Polynomial Kernel**
- **SVM with Radial Gaussian Basis Kernel**
- **Random Forest**
- **Neural Networks** :- 3 layered network with 128, 64 and 32 units respectively. Early stopping with patience 20, 100 epochs and a batch_size of 32 were used. Dropout rate and L2 regularization hyper-parameters were tuned.

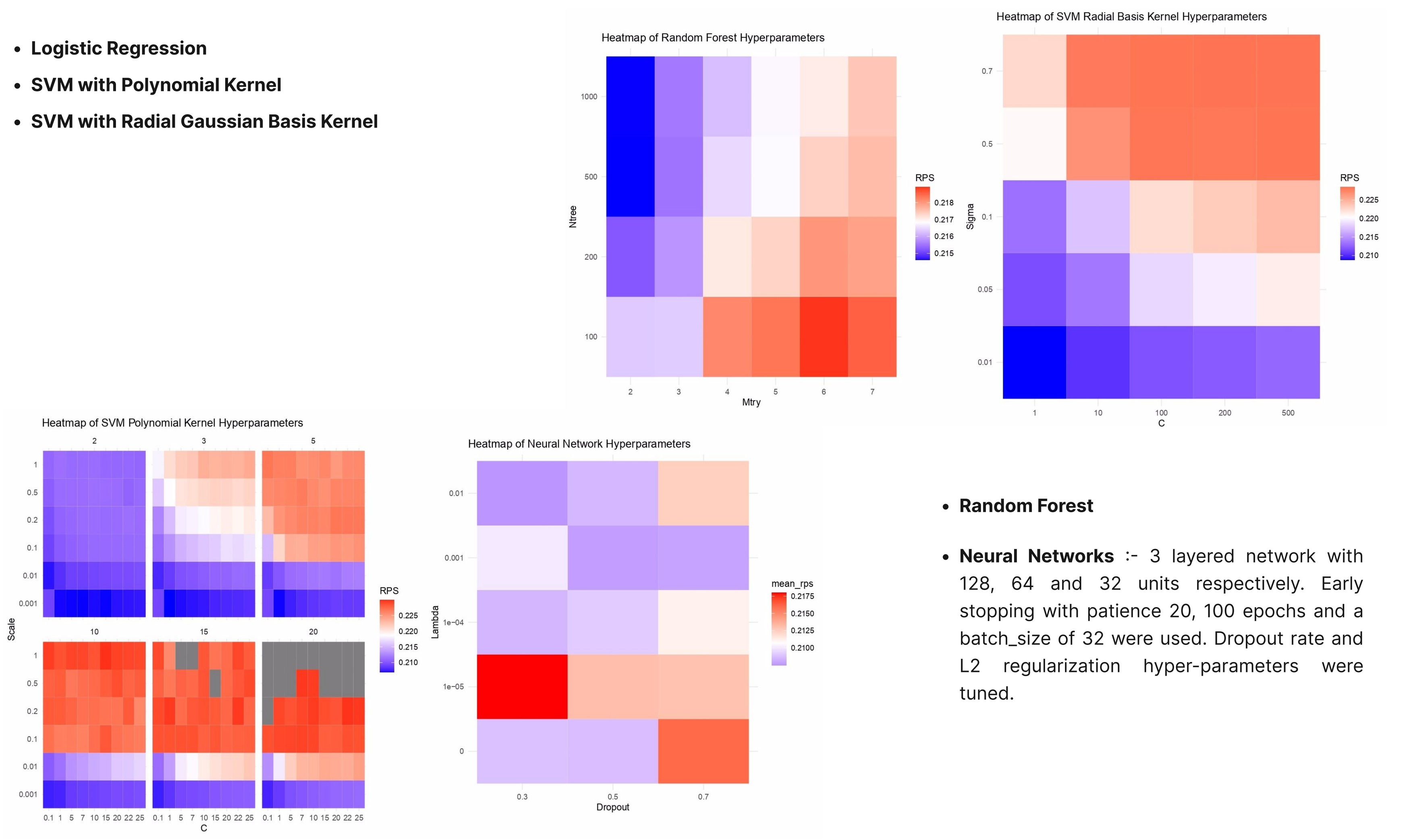

Figure 5 : Heatmap of Models' Hyperparameters

Model Evaluation and Generalized Predictive Performance

Along with **Mean F1** and **F1 of Draw class** (which was the hardest class to predict), we used **RPS** too to finalize the model which performed the best. A summary is shown in Figure 6.

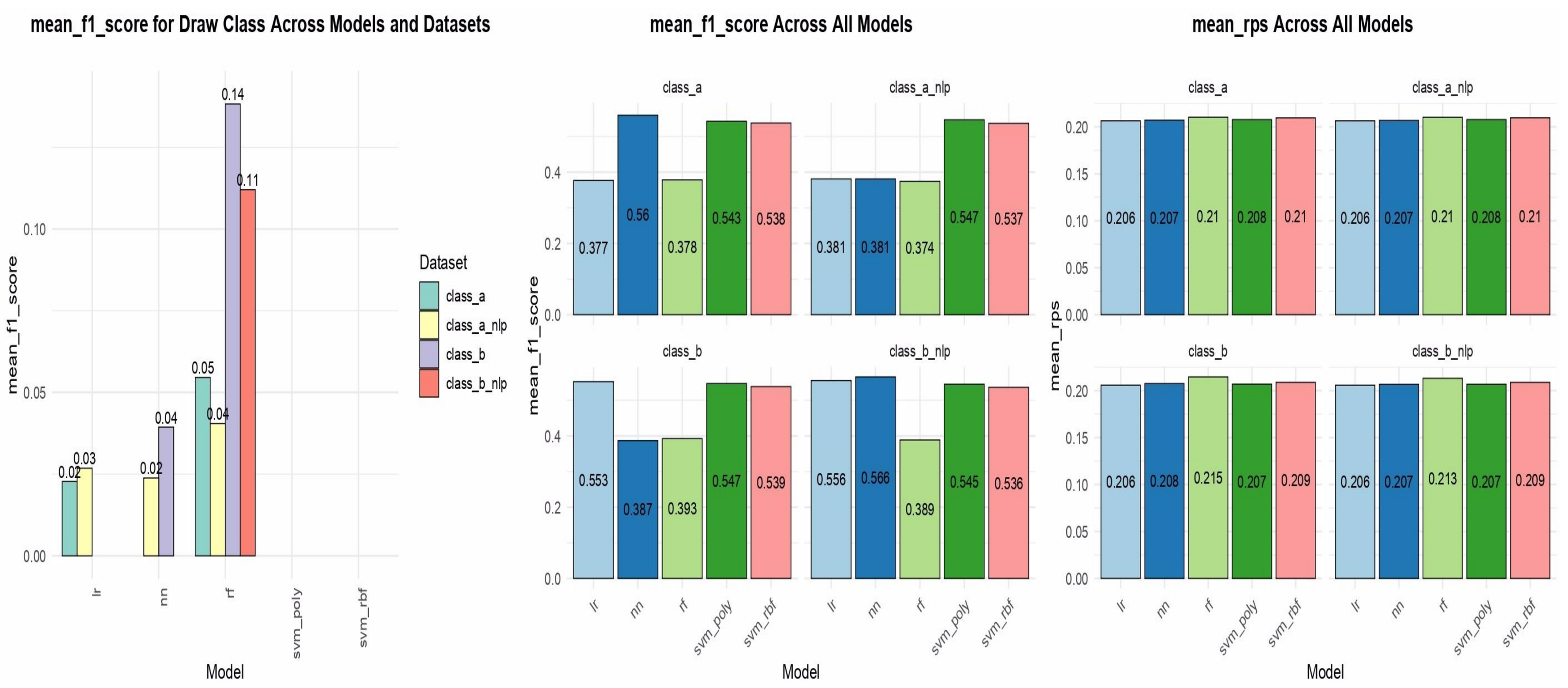

Figure 6 : Model Comparison using F1 Score and RPS

Model	Dataset	F1 Score	Balanced Accuracy	RPS Score	F1_Draw
Random Forest	Class B	0.389	0.566	0.213	0.112
Random Forest	Class B NLP	0.392	0.564	0.215	0.138

Table 1: Model Rankings

The best model selection was based on a heuristic involving **Mean_F1 Score** and **RPS.** From the table, the best performing model was **Random Forest** trained on the dataset **Class B NLP** (differential features with sentiment scores)

The selected model was used to predict the unseen data , and gave the following results, with an **RPS** of **0.2155** and an overall **Balanced Accuracy** of around **57**%, as can be seen in Figure 7.

Class	Sensitivity	Specificity	F1 Score	Precision
Class A	0.426	0.795	0.450	0.476
Class D	0.040	0.951	0.067	0.207
Class H	0.783	0.404	0.625	0.520

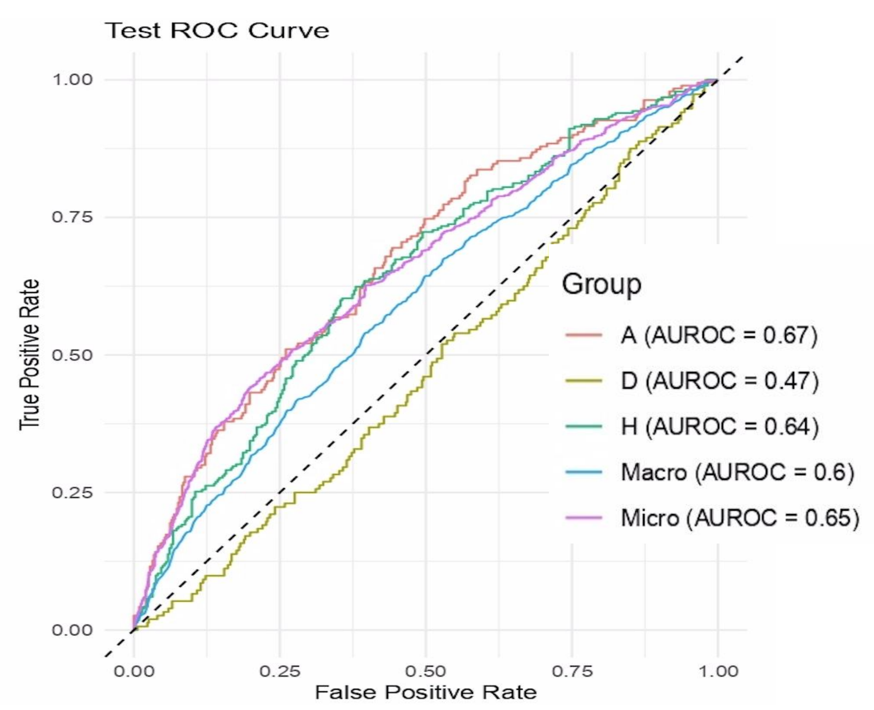

Figure 7 : Generalized Predictive Performance

Did We Succeed?

- We tested the performance of our model against the benchmark setting - **BET365 Odds.**
- The implied probabilities (inverse odds) were used after basic normalization to remove unfairness. RPS Score was then calculated using these probabilities **[9].**
- Even though our model did not beat the benchmark setting, it performed remarkably well and fared slightly better than the model used by the research paper which was used as this project's baseline, shown in Figure 8. Also, the set of features having sentiment scores performed the best, indicating that a better sentiment analysis using LLMs could given even better results.

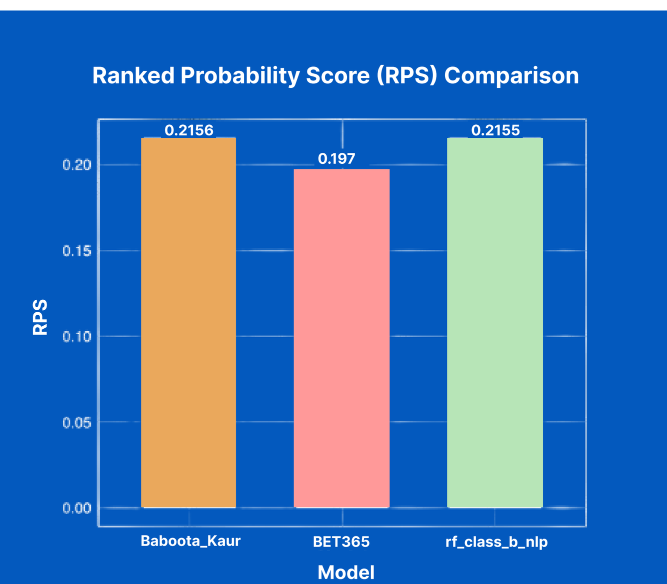

Figure 8 : Rps Score Comparison

Future Scope

- To tune the number of matches to be considered for calculating the engineered features, instead of the static value 5.
- Use better techniques for sentiment analysis.
- Integrating the model with a GUI which will enhance the ease of use.

References

[1] Rahul Baboota and Harleen Kaur. Predictive analysis and modelling football results using machine learning approach for english premier league. *International Journal of Forecasting*, 35(2):741–755, 2019.
[2] Ryan Beal, Stuart E. Middleton, Timothy J. Norman, and Sarvapali D. Ramchurn. Combining machine learning and human experts to predict match outcomes in football: A baseline model. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(17):15447–15451, May 2021
[3] Bing Choi, Lee-Kien Foo, and Sook-Ling Chua. Predicting football match outcomes with machine learning approaches. *MENDEL*, 29:229–236, 12 2023.
[4] Anthony Constantinou and Norman Fenton. Solving the problem of inadequate scoring rules for assessing probabilistic football forecast models. *Journal of Quantitative Analysis in Sports*, 8, 01 2012
[5] FIFA Index. FIFA Index, 2024. [Accessed: 2024-06-10]
[6] Football-Data.co.uk. Football-Data.co.uk, 2024. [Accessed: 2024-06-10].
[7] Farzin Owramipur, Parinaz Eskandarian, and Faezeh Mozneb. Football result prediction with bayesian network in spanish league-barcelona
team. *International Journal of Computer Theory and Engineering*, pages 812–815, 01 2013.
[8] WhoScored.com. WhoScored.com, 2024. [Accessed: 2024-06-10]
[9] Erik ˇStrumbelj. On determining probability forecasts from betting odds. International Journal of Forecasting, 30(4):934–943, 2014

Scan me!

Scan to view GitHub Repository