Electronics and Computer Science Faculty of Physical Sciences and Engineering
University of Southampton

Author Name: Ben Hipwell

May 2023

Predicting Fantasy Premier League Points with Machine Learning: The Effect of Social Media Sentiment

Project Supervisor: Luis-Daniel Ibáñez

Second Examiner: Eric Rogers

A project report submitted for the award of
MEng Computer Science

1 Abstract

In this project, I investigate the impact of online opinions on machine learning systems to predict real world football performances. This is in the form of 'fantasy football' points within the popular Fantasy Premier League (FPL) game, where individual players gain points depending on aspects of their performance per match in the Premier League. I started off creating a baseline machine learning model to forecast a players expected points for each FPL position (goalkeeper, defender, midfielder & forward) using standard football statistics. I have then taken data from Twitter, Reddit and the FPL playerbase to create new features that can be combined with those used to train the baseline models. I have used Natural Language Processing to extract information that is useful for training a model, in this case the sentiment and frequency of the social media data collected. As a result of this, I could then examine the change in error of the predictions, using RMSE, when adding various combinations of the new features, of which represent the fans' opinions. This concluded with new models that saw a decrease in the RMSE & MAE of between 4 and 10% depending on the position model. This lead to the midfielder model, trained on the baseline features plus all of the new features discussed, becoming the most accurate model with an average error of 1.58 points.

Statement of Originality

- I have read and understood the ECS Academic Integrity information and the University's Academic Integrity Guidance for Students.
- I am aware that failure to act in accordance with the Regulations Governing Academic Integrity may lead to the imposition of penalties which, for the most serious cases, may include termination of programme.
- I consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to verify whether my work contains plagiarised material, and for quality assurance purposes.

You must change the statements in the boxes if you do not agree with them.

We expect you to acknowledge all sources of information (e.g. ideas, algorithms, data) using citations. You must also put quotation marks around any sections of text that you have copied without paraphrasing. If any figures or tables have been taken or modified from another source, you must explain this in the caption and cite the original source.

I have acknowledged all sources, and identified any content taken from elsewhere.

If you have used any code (e.g. open-source code), reference designs, or similar resources that have been produced by anyone else, you must list them in the box below. In the report, you must explain what was used and how it relates to the work you have done.

I have not used any resources produced by anyone else, other than:
NLTK: https://www.nltk.org/, Pandas: https://pandas.pydata.org/, XGBoost: https://xgboost.readthedocs.io/en/stable/, Numpy: https://numpy.org/, Scikit-learn: https://scikit-learn.org/stable/, Matplotlib: https://matplotlib.org/, PRAW: https://praw.readthedocs.io/en/stable/, Transformers: https://huggingface.co/docs/transformers/index, PyTorch: https://pytorch.org/

You can consult with module teaching staff/demonstrators, but you should not show anyone else your work (this includes uploading your work to publicly-accessible repositories e.g. Github, unless expressly permitted by the module leader), or help them to do theirs. For individual assignments, we expect you to work on your own. For group assignments, we expect that you work only with your allocated group. You must get permission in writing from the module teaching staff before you seek outside assistance, e.g. a proofreading service, and declare it here.

I did all the work myself, or with my allocated group, and have not helped anyone else.

We expect that you have not fabricated, modified or distorted any data, evidence, references, experimental results, or other material used or presented in the report. You must clearly describe your experiments and how the results were obtained, and include all data, source code and/or designs (either in the report, or submitted as a separate file) so that your results could be reproduced.

The material in the report is genuine, and I have included all my data/code/designs.

We expect that you have not previously submitted any part of this work for another assessment. You must get permission in writing from the module teaching staff before re-using any of your previously submitted work for this assessment.

I have not submitted any part of this work for another assessment.

If your work involved research/studies (including surveys) on human participants, their cells or data, or on animals, you must have been granted ethical approval before the work was carried out, and any experiments must have followed these requirements. You must give details of this in the report, and list the ethical approval reference number(s) in the box below.

My work did not involve human participants, their cells or data, or animals.

ECS Statement of Originality Template, updated August 2018, Alex Weddell aiofficer@ecs.soton.ac.uk

2 Acknowledgments

I would like to thank my project supervisor Luis-Daniel Ibáñez for helping me throughout the course of this project by answering all of my questions thoroughly and providing me with excellent guidance, making this learning experience engaging and enjoyable.

Contents

3 Introduction

3.1 Problem

The current popular models used for forecasting football outcomes do not consider social media opinions. As football is such an opinionated sport, with fans around the world expressing their own opinion, this data could improve the current statistic driven prediction systems. For example, not all aspects of a football player's performance during a game can be converted into statistics, e.g. correct player positioning and seeming technical ability, and therefore these could prove important metrics when it comes to the addition of social media data.

3.2 Goal

To use football player statistics, with the addition of social media data and other sources of fans' opinions, to train a machine learning system with the aim to predict a player's Fantasy Football score for a given 'gameweek'. The goal is to refine this model so that it can outperform other data driven predictions that do not use social media opinions. This may entail:

- Data collection from popular social media sites.
- The Natural Language Processing of social media data.
- Data collection from the Fantasy Premier League dataset.
- The data engineering of all collected data, formatted so that it is ready for training.
- Feature selection
- Creation and training of ML models to predict a player's performance.

3.3 Scope

To use social media data from platforms such as Twitter and Reddit, and fans' opinions via FPL metrics, to assess whether they can improve football predictions. The social media data that I use is chosen because of its usefulness within my model and how accessible it is to obtain and use effectively. I will then use this data to create tailored models for each football position within the Fantasy Premier League game. The data being used from the FPL dataset and across the social media sources will cover the 2021/22 and 2022/23 (up to gameweek 28) Premier League seasons.

4 Literature Review

4.1 Background

Fantasy Premier League is a ‘fantasy football’ game in which a player, often referred to as a ’manager’, picks a team of real Premier League players with the goal of selecting who they think will perform well in the upcoming matches. Each position (Goalkeeper, Defender, Midfielder and Attacker) of player scores points based on different metrics, e.g. goalkeepers score points for saves, however some are shared, e.g. minutes played, assists and bonus points. Additionally, each player has a value, which can change throughout the course of the season, where a player’s total team cannot exceed the budget of £100 million.

The Bonus Points System (BPS) rewards players for smaller achievements and statistics that reflect their performance during the game. These points are given for positive impacts, e.g. key passes & assists and are taken away for negative impacts, e.g. own goal & yellow card. The players with the top three total bonus points scores by the end of the game receive 3,2,1 extra points respectively.

The term ‘gameweek’ represents a series of matches in which every team will play a game for which your team scores points for. Before each gameweek, players are given the opportunity to make transfers and substitutions in time for the next games.

Additionally, in FPL each player is given an Influence Creativity Threat (ICT) Index score which ranks them compared to other players in the same position on these three aspects of their game. These are calculated by FPL for each player and ranks them for each of the three indexes and an overall combined index. (Premier League 2022)

4.2 Current Point Prediction Research

A very popular site used to analyse which players to pick is fantasyfootballscout.co.uk, who use current and historic data in their algorithms to predict player points for the next 6 ‘gameweeks’. The key statistics that they claim to use include goals, assists, clean sheets, minutes played, bonus points, yellow cards and previous points. (Fantasy Football Scout 2022)

’xG’ is a term representing ’expected goals’ that can be calculated for an individual player or team. A popular website ’fbref.com’ is used thoroughly by serious FPL players, especially by the FPL community on Reddit. They calculate this by comparing statistics such as the location, body part, type of pass and type of attack of a shot to thousands of similar shots and determining the chances of it converting into a goal. The total ’xG’ value can give a good representation of how well a team or player is performing. (Sports Reference 2022)

4.3 Similar Project Research

4.3.1 FPL Data Science Report 1: Nisumaa 2019

In this project, they were able to create two FPL teams that outscored the player base average and even experts two weeks in a row. They used the WEKA framework to perform machine learning, which is a more simplistic platform, however towards the end of the report they determined it was a limiting factor to the overall project and that someone with more ML knowledge could get more out of using Python libraries. Additionally, they broke down the model attributes into ‘player’, ‘team’ and ‘FPL’ (e.g. ICT, BPS. . . etc), which in total came to 38 various attributes spread across these categories.

For their results, Naïve Bayes produced a 63.6% accuracy score, interestingly 73.4% in low scoring players (1-3 points). ANN managed to get an accuracy of 96.3% and Random Forest 96.6%, which seemed to make heavy use of the ICT attribute. They discovered accuracy while using the FPL attributes was significantly higher across all ML algorithms than not using them.

To improve they would use Bayesian networks and would take the features of the next fixture and form into account. It was also decided that using more data to train the ANN and RF would have been beneficial and that separate models for each position would improve the model. (Nisumaa 2019)

4.3.2 FPL Data Science Report 2: Walters 2021

Firstly, for this project they created an XGBoost regression model for each player position, trained using different sets of features. They used the same dataset they I am aiming to use, using data from the 18/19 to 20/21 seasons, using the sets ‘cleaned_players’, ‘understat_player’ and ‘merged_gw’. When merging the collected data together, they changed ‘first_name’ and ‘second_name’ attributes to a combined ‘player_name’ to help with the merge, and to remove any special Unicode characters as they were causing errors. Using XGBoost via Scikit-learn, they compared five gameweeks against the average player score within FPL. Also used rolling dataframes to manage input of consecutive gameweeks, to take form into account, via the Pandas ‘method.rolling()’ method and np.sum to sum the frame. Also used the ‘gain’ values to see which features were not useful within the model and removed them. Used ‘.isnull().sum()’ to make sure no null or NaN values in the dataset.

Using RMSE to measure the distance between the expected and actual results, they found the Midfielder position to have the most data, especially the most BPS data, and therefore became the most accurately predicted position. Conversely, Goalkeepers had the least data available to make good predictions, and therefore resulted in the least accurately predicted position. Adding the xG and xA features reduced error but not by that much.

To improve, they would increase the number of seasons they train the models on, past the 18/19 season. They believe that additional statistics such as dribbles completed, tackles won. . . etc would have improved the accuracy. Additionally, they would compare using other machine learning algorithms with XGBoost to see whether improved results could be obtained. (Walters 2021)

4.3.3 NLP Report: Melton et al. 2022

This report focuses on the use of Reddit and Twitter data to determine sentiment analysis on the topic of COVID-19 vaccines on social media. The BERT algorithm was used, which is an AI based NLP algorithm developed at Google that specialises in text classification and can be fine tuned using custom data so that it can adapt to a certain text style.

In this report, they used ‘DistilRoBERTa’ for sentiment analysis, which is a more robust, optimized BERT algorithm that doesn’t include the next sentence prediction feature and instead uses ‘dynamic token masking’ during training. They fine tuned the model using the ‘Hugging Face Trainer class’ via Pytorch, and with the hyper-parameters they used, were able to reach an accuracy of 0.9592. The social media data was then processed through the ‘Hugging Face’ pipeline to calculate sentiment of each tweet and comment. The output is binary, being either ‘positive’ or ‘negative’ and a probabilistic confidence score is also given, ranging from 0 to 1.

Their results displayed that 54.8% of the 9.5 million tweets were in fact more negative than positive, whereas 37.7% of the 67,962 Reddit comments where more negative. There were similar patterns in the changes in sentiment of each month, however Reddit was generally more positive about the COVID-19 vaccines. They predict this may be down the character limit on Twitter, pushing people to post a lot of short, opinionated content to quickly spread information whereas the Reddit comments were likely to be longer and more detailed opinions. Additionally, there may also be the challenge of facing ‘bots’ on Twitter posting lots of misinformation to alter the overall narrative. (Melton et al. 2022)

4.3.4 Conclusions

Firstly, the use of the XGBoost ML algorithm proved successful on the dataset that I will be using, along with the effective use of Python and the Sci-kit learn library, and therefore it seems logical to use it for this project. The progression of the model from Nisumaa’s paper (Nisumaa 2019) to Walter’s (Walters 2021), where separate models were created for each player position, is something that I will implement in this project. Additionally, the use of rolling dataframes seems to be vital for this type of data, with the form of the players being very important, and hence will implement. Lastly, reviewing both of these papers has given me a better insight into which features I should select for each model, especially the impact of the BPS values. On the other hand, my conclusions from the NLP side of this literature review displays the popularity of the BERT algorithm. This is something I will consider using if appropriate fine-tuning data is accessible, and therefore within scope.

4.4 Additional Research

4.4.1 Offline vs Online Learning

Within Offline learning, the complete dataset is available at once, where it is trained from the beginning until the final state, ready for use. On the other hand, Online learning is where the model is trained whenever new training data is provided over time, important for when the input depends on the previous output.

Despite online learning making sense chronologically with the gameweeks, I believe offline learning is suitable. I would have sufficient time to retrain the model between the gameweeks as it should not take too long, however finding a method to efficiently add & combine any new data will be a challenge to overcome.

4.4.2 Natural Language Processing

Natural Language Processing is the technique of analysing human language computationally. I aim to use NLP to perform sentiment analysis on the extracted data, e.g. tweets and comments, to gather whether they are negative, positive or neutral. This will help to gain a consensus of the general feeling around a certain player at any given time. I am to generate a score from overall sentiment of the text data to later use as inputs into the model.

From the literature review, I gathered that a BERT algorithm was a popular and effective method for performing sentiment analysis, however my own fine-tuning to FPL data is out of scope. Therefore, I will be using a fine-tuned BERT model based on Twitter data as this should also work well for Reddit comments due to their similar text styles and formatting. I had a contingency plan for sentiment analysis in case this was not possible, using the VADER open-source social media sentiment analysis library, or even as a backup, a generic sentiment analysis library such as NLTK (Natural Language Toolkit) that is not so fine tuned for analysing social media text.

4.4.3 XGBoost

XGBoost stands for eXtreme Gradient Boosting and is an open-source library that extends gradient-boosted decision trees to perform regression and classification in a supervised learning manner. It is very well optimized for speed and accuracy, and therefore is suitable for my hardware and the scope of this project. As XGBoost has been used thoroughly for machine learning projects similar to this with optimal results, as seen in my literature report, I decided to only use this machine learning technique. (Great Learning Education 2022)

5 Data Sources

5.1 FPL Dataset

The FPL dataset, on GitHub (Anand 2022), includes all of the football statistics to train the model and is updated regularly for each gameweek. It includes an abundance of data from the most recent gameweek to the 16/17 season. The largest challenge is to manipulate it into a format that suits my system: separating the players into their FPL positions, selecting the appropriate features, and formatting it for social media features to be added. For the scope of this project, I use similar features to those found in the previously reviewed literature. From the research gathered in my literature review, I use four different types of attributes: player/individual, player's team, opposing team and FPL specific statistics.

5.2 FPL Metrics

Within the FPL dataset, there are also features regarding the FPL playerbase. These features include the number of 'managers' (users) who have the player selected in their team, the number of managers who have transferred the player into their team this gameweek, the number of managers who have transferred this player out of their team this gameweek and the value of the player within FPL. These four FPL metrics can provide useful information on the fans' opinions across the playerbase, and therefore are something I believe are within the scope of this project. Additionally, other than the 'value' feature, these are new additions over the models researched in my literature review and can be used to evaluate the effect of fans' opinions on a machine learning model to predict FPL points.

5.3 Twitter

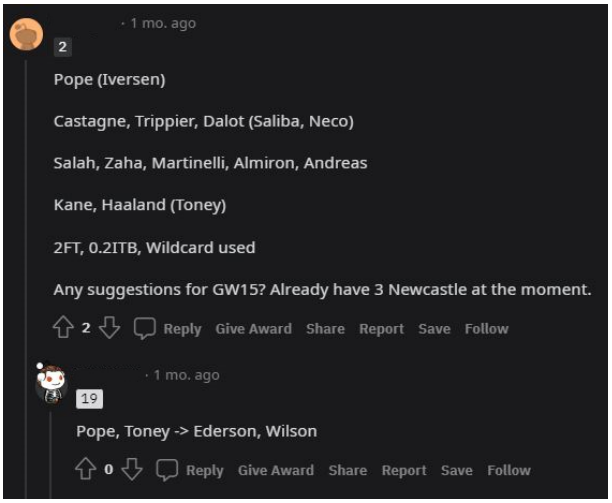
Twitter consists of users posting text or various types of information either on their own profile for their followers to see or as a reply to another user's post. These 'Tweets' often include 'hashtags' which identifies and groups it within a certain topic which enables it to be found by other users browsing related tweets. Via the Twitter API v2 with Academic Access, I am able to search and collect any Tweets in the Twitter archives, with various parameters for the search, including the use of certain hashtags, the date window, and the avoidance of re-posted Tweets known as 'retweets'. (Twitter 2023)

5.4 Reddit

Firstly, the social media site Reddit is made up of many smaller communities, often known as 'Subreddits', which separate posts between topics. Users can then join subreddits to become a part of that community and follow any posts within it. For this project, I am taking data from the FPL subreddit, known as 'r/FantasyPL' which has an active community of over 640,000 members. In particular, there are frequent 'Rate My Team' posts in which members discuss their teams and potential transfers and 'rant' threads for general discussion about individual players and teams. The Reddit API does not have any date range limitations for posts that I can retrieve that will affect my data collection, and therefore can provide thread comments across the two seasons.

Within these 'RMT' posts, people sometimes post their entire teams for people to comment on.

The format of these are usually something like:



[redacted] · 1 mo. ago

2

Pope (Iversen)

Castagne, Trippier, Dalot (Saliba, Neco)

Salah, Zaha, Martinelli, Almiron, Andreas

Kane, Haaland (Toney)

2FT, 0.2ITB, Wildcard used

Any suggestions for GW15? Already have 3 Newcastle at the moment.

[redacted] · 1 mo. ago

19

Pope, Toney -> Ederson, Wilson

Therefore, extracting the players from these teams within this post and counting the frequency of each player may give a good metric into their popularity. Additionally, other comments about specific players within these RMT posts and the rant threads can be processed using the same semantic analysis method as the Twitter data. Additionally, 'Upvotes' are a net score of how other members view their comment, so if lots of people agree with a certain opinion, it is likely to have more upvotes and therefore provides extra sentiment analysis data.

5.5 Hypothesis

My first prediction for this data science experiment is that Reddit will be the most important social media data source. From the early stage in data collection, the Tweets about various players often have no sentiment, especially the lesser known players with few tweets. On the other hand, Reddit's FPL community seem like a dedicated group of people who take it reasonably seriously, expressing their sentiment much more often. Despite this, the FPL data from the playerbase statistics may prove to be a more effective way of finding the trends in the popular, high scoring players.

6 Final Design of the System

6.1 Baseline Model

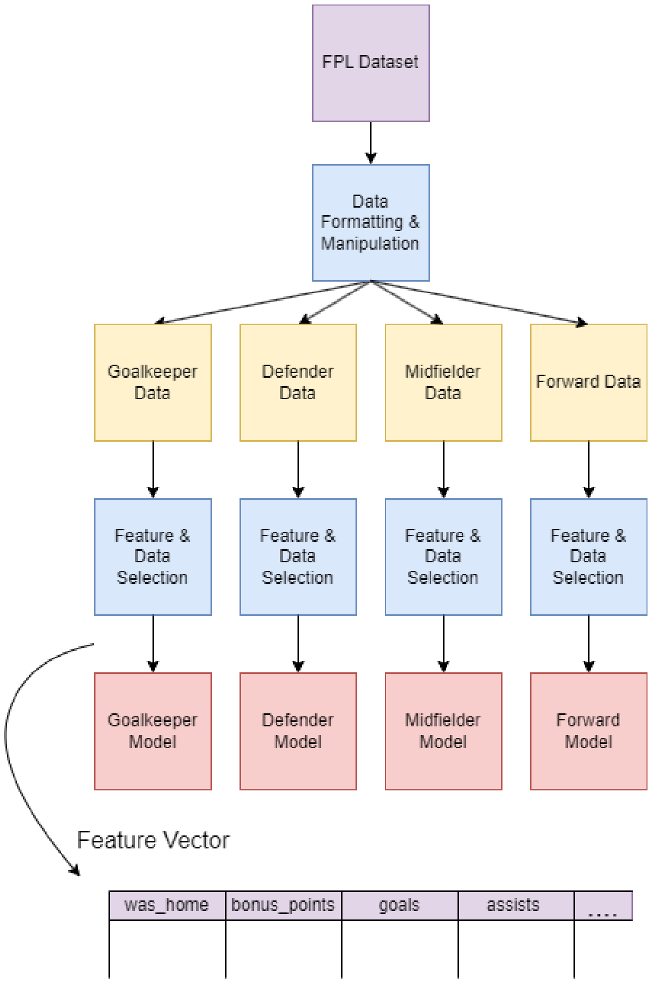


Figure 1: Design of the system before new feature data added

My final design of the system starts with the gathering of necessary data from the FPL dataset, splitting it by each position before performing unique feature selection. This data is then used to train an ML model for each position, making sure it performs similarly to those mentioned in my literature report. Before training the models using XGBoost, the final datasets are created by including data from previous gameweeks for each entry, given the size of the window, for each season and position dataset. This is the overview of the baseline model, as seen in the diagram above.

6.2 Extended Model

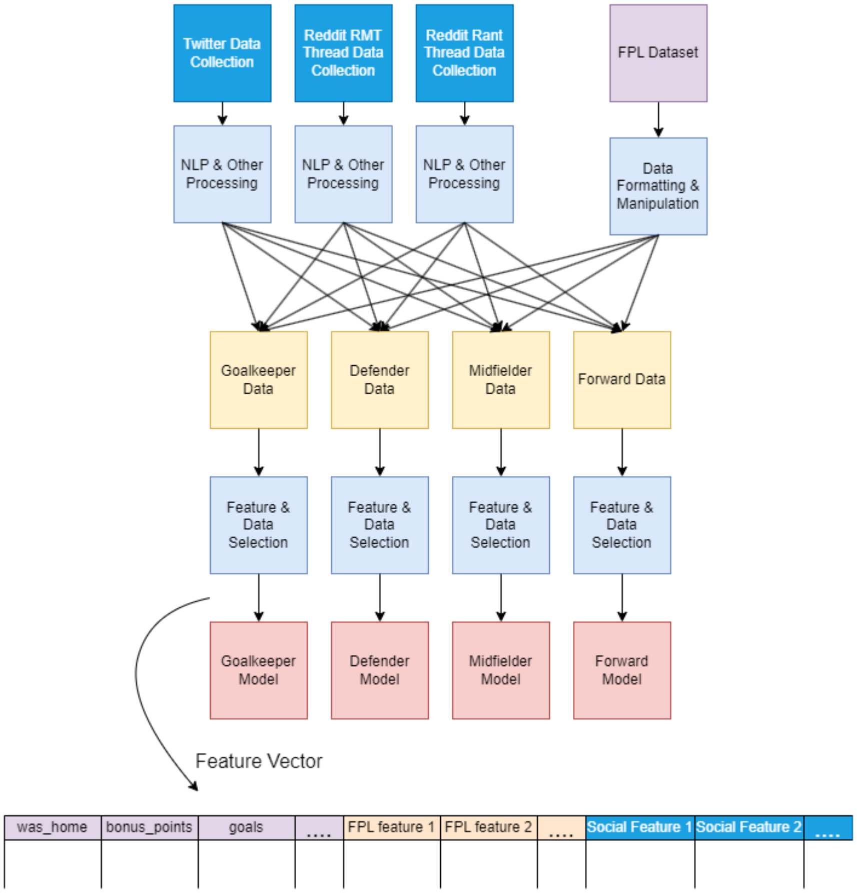


Figure 2: Design of the system after new feature data added

The baseline model is then extended on to investigate the effect of the new features revolving around fans' opinions. Data is then collected using the two APIs for Twitter and Reddit, before it is sorted based on the player being talked about and then processed using sentiment analysis. These new features are then combined with the FPL datasets for each player before creating the final datsets using the previous algorithms to include the window of previous gameweeks. From there, new models are trained on the new datasets, including and excluding various new features to analyse the effect of the different combinations.

7 Implementation

7.1 Data Collection

For both of the Twitter and Reddit APIs, I have only collected the comment/Tweet text data along with the date it was posted. This allows me to avoid collecting any usernames or personal information of any of the users who have posted this data, avoiding possible ethical issues that could arise with identifiable information.

7.1.1 Player Name Cleaning

To collect and search for data on every player using the Twitter API, and to sort the collected Reddit data for every player, I used the list of players within the FPL dataset. This list of players, unique to each season, provided me with their respective id for that season, along with their official first and surname. Despite this being very useful for collecting social media data, the official names of some players did not align with how users online would address them. To make the searching and sorting of social media data effective, I manually added each player's simplified first and surname, e.g. removing any symbols that people are not likely to include online or providing the shortened name that is used more online or on TV, such as 'Martinelli Silva' being called just 'Martinelli' most of the time or full other names such as 'Marcus Oliveira Alencar' being known as just 'Marquinhos'. Especially on Reddit, some users seem to use specific acronyms for players, e.g. Kevin De Bruyne is often reffered to as just 'KDB'. Therefore, I have gone through the effort of identifying these acronyms and storing them in the alternative name column to use for searching and sorting.

7.1.2 Reddit

For the collection of Reddit data, I first had to identify the specific threads that I needed to take the comments from. To make it easier to collect data using the Reddit API, I am using the PRAW python library to search for the required threads using the full title, collecting the thread ID, and date of creation for each one to collect the comments from. This would enable me to sort the threads by gameweek and season, where necessary, and collect the comments from each in an organised manner. PRAW is a library that is licensed under the GNU GPLv3, meaning it is open source, and can even be used for commercial purposes, and therefore is appropriate to use for research purposes in this project. (Bryce Boe 2023) (Free Software Foundation (FSF) 2023)

7.1.3 Twitter

For the Twitter collection, I used the Twitter API to retrieve tweets given various search queries for each player. This would be performed for each gameweek over the two seasons, querying between the dates after the previous gameweek and before the one being predicted for. I used the following search queries to provide the best response and collect the most useful data:

1. First name & surname
2. Surname & '#FPL'
3. Simplified first name & simplified surname
4. Simplified surname & '#FPL'

5. Alternative name & ‘#FPL’

For the search terms where I am not including their first and surname, I am including the use of ‘#FPL’ to ensure they are about football and FPL. This is because a surname or alternative name could collect a lot of data that is not about the intended player, e.g. another person with the same surname.

Within the Premier League, there are only a few players with the same surnames. These search queries and my later data engineering should ensure all of the data is being linked to the correct player.

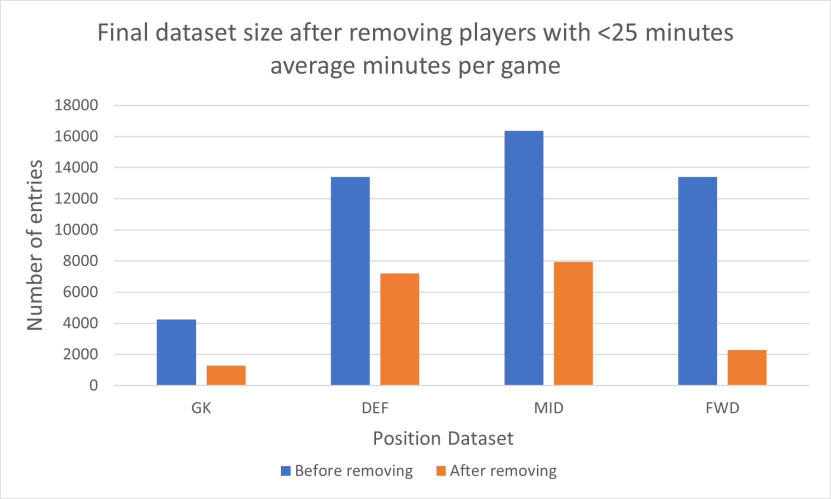
7.2 Data Engineering

Each of the three data sources were stored separately and by gameweek, with Twitter having individual CSV files for each player. To create separate models for each of the positions, I had to separate the FPL and social media data into their respective positions. Additionally, because the FPL data within the dataset is specific to each season, e.g. the player ids linking all of the data together are not consistent between seasons as the list of players are not the same, it was important to keep the data of the two seasons separate until the final merging of the data. As the Twitter data had already been matched to each player due to the nature of its collection, the Reddit threads were a list of comments, and therefore each comment had to be identified to a particular player, using the list of cleaned player names.

Once all of the data had been collected and sorted into the correct season, position and player, it was ready for the Natural Language Processing. Once the NLP had been completed and merged into the same file with the rest of the FPL data, it was ready to be merged into the main dataset. To do so, I had to implement a way to take the previous few gameweeks into account for each entry, with a set window size. To complete this, I took the necessary features from the previous specified entries in the dataset and merged it all into one entry, for every entry. Therefore, each entry being put into the final dataset already has the ’window’ of previous gameweek data already combined as the features. These features were named accordingly, for example ’minutes1’ would be the gameweek at the ’top’ of the window, and ’minutes3’, for a window size of 3, would be regarding the most recent gameweek.

7.2.1 Preventing a naive model

During the merge of all the data into the final datasets for each position and season, I made sure to make some necessary exclusions. If all players listed in FPL were used to train the model, the points for each gameweek will end up being nearly entirely between 0 and 2, as a lot of them do not regularly play. This would therefore show as being an accurate model as it would keep predicting low scores for every player, and being very poor at predicting the points for players who do play consistently. This is especially important for FPL as when selecting players, you would look towards those that play regularly and score a lot of points. Therefore, I am excluding all data from players that have an average of less that 25 minutes played per match to ensure the model is training on quality data. This should provide much more useful predictions, compared to a naive model. The graph below puts into perspective how many players have been removed because of this:



7.3 Natural Language Processing

For the NLP, I used a roBERTa-base model for sentiment analysis called Twitter-roBERTa-base, which has been fine-tuned using around 58 million tweets (Cardiff NLP 2023). This has been downloaded and used within the 'transformers' python library to perform sentiment analysis. This highly powerful model is loaded using the 'transformers' library before tokenizing the input specifically for the type of model chosen and passing it through the model's encoder. These encodings for each input are then brought together to receive a vector representation that is then used by the models predictive ability to estimate the sentiment as a value between -1 and 1. (Lewis Tunstall 2022) (Barbieri et al. 2020)

Before the Tweet text is passed into this model for sentiment analysis, I have included some pre-processing steps to help improve its performance. These steps include removing duplicate entries, removing Twitter handles, removing URLs, removing line breaks and extra whitespace, removing stop words, and lemmatization. This is important as the model I am using for sentiment analysis is trained using tweets where the Twitter handles are anonymised and URLs and line breaks are removed. (Barbieri et al. 2020)

The number of social media entries for each data source is another aspect of the data collected that I will be using as a feature in a model. The aim of this is to measure the popularity of a player based on how much they are talked about across the two platforms, with the hope that this could prove to be a useful metric for predicting FPL points.

7.4 Model

7.4.1 Random Seeds & Averaging

When testing during the development of the XGBoost model, I noticed a variance in the performance metrics for the same train data. To solve this issue, I set a constant random seed value during initialisation to ensure that the same sequence of numbers was being used during each run of the model. This makes the model reproducible and therefore appropriate to be able to compare the change in performance when making slight changes in the data or model. Additionally, there seemed to be a small but noticeable difference in performance depending on the seed being set. To tackle both of these issues, each run would train 15 different, with 15 specific seeds to keep it consistent, before averaging the results across all of the models trained.

7.4.2 Hyperparameters

Additionally, hyperparameters are likely to perform differently on varying datasets. In this case, finding optimal hyperparameters for the baseline model, and keeping them set when adding the new FPL and social media features, could have an impact on the later models that I will be evaluating. If the hyperparameters are tuned for the baseline model, they may not be effective for the other models, and therefore to keep the consistency in my results and evaluation, I have only specified two hyperparameters. These include 'n_estimators' and 'learning_rate', which I have set to 1000 and 0.01 respectively. The number of estimators refers to the number of decision trees to be within the ensemble, where I have increased it enough to train effectively, but not too much to cause overfitting. The learning rate controls how much of a step to make to correct from mistakes from previous trees in the ensemble. I have specified a value low enough to train thoroughly, although not too small to the point where it takes too long to train each model, especially taking into account that I am training 15 models and averaging the results. (xgboost developers 2023)

7.5 Feature Selection

Feature	Description
Features specific to the current gameweek	
points (label)	The number of points scored
was_home	Boolean value on whether the game will be played at home
opponent_strength_attack_home	Strength of the opponent teams attack playing at home
opponent_strength_attack_away	Strength of the opponent teams attack playing at away
opponent_strength_defence_home	Strength of the opponent teams defence playing at home
opponent_strength_defence_away	Strength of the opponent teams defence playing at away
opponent_last_finish	Position in the table that the opponent team finished last season
opponent_strength	Strength of the opponent team overall
total_points_tally	Total FPL points so far this season
goals_scored_tally	Total number of goals scored so far this season
goals_conceded_tally	Total number of goals conceded so far this season
assists_tally	Total number of assists so far this season
bps_avg	Average number of FPL Bonus points this season
minutes_avg	Average number of minutes played this season
selected	Number of FPL players who have this player in their team
transfers_in	Number of FPL players who have transferred this player in for this gameweek
transfers_out	Number of FPL players who have transferred this player out for this gameweek
value	Value of the player on FPL
rant_sentiment	Sentiment of the player within the Rant Reddit thread
rant_count	Number of mentions of the player within the Rant Reddit thread
rmt_sentiment	Sentiment of the player within the RMT Reddit thread
rmt_count	Number of mentions of the player within the RMT Reddit thread
twitter_sentiment	Sentiment of the player from the collected Twitter posts
twitter_count	Number of Tweets collected about the player
Features taken from set number of previous gameweeks	
total_points	Number of FPL points scored
bps	Number of FPL Bonus points scored
goals_scored	Number of goals scored
goals_conceded	Number of goals conceded
assists	Number of assists
yellow_cards	Number of yellow cards
red_cards	Number of red cards
minutes	Number of minutes of the match played
saves	Number of saves made by the player (GK Only)
penalties_saved	Number of penalties saved by the player (GK Only)
opponent_strength_attack_home	Strength of the opponent teams attack playing at home
opponent_strength_attack_away	Strength of the opponent teams attack playing at away
opponent_strength_defence_home	Strength of the opponent teams defence playing at home
opponent_strength_defence_away	Strength of the opponent teams defence playing at away
opponent_last_finish	Position in the table that the opponent team finished last season
opponent_strength	Strength of the opponent team overall
creativity	FPL creativity value
influence	FPL influence value
threat	FPL threat value

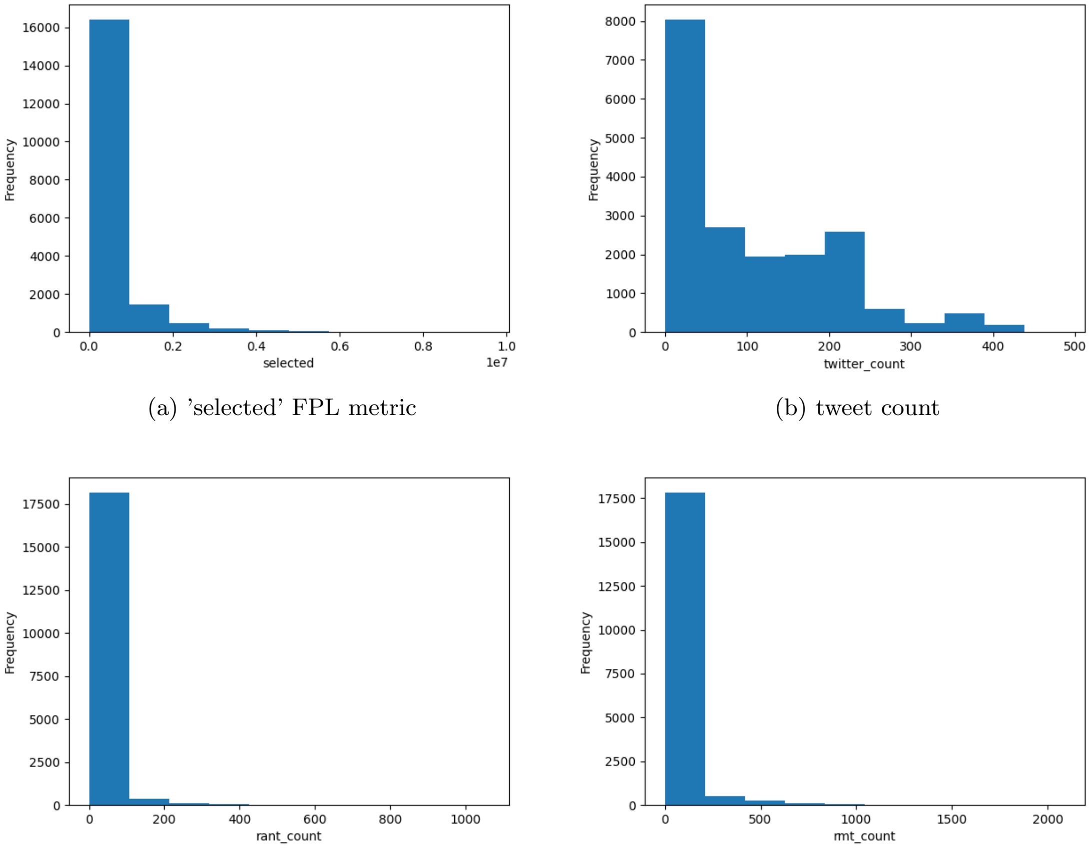
🟩 Added FPL features
🟦 Added social media features

When creating the baseline model and final datasets, I made what I believe to be sensible choices for the feature selection, taking valuable features from the FPL data whilst taking inspiration from the models researched in my literature report. I also had to make sure that any features regarding the current gameweek, which the points are predicted for, would be available prior to the FPL deadline to change your team before the matches start. For example, whether the game was at home or not is available long before kick-off, along with the strength of the opponent team and the players average stats for various features so far during the season. Especially with the size of the window multiplying the number of features within the 'Features taken from set number of previous gameweeks' set, I had to ensure that I did not create too many features to avoid the curse of dimensionality and creating too much noise.

Additionally, I had to create some features manually. For the 'tally' and 'avg' features, I calculated the average or tally of various features at each gameweek in the season for each player so that the model could get a sense of how that player has been performing across the whole season and not just the last couple of games. Along with this, using separate datsets, I worked out the position the opponent team finished in the league last season and found official strengths metrics for each team within the FPL dataset.

7.6 Dataset Analysis

7.6.1 Social Media Frequency Distribution



(a) 'selected' FPL metric

(b) tweet count

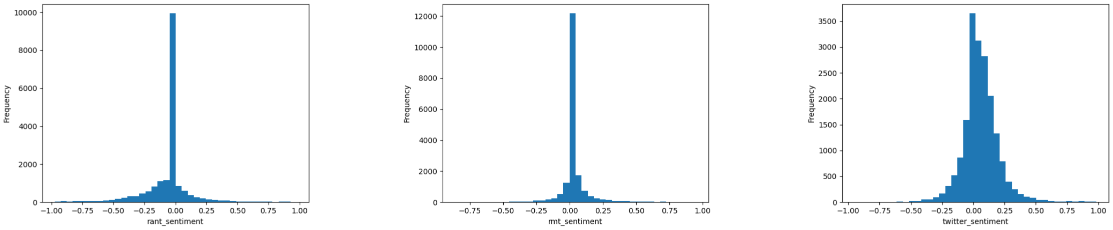
(c) rant thread comment count

(d) rmt thread comment count

Figure 3: Distribution of the number of social features compared to distribution of FPL selected metric

The above graphs display the frequency distribution of the social media data features along with the distribution of the 'selected' FPL metric. This seems to show that only a relatively small handful of players are really talked about, especially on Reddit. This suggests that Reddit users seem to only talk about those players that are performing especially well within FPL, of which will be the same players being selected by a majority of FPL players. This is proven by the similar shape of the two Reddit distribution graphs and the 'selected' FPL metric distribution graph. On the other hand, Twitter seems slightly more varied in the players that are talked about. Note that these distributions are after removing the data from players that play less than 25 minutes on average per gameweek, suggesting that these graphs would be even more extreme if that data had not been removed.

7.6.2 Social Media Sentiment Distribution



(a) Rant Thread

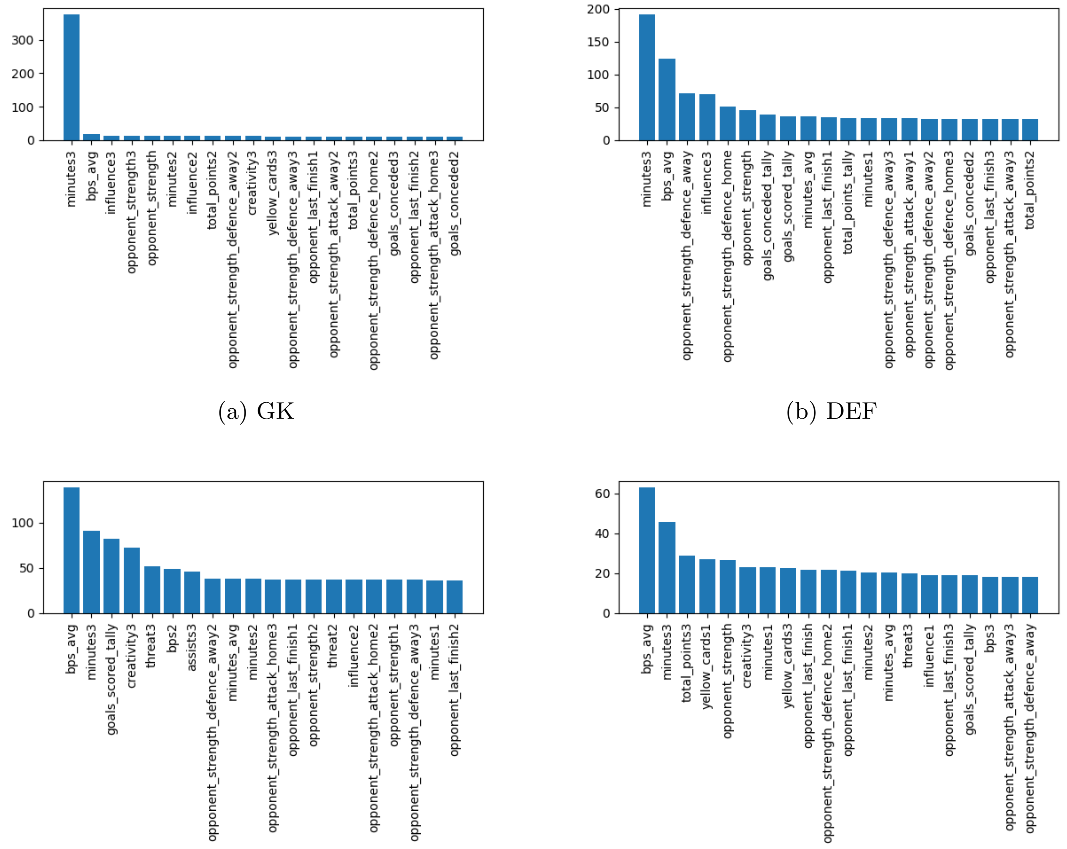
(b) RMT Thread

(c) Twitter data

Figure 4: Social media sentiment distributions

The graphs above display the distribution of the sentiment data for each of the social media sources. If there is no social media data for a player, it results in a sentiment score of 0, meaning it is completely neutral. As seen in these sentiment distribution graphs, there is a large skew in data entries with a sentiment score of 0 due to these players not being talked about online for the given gameweeks, especially within the Reddit community. On the other hand, the Twitter data seems to vary slightly more. This could be due to Twitter being a much larger platform that contains a lot more data on players that is not only related to FPL, whereas the Reddit community is purely for selecting the best players for FPL. Despite this, I expect the Reddit sentiment data to be more valuable when training the model, with positive and negative sentiment possibly having a greater indication of the most talked about players.

7.6.3 Baseline Feature Importance



(a) GK

(b) DEF

(c) MID

(d) FWD

Figure 5: Baseline model 'gain' feature importance

These figures above display the top 20 most important features based on the 'gain' of the XGBoost baseline model. It can be seen that the feature 'minutes3' is particularly important across all of the baseline models, suggesting just how important a players predicted points for given week can be down to how many minutes they played the match before. This is especially the case for the GK model, where it is common for a goalkeeper to play many matches in a row, only swapping out if they are injured, or are replaced by a #1 choice goalkeeper that has been injured. Additionally, the average number of bonus points per game across the season so far 'bps_avg' seems to be a good indication of how well the player has been playing, and therefore how many points are forecasted for them, as it is within the top two features along with 'minutes' for every model. The 'weight' feature importance of these XGBoost models displays a vastly different set of the top features, which suggests it is not too much of an issue at this point having 'minutes3' and 'bps_avg' dominating in importance.

I have used the 'gain' metric for feature importance at this stage to gather the impact of each feature on the model. This method seems to be the most common for measuring feature importance, over other metrics such as 'weight', which I explore further in my results, and 'cover'. As this is just looking at the baseline model features, 'gain' is sufficient for measuring importance. (Amjad Abu-Rmileh 2023)

8 Results & Evaluation

8.1 Model Accuracy Metrics

I will be using the Root Mean Squared Error (RMSE) metric when measuring the performance of the models during my testing and results over other metrics such as Mean Average Error (MAE) and Mean Squared Error (MSE). Firstly, this is because within my dataset, I am expected either a player to score between 0 and 2 points or do particularly well and score a lot higher at around 5+ or some outliers of 15+. Therefore, RMSE penalises these errors more and provides a more accurate prediction to how well the model has estimated a player performing well or not. Additionally, the previous literature that I am working from also uses RMSE, which allows me to compare results between the two. Despite this, I will also be using MAE for my final results as these show exactly how many points on average the model predictions are off, providing more of a context on how well they perform.

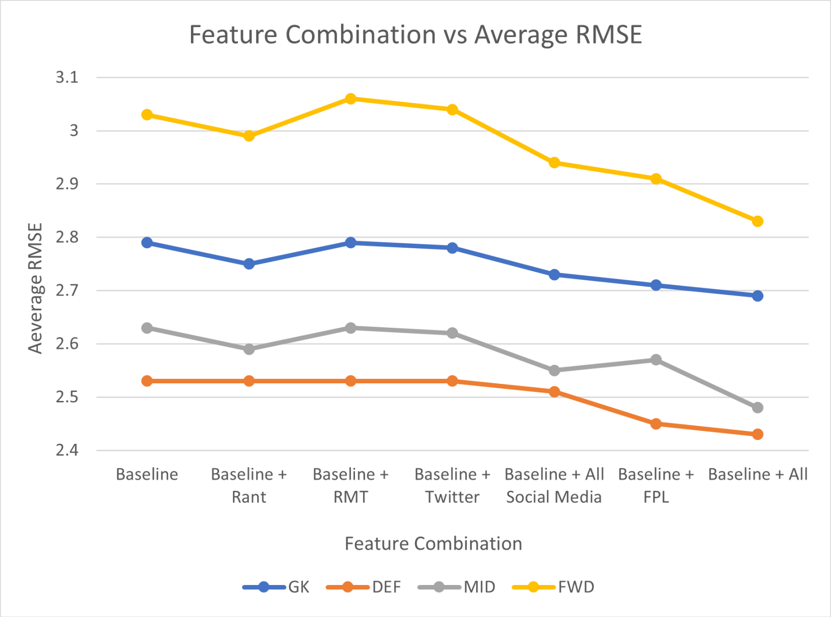
8.2 Model Setup

I will begin my results by comparing the impact of various combinations of the new features added to the baseline model with a fixed window size to keep it consistent. The window size to decide how many of the previous gameweeks to consider will be set to 3, with a 85/15 train/test split using FPL and social media data covering both the 2021/22 and 2022/23 (up to and including gameweek 28) seasons. As mentioned in the implementation section, all players with less than 25 minutes average per game have been removed from the data and each result is the average RMSE across 15 training scenarios, with the set random seed values. A window size of 3 seemed most appropriate during my initial stages of testing, however after the initial results have been evaluated, I will investigate how various window sizes can impact the predictive ability of each of the models.

For my initial results, I display the difference in RMSE between the various combinations of the new features added to the dataset before training the XGBoost model. These combinations include:

1. the baseline plus only the Reddit rant thread features
2. the baseline plus only the Reddit RMT thread features
3. the baseline plus only the Twitter data features
4. the baseline plus all of the social media data (Reddit rant, Reddit RMT and Twitter) features
5. the baseline plus only the FPL metric features
6. the baseline plus all of the new added features mentioned

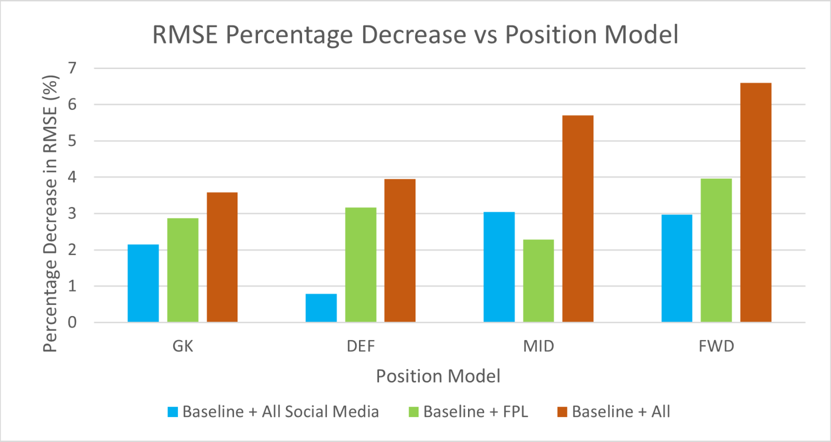
8.3 Initial Results



Results (Average RMSE)							
Position Model	Baseline	Baseline + rant	Baseline + RMT	Baseline + Twitter	Baseline + All Social Media	Baseline + FPL data	Baseline + All
GK	2.79	2.75	2.79	2.78	2.73	2.71	2.69
DEF	2.53	2.53	2.53	2.53	2.51	2.45	2.43
MID	2.63	2.59	2.63	2.62	2.55	2.57	2.48
FWD	3.03	2.99	3.06	3.04	2.94	2.91	2.83

From the results seen above, there is a clear improvement on the predictive performance of every position model after the inclusion of fans' opinions via the use of social media data and FPL metrics. Out of the individual inclusions of the social media data, it seems as though the Reddit Rant thread data has been the most effective at improving the baseline model, as seen by the 'dips' in the graph at the second point when compared to the baseline + RMT and baseline + Twitter points. The inclusion of the FPL playerbase statistics, as seen in the second to last data points on the graph, seems to outperform the social media features for all of the positions other than the midefield model. Although the FPL metrics seem to have a greater impact overall than social media data, when the baseline model includes all of the added features, it outperforms all other combinations.

One especially interesting set of results to investigate is the impact of the new features on the FWD position model. There is a noticeable increase in the RMSE when only including each of the social media data sources, suggesting these features could be acting as noise, impacting the models predictive ability. However, when using all of these features for training the model, instead of only one at a time, there is a sudden decrease in the RMSE. This pattern, despite not being quite as obvious for the other models, it displays how the social media features alone are not significant enough to provide any significant improvements, and therefore should all be used together to provide the best possible models.



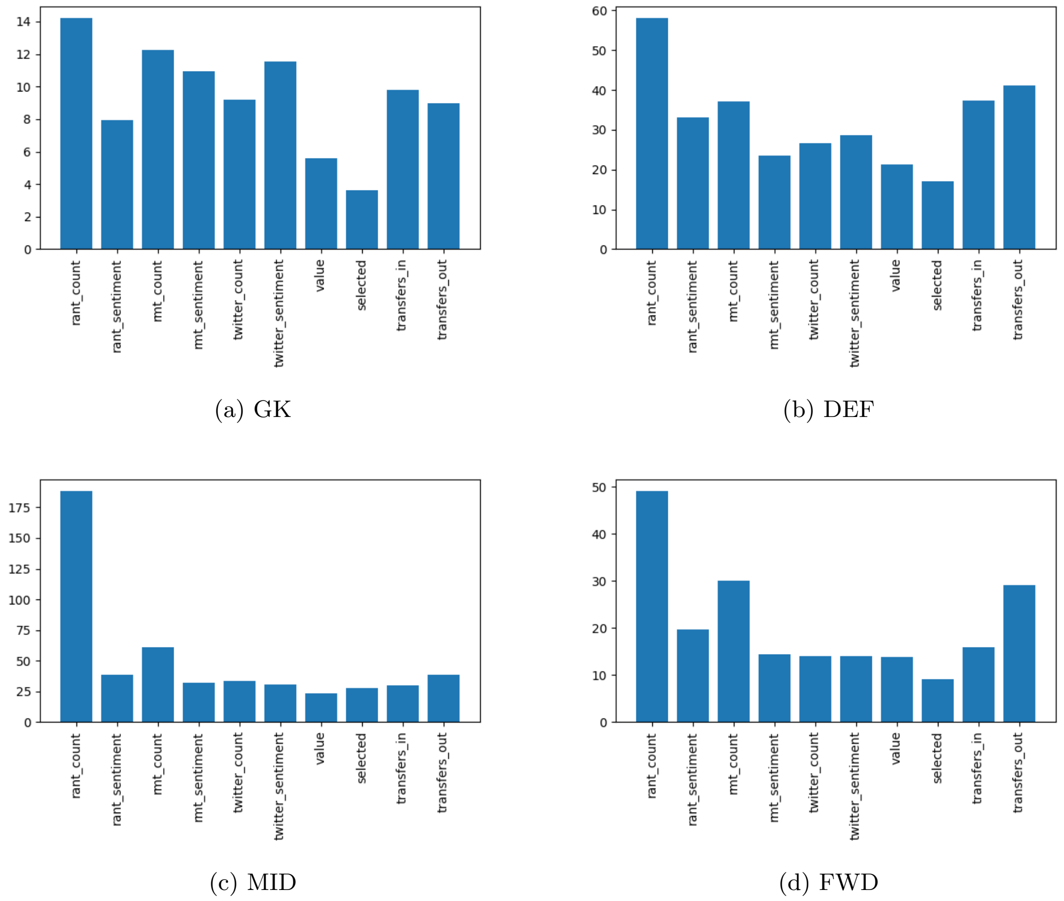
This graphic displays clearly how the three scenarios (1.Baseline model with social features, 2.Baseline with the FPL features and 3.Baseline with both FPL and social features) improve each of the position models, as shown by a percentage decrease in the RMSE.

8.4 Feature Importance

To measure how impactful each of the new features are compared to each other, I have investigated their feature importance within the models. Because XGBoost uses parallel decision tree boosting, feature importance can be measured in many different ways. The two most common of these being 'weight' and 'gain'. I have explored both to gain a full understanding of the new features.

8.4.1 Gain

Gain is the feature importance metric to measure 'the average gain across all splits the feature is used in'(xgboost developers 2023). This essentially measures the relative contribution of each feature using an F score. Below are the F score values in terms of gain for each of the new features added to the model, for each position model.



(a) GK

(b) DEF

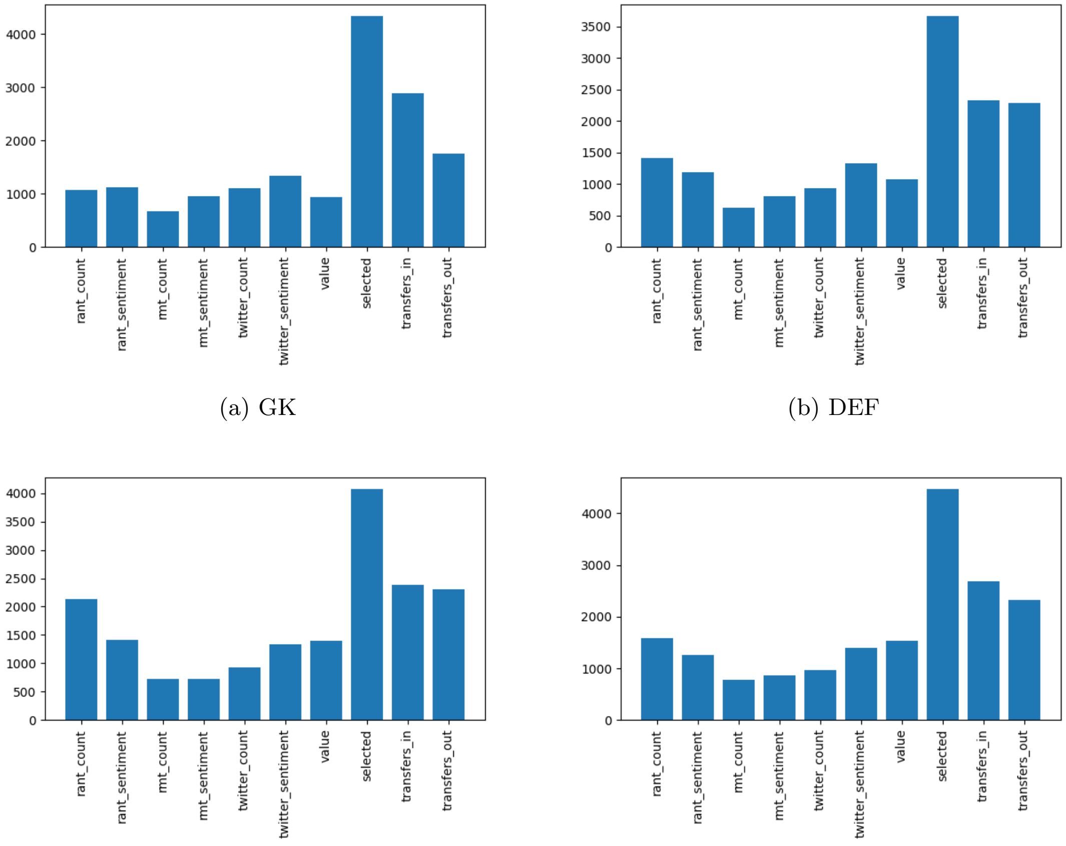
(c) MID

(d) FWD

Figure 6: Importance of new features in terms of gain

8.4.2 Weight

Weight is the feature importance metric to measure 'the number of times a feature is used to split the data across all trees'(xgboost developers 2023). This instead measures the absolute contribution of each feature using an F score. Below are the F score values in terms of gain for each of the new features added to the model, for each position model.



(a) GK

(b) DEF

(c) MID

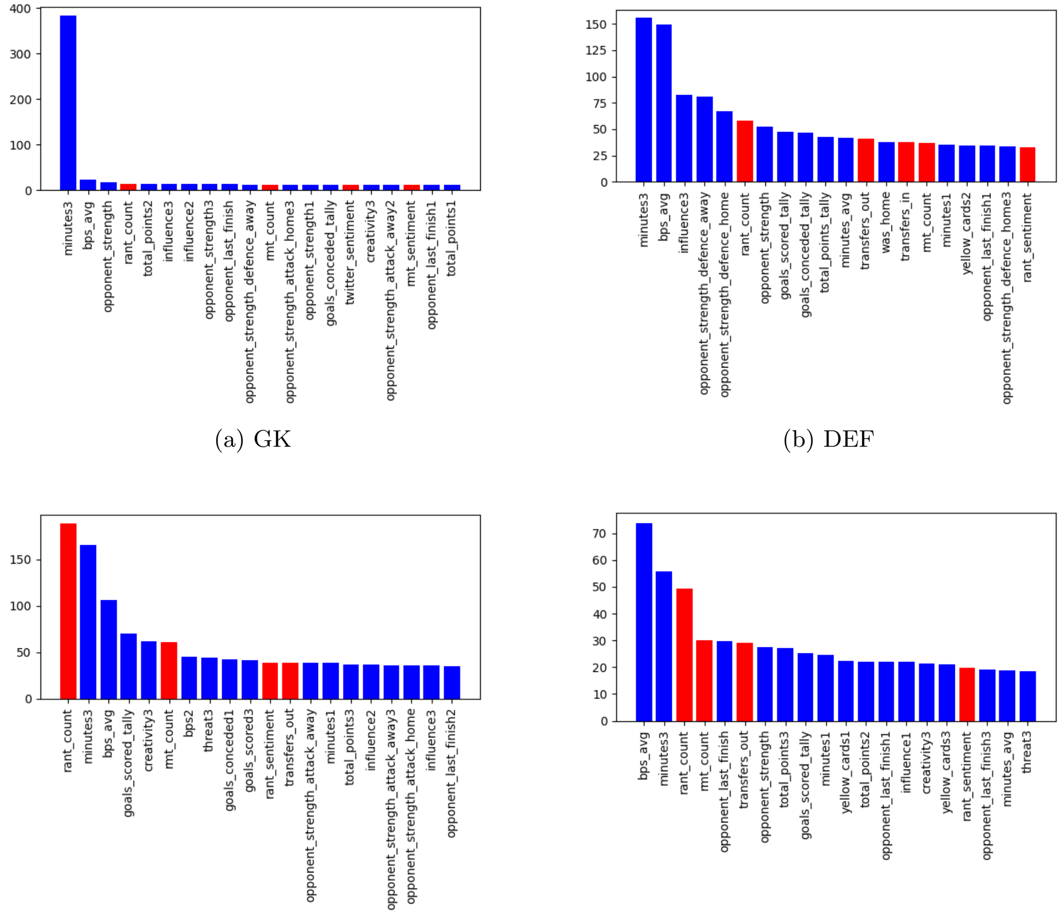
(d) FWD

Figure 7: Importance of new features in terms of weight

8.4.3 'gain' vs 'weight' Importance Comparison of new features

The two features that do not correlate between the gain and weight feature importance metrics are 'rant_count' and 'selected'. Firstly, 'rant_count' has a high gain and a much lower weight, which displays how it is used less for splitting the data, however when it is used to split the data, it leads to significant improvements in the performance of the model. This suggests that it is still a highly informative feature, even if it is not used too much within the decision trees. On the other hand, the 'selected' feature from the FPL dataset has a much higher gain and a lower weight. This suggests that it is still an important feature that is highly informative, however it is likely that it is already correlated with other features.

8.4.4 Baseline Feature Importance Comparison



(a) GK

(b) DEF

(c) MID

(d) FWD

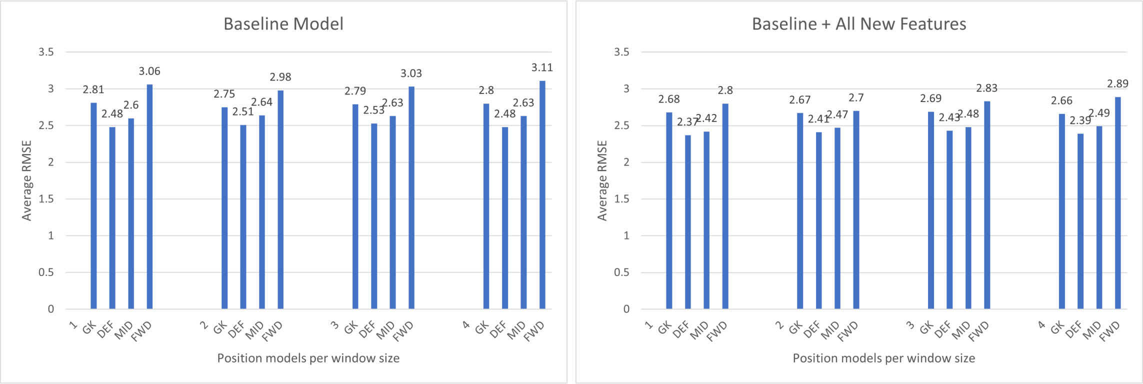
Figure 8: Feature Importance in terms of gain across the top 20 features

These graphs above display the same graphs explored in the implementation section, plotting the feature importance of the top 20 features in terms of gain. Highlighted in red are the new features added to the baseline models that are now within the top features. It is clear that these new features are having a significant impact, being important across all of the models. Particularly, the number of comments regarding a player within the two Reddit threads are consistently some of the most important features for every model. Despite this, 'transfers_out' is important for the outfield position models, along with the sentiment of the Reddit rant thread regarding the player. These graphs prove my hypothesis correct, as the Reddit features are proven to be more important than the Twitter features across all of the position models.

The Goalkeeper model is vastly different to the other position models, with 'minutes3' still being drastically more important in terms of gain than any other feature. In addition to this, it is the only model which has none of the new FPL features within the top 20, only social media features.

8.5 Window Size

During the initial creation of the baseline model, the window size of 3 seemed the most appropriate in terms of being a sensible number of previous games to include and being the most consistent in performance. However, after the final changes to the model had been made, along with the inclusion of the social media and FPL data, it is clear that a window size of 3 is not always the most optimal. In fact, these graphs display that the optimal window size is not the same for each position model. Especially for the Forward model, too high of a window size possibly creates noise and increases the error, however too low of a window size could be too simple to train effectively. Although this may be true for the Forward model, it is clear this is not the case with other position models.



(a) Baseline Model

(b) Baseline + All Model

Figure 9: Window size comparison across both types of position model

8.6 Final Models

Best Performing Position Models							
Position Model	Window Size	**RMSE**	**MAE**	Baseline RMSE	Baseline MAE	RMSE Decrease (%)	MAE Decrease (%)
GK	4	**2.66**	**2.05**	2.8	2.16	5	5.09
DEF	1 or 4	**2.37**	**1.67**	2.48	1.75	4.44	4.57
MID	1	**2.42**	**1.58**	2.6	1.7	6.92	7.06
FWD	2	**2.7**	**1.83**	2.98	2.03	9.4	9.85

It is clear that the best performing model for each position is the one with all of the new social features and FPL features included in the dataset. This table displays the best performing window size for each of these optimal position models, displaying the decrease in error for each, using the MAE and RMSE metrics. Overall, these optimised models are relatively successful for forecasting

FPL points for a given player and match, being able to estimate within a low of 2.05 points for goalkeepers, and a high of 1.58 points for midfielders on average.

Fans opinions are orientated around a particular player, where a forward in football may perform better due to individual skill and moments in the game. Therefore, the percentage decrease in the RMSE and MAE may be highest for the FWD position as it gives a better indication of how a player might stand out beyond their given stats and team performance.

9 Conclusions & Future Work

9.1 Conclusion

From my results, I can conclude that the inclusion of data that portrays fans' opinions can positively effect a machine learning model, in this case to predict fantasy football points for a given player. The nature of predicting how a football player is going to perform in a real world match is impossible to do perfectly, and therefore lowering the error in the predictions is difficult. As seen in the results and evaluation, the addition of all the new features managed to decrease the RMSE between 4 and 9.4% across the four position models. Despite this not being a huge decrease in the error of the predictions, it is still a notable positive change and therefore proves my experiment successful. In the context of predicting real world sports, such as fantasy football points, it would make sense that the lower the error, the harder it is to improve. Given this, my most optimal models could predict a player's FPL points for given gameweek within 2.05 points for goalkeepers, 1.67 points for defenders, 1.58 points for midefielders and points 1.83 for forwards on average.

Therefore, these optimised models from this experiment could now be used by FPL 'managers' to help decide on which players they are going to pick in any upcoming gameweeks and gain a possible advantage over those who are not using data science. In this case, using the FPL data and social media data representing fans' opinions online, these models are potentially more powerful at predicting FPL points for a given player than many of the currently existing ML models that are just using football statistics.

9.2 Future Work & Possible Improvements

My efforts to include football fans' opinions into a predictive machine learning model have proven to be successful, however this is just the beginning of what could be explored further.

For each of the position models, I found the optimal window size after determining that having all of the new features included within the dataset provided the best predictive models. Despite this, to further optimise these final models, the XGBoost hyperparameters can be tweaked to get the best possible performance for each of the models, tailored to their features. I felt this was out of scope of the project, veering a little too far out of the main aims of the experiment, however if I were to try and improve on the models, this would be my first idea.

Additionally, other machine learning techniques could be used to train a model for each position. For example, instead of only using XGBoost, other ML regression algorithms could be used, such as Linear Regression or Stochastic Gradient Descent via the sklearn python library or even deep learning with a neural network using tensorflow. Even though I had been working from previous literature which heavily reccomended the use of XGBoost for this experiment, it may still be a good idea to explore other algorithms in case they perform better, especially with the addition of the new features.

Regarding changes in the approach to the NLP aspect of this experiment, other models could have been used to perform sentiment analysis. The TweetEval benchmark compares various models across a range of capabilities in a Twitter based context, with semantic analysis being one of them. Even though the Twitter-roBERTa-base model that I used scores well in this leaderboard for sentiment analysis, the TimeLMs-2021 model performs slightly better. This could potentially help provide more accurate sentiment analysis values which could lead to better quality social media features and improved models (Loureiro et al. 2022). In addition to this, testing whether my pre-processing step of removing stopwords improved the sentiment analysis accuracy or not, as the model used does not seem to be trained on data where the stopwords had been removed (Barbieri et al. 2020). When testing my script that performs sentiment analysis, I noticed that it did not handle sarcasm well. Sarcasm is often used in Reddit threads and Twitter and therefore could have an impact on the quality of the data used to train the models. If the goal is to lower the error on the predictions as much as possible, this could be another area to be improved upon, despite its difficulty.

Furthermore, another possibility to improve the performance of the models could be to create more features from the data I had access to. For example, to improve the GK position model, I could create features such as the tally of saves or the average number of saves per game so far in the season like I have with other features in the dataset.

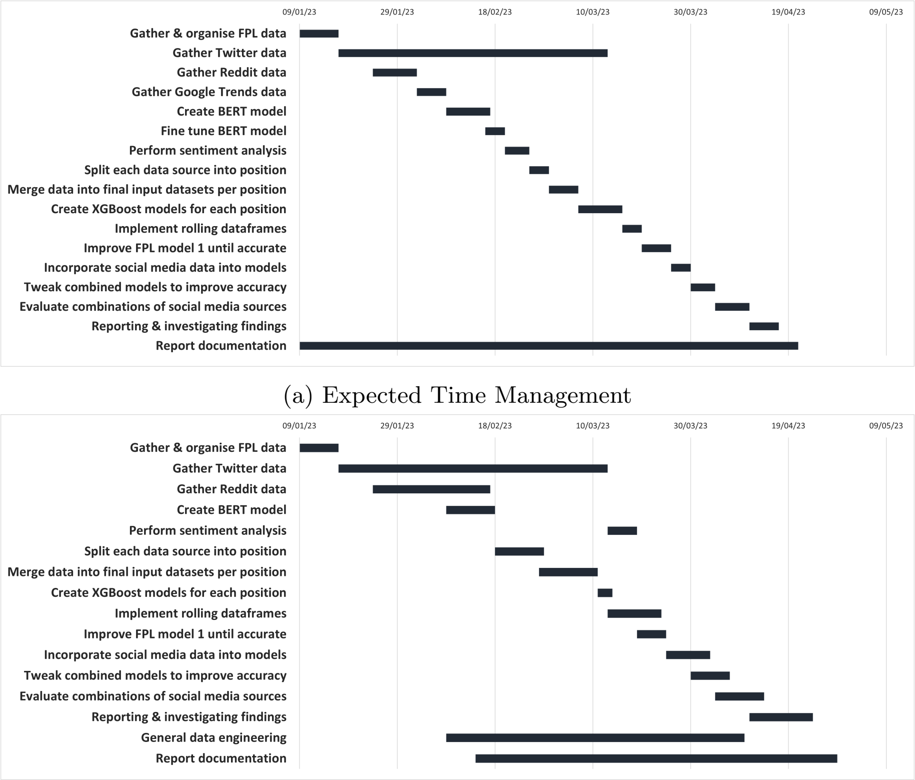
To improve on the data sources used to create the new features in the dataset that represent the fans' opinions, I could include other social media sources. For example, taking data from various FPL discussion groups on Facebook, or data from Google trends where the search frequency for a player could possibly provide an insight into the excitement around a player and therefore FPL points. To improve the existing data sources, I could perform more rounds of querying for Twitter data, increasing the number of Tweets gathered, possibly improving the quality and quantity of the Twitter data. This would result in a greater difference in the number of Tweets collected between players that are talked about online, and those who are not, however this could still help to improve the predictive ability of the models.

Finally, taking more of the football context into account could also help provide an improvement to the points forecasting. This could include identifying player injuries and incorporating this into the model training, or even include their performance across other competitions that are not just the Premier League. On top of this, I could incorporate more seasons worth of FPL and social media data to create larger datasets for the models to train from, potentially gaining an enhancement in predictive ability for FPL points.

These models could be taken into a more real world application from here, converting the whole system of predicting a player's FPL points into a more fluid system. For example, a GUI application to select players to predict points for the next gameweek, where it would collect the data from the social media sources, automate the data engineering aspects and output the predicted points for each of the players within one pipeline. The output could be displayed in order of the highest predicted points for example, and the models would fetch the new gameweek data to retrain on after each gameweek. This software, e.g. via a webapp or mobile app, could then be used by the FPL playerbase to help make their choices on which transfers they should make. Additionally, like the aims of some of the literature I have used for research, it could provide the optimal team for a given gameweek with the £100 million budget, or even have the user input their team and have the software advise the best possible transfers from the predictions.

10 Project Management

10.1 Time Management



(a) Expected Time Management

(b) Actual Time Management

Figure 10: Expected vs Actual Time Management Gantt Charts

One of the clearer differences between the two Gantt charts in terms of my use of time for this project is how I expected to finish each task before working on another. In reality, it was at times important to be working on multiple different parts at the same time to prepare accordingly for the next stages of development. Examples of this were the two tasks 'implement rolling dataframes' and 'improve FPL model 1 until accurate'. This is where I was adapting how the datasets were formed using the rolling dataframes in a sense to create the effect of including a window of previous gameweekas at the same time of editing and improving the model depending on the dataframe changes. There were also some tasks that I didn't need to do, such as the fine-tuning of the BERT model due to there existing an already fine tuned model using Twitter data. On the other hand, the new task of 'general data engineering' has been added where a lot of the work done behind the scenes took a lot of the time throughout most of the tasks, consisting mostly of writing scripts to organise the data appropriately at each stage.

10.2 Limitations & Issues Faced

The main issue that I experienced during this project was down to the Twitter API. During the early stages, I was limited to the basic Twitter API access level, only being able to receive tweets from the last 7 days. After eventually providing enough evidence, I was granted access to the academic level, providing me with access to the full archive of Tweets, being able to search for them between given dates. Nevertheless, I was faced the limitations of collecting Tweets through the API, where I could only make 50 requests every 15 minutes. I was querying a maximum 5 times per player, per gameweek, per season, and therefore collecting all of the data took a very long time. This became an issue whenever the internet would go down, or realised I had made a mistake, however I managed to eventually collect all of the required data, even as the gameweeks from this season were being played. This limitation caused my dataset to involve only 2 seasons worth of data, instead of a possible 3+ seasons.

References

Amjad Abu-Rmileh (2023). *The Multiple faces of 'Feature importance' in XGBoost.* URL: https://towardsdatascience.com/be-careful-when-interpreting-your-features-importance-in-xgboost-6e16132588e7.

Anand, Vaastav (2022). *FPL Historical Dataset.* Retrieved August 2022 from https://github.com/vaastav/Fantasy-Premier-League/.

Barbieri, Francesco et al. (2020). "Tweeteval: Unified benchmark and comparative evaluation for tweet classification". In: *arXiv preprint arXiv:2010.12421*.

Bryce Boe (2023). *PRAW: The Python Reddit Api Wrapper.* URL: https://praw.readthedocs.io/en/v3.6.2/.

Cardiff NLP (2023). *Twitter-roBERTa-base Hugging Face Model.* URL: https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment.

Fantasy Football Scout (2022). *How to rate your FPL team using our points projection tool.* URL: https://www.fantasyfootballscout.co.uk/.

Free Software Foundation (FSF) (2023). *GNU General Public License.* URL: https://www.gnu.org/licenses/gpl-3.0.en.html.

Great Learning Education (2022). *Understanding XGBoost Algorithm.* URL: https://www.mygreatlearning.com/blog/xgboost-algorithm/.

Lewis Tunstall Leandro von Werra, Thomas Wolf (2022). "Natural Language Processing with Transformers, Revised Edition". In.

Loureiro, Daniel et al. (2022). "Timelms: Diachronic language models from twitter". In: *arXiv preprint arXiv:2202.03829*.

Melton, Chad A et al. (2022). "Fine-tuned Sentiment Analysis of COVID-19 Vaccine–Related Social Media Data: Comparative Study". In: *Journal of Medical Internet Research* 24.10, e40408.

Nisumaa, Kristian (2019). "Predicting Player Performances in Fantasy Premier League Using Machine Learning". In: *ECS Archives*.

Premier League (2022). *Fantasy Premier League.* URL: https://fantasy.premierleague.com/.

Sports Reference (2022). *xG Explained.* URL: https://fbref.com/en/expected-goals-model-explained.

Twitter (2023). *Twitter API Documentation.* URL: https://developer.twitter.com/en/docs.

Walters, Brody (2021). "Using ML to predict the performance of Fantasy Premier League players". In: *ECS Archives*.

xgboost developers (2023). *XGBoost Documentation.* URL: https://xgboost.readthedocs.io/en/stable/python/python_api.html#xgboost.Booster.get_score.