Applied Data Science Project: Predicting the Premier League with Twitter

Nikita Arsentev
University of Bristol
Dept of Computer Science
Bristol, UK
bx18273@bristol.ac.uk

Cheuk Ho Chan
University of Bristol
Dept of Engineering Maths
Bristol, UK
pq18875@bristol.ac.uk

Tom Bewley
University of Bristol
Dept of Computer Science
Bristol, UK
tb14344@bristol.ac.uk

Ting-Hung Chen
University of Bristol
Dept of Computer Science
Bristol, UK
hu18727@bristol.ac.uk

James Keen
University of Bristol
Dept of Engineering Maths
Bristol, UK
jk15863@bristol.ac.uk

***Abstract*—This project aims to understand the accuracy and biases of predictions about English Premier League football matches made on the Twitter social media platform. A detailed, multi-view comparison of predictions to both pre-match odds and true results reveals a multitude of observable trends and patterns, which may contribute to better understanding of the prediction and betting markets around the sport. Individual Twitter predictions yield an average accuracy of 53% for all match results in the 2018-19 Premier League season. Moreover, it is shown that it is possible to aggregate Twitter predictions into a model that performs marginally better than The Guardian's pre-match odds for predicting true results. This indicates the strength of the wisdom of the crowd effect.**

I. Introduction

A. Football Betting and Prediction

Football is an exceptionally popular sport with an immense following: an estimated half the world's population in the case of the 2018 FIFA World Cup [1]. The English Premier League – the focus of this project – has enjoyed steady viewership numbers of around 12 million worldwide [2]. Such strong a interest in sport generates a great deal of hype around betting and prediction activities, which are considered by many to be as engaging as the matches themselves.

The date of commencement of football betting is not fully known, but the first instances are understood to have occurred either in the streets or in criminal safe houses. The first recorded betting pools date back to 1923 [3], and were organised Littlewoods Pools, which later grew to become the largest private company in Europe [4]. Since then, betting has changed greatly and is now primarily conducted online. For modern bookmakers and punters, the task of computing and evaluating odds is a complex one, often demanding deep knowledge of the betting subject and the prevailing public sentiment. Experts spend years researching football teams, their performance and news about individual players yet they still often fail to generate accurate predictions.

With the rise of social networks, the public sharing of predictions has become a ubiquitous activity [5]. Entire communities have formed around sports betting and football betting in particular, with technically-minded enthusiasts employing countless metrics and odds calculation methods. Elsewhere, thousands of more informal scoreline and result predictions are shared by fans and pundits alike in the lead-up to every match, most notably on Twitter, where virtually all content is publicly-visible. It is the latter form of prediction that is the primary focus of this project.

B. Social Media for Prediction

Social media are virtual platforms that allow for user-generated content to be shared and distributed at next to no cost. The explosion in their popularity has enabled the aggregation of opinions and wisdom of the crowd that allows us to discover the underlying patterns that may provide valuable information for the betting market.

This perspective is built on the concept of the *wisdom of the crowd* [6]. The idea is that whilst individuals may not be good at estimating, the collective crowd can be more reliable [7]. Predictions are made by a wide variety of people ranging from experts to casual viewers, resulting in potentially massive data sets that can be used to derive the expected result of a game. Thus we are undertaking this project with the hopes of gaining valuable insights into football predictions.

C. Aims and Objectives

This project aims to investigate the world of football predictions, through the lens of fans and experts alike. In aggregating

data from these sources, we hope to uncover bias and incentives that may impact the betting odds. The current season (2018-19) of the English Premier League was chosen to be our domain of enquiry, as this competition features a limited range of teams but a large number of matches, and since all the teams are based in England and Wales there is a significant quantity of English-language commentary available online.

Specific questions we would like to answer include:

- What is the overall prediction accuracy on a game-by-game basis?
- How closely aligned are the betting odds from various sources to the actual results?
- How do teams' performances compare with fans expectation?
- How does bias influence predictions?
- How to replicate the construction of the league table?

The global market capitalisation for sports betting is 250bn [2], with football as the single largest sport. This implies that there is significant monetary value in the accurate prediction of match results, and in the understanding of the biases present in the predictions of others. If the models in this report can successfully anticipate match results, there is a clear monetary benefit. However, should they fail to produce accurate predictions, it is equally interesting to analyse what makes public opinion inadequate.

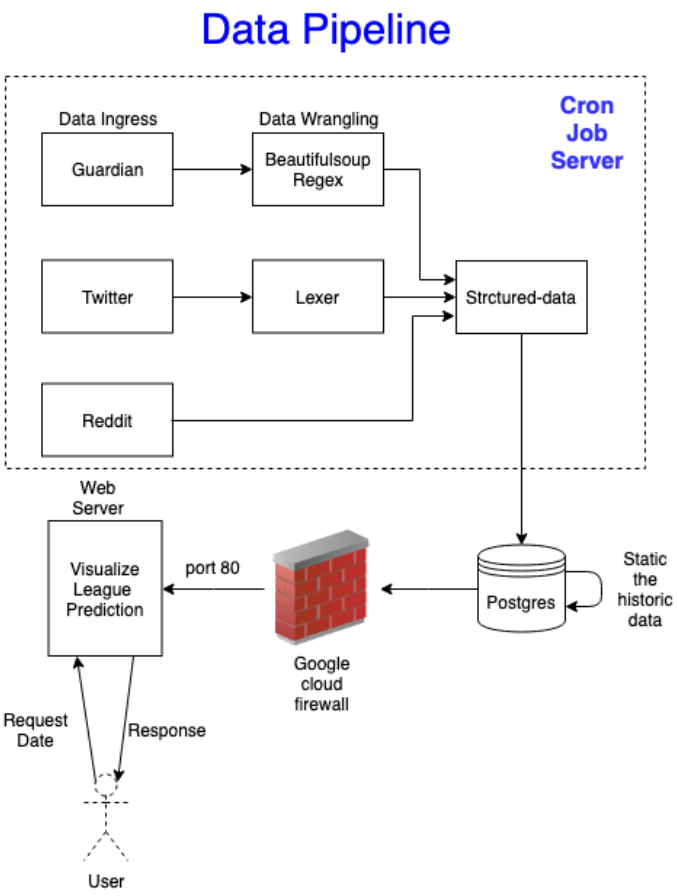

Fig. 1: Data pipeline constructed for this project

II. Solution Summary

Figure1 depicts the data pipeline that we built for this project. This pipeline can be divided into five parts. Firstly, in the data ingress stage, we used APIs and data scraping libraries to download public posts from both the Twitter and Reddit social media platforms, and also wrote a crawler to obtain fixtures, results and pre-match odds from The Guardian. Secondly, a lexical analyser (lexer) was used to perform data wrangling on the natural-language social media posts, obtaining concrete match predictions. After wrangling, the data is integrated and and stored in a PostgreSQL database. This database contains a number of tables, providing multiple views on the dataset as required for the various analysis stages. The final stage of the system is a simple web application which visualises one aspect of our analysis, namely the reconstruction of the Premier League table from individual match predictions.

III. Data Ingress and Storage

A. Predictions from Twitter

Twitter was chosen to be one of our data sources as it continues to rank as one of the leading social networks worldwide based on active users [8], and particularly plays host to vibrant sporting discussion. To obtain user predictions, the Python package Beautiful Soup was used to create custom-made queries to obtain relevant tweets dating back to the beginning of the Premier League season. Football teams have an ever-growing list of nicknames and aliases given to them by fans, and these queries had to factor them in order to capture the full diversity of predictions. In addition, people use a variety of phrases to indicate that they are making a prediction. Using the OR command built into the Twitter search tool, it is possible to select multiple keywords in this way without sacrificing specificity. Below is an example query for one Premier League team:

```
("@Wolves" OR "Wolves" OR "Wolverhampton"
OR "wolverhamptonwanderers" OR
"Wolverhampton Wanderers" OR "WOL")
(predictions OR "I predict" OR "we predict")
-RT
```

The `-RT` command ensures that no retweets are included in the list of results. Some tweets found by the above query are shown in fig. 2.

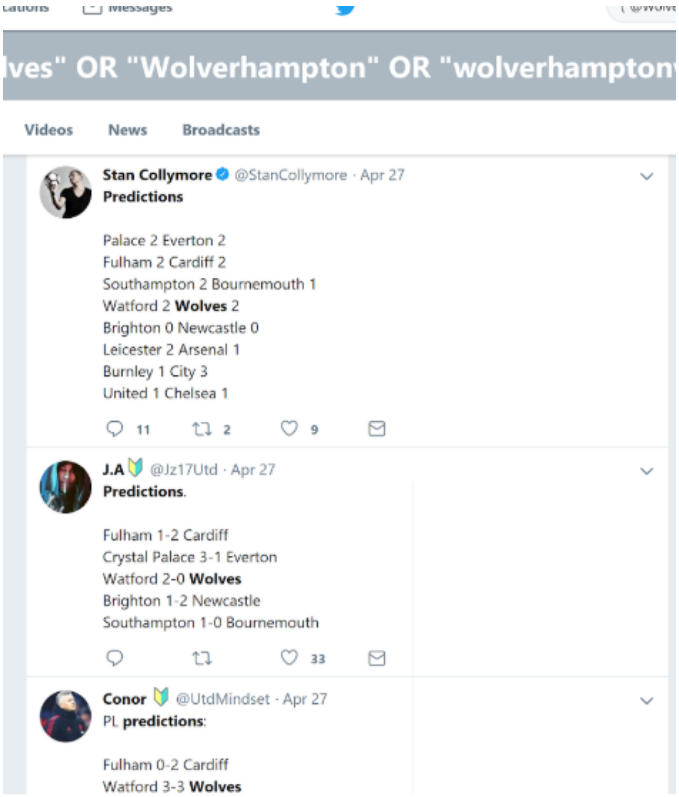

Fig. 2: Results from a query relating to Wolverhampton Wanderers

B. Predictions from Reddit

Reddit was also chosen are a prediction source, due to its having whole communities dedicated to certain topics. In this case, content was taken exclusively from the subreddit `r/Soccerbetting` due to the specific nature of discussions on it, which we hoped would limit the amount of irrelevant content. All top-level comments from "Daily Picks" threads since the start of the Premier League season were extracted using the Reddit API via the PYTHON REDDIT API WRAPPER library. The thread is used only for game predictions. This process yielded $6,000$ results, some of which are shown in fig. 3.

C. Fixtures, Results and Odds from The Guardian

Finally, we obtained match dates, results and pre-match odds from The Guardian's sports website, which has an article for each Premier League game. We chose this data source because all three information types are presented on a single page, inside a unified table, which reduced the effort and time required for data collection. The Guardian provides an API, but some content is not accessible using this tool. For this reason, we analysed the Document Object Model (DOM) of each webpage to create an efficient algorithm to obtain the data we need. We used several Python libraries to build a crawler including BEAUTIFULSOUP4, REQUESTS and REGEX. For an

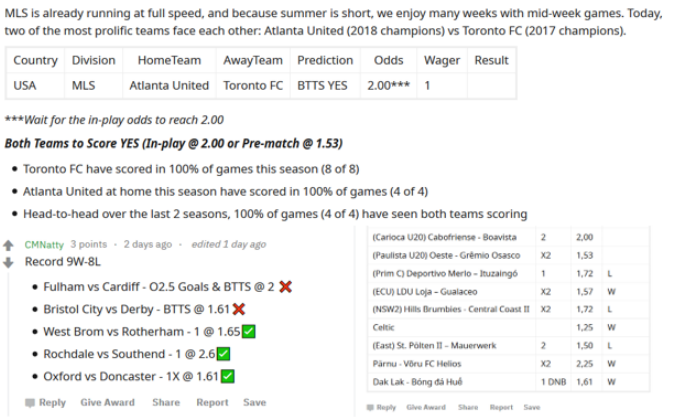

Fig. 3: A selection of Reddit comments predicting football results. These examples come from a largely-unfiltered dataset, and so concern matches both within and outside of the Premier League.

unknown reason, articles were not available for all matches; 235 could be obtained out of the 299 matches that had taken place at the time of data collection.

D. PostgreSQL Database

In order to store all of the the data that we had collected, we used PostgreSQL as our database since much of the content comes from APIs in JSON format, and PostgreSQL RDBMS database which can store the JSON type data directly. We set up our database on Google Cloud, allowing real-time data collection and access without the need for manual synchronisation. Moreover, we used a Docker container to build our local environment. This meant that when we deployed aspects of our system as a web application, we did not need to consider the operating system or revise the online data.

In order to have a rule to manage our table, we used Object Relational Mapping (ORM). There are two reasons for this. Firstly, using ORM helped us to read the table schema without login to database. Secondly, whenever we changed a schema, ORM made the data wrangling process easier.

IV. DATA WRANGLING AND FUSION

A. Twitter Data Wrangling

Natural language data is challenging to analyse numerically without first extracting information from it. It can contain inconsistent language structures, spelling and syntax errors, and subtext, making the true meaning of a text unclear. Twitter is especially prone to inconsistency as well as poor grammar and spelling, therefore a flexible lexer capable of extracting data is required.

The 140 character limit imposed by Twitter often contributes to some of the grammatical errors and contractions of particularly long words, although the limit does provide one benefit [9].

It compels the author to be as concise as possible. If a user wishes to make predictions about every upcoming match for the next 7 days, they must compress these to one of three different formats.

- Pattern 1 - TEAM, DIGIT, TEAM, DIGIT
- Pattern 2 - TEAM, DIGIT, DIGIT, TEAM
- Pattern 3 - TEAM, TEAM, DIGIT, DIGIT

This structure enables the information contained within the Tweets to be extracted with relative ease. To achieve this, the SPACY Python library was used. SPACY is a powerful natural language processing module that will parse text in search of particular patterns, such as those shown above. Whenever a pattern was located, the underlying data was extracted and stored for further preprocessing. This module also allows users to train classifiers that recognise particular words and assign them a label, referred to as a Named Entity Recogniser (NER). We trained an NER to recognise football teams by a variety of different monikers. For instance, Southampton FC is often known on Twitter as one of SAINTSFC, SOTON and @SOUTHAMPTONFC, so whenever the NER identified the use of one of these nicknames, it was replaced with the full name of the team. An example of the NER is shown in fig. 4.

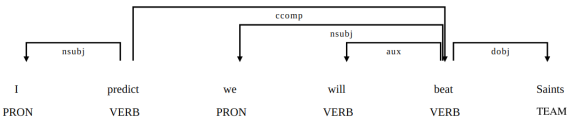

Fig. 4: A figure showing a SpaCy visualisation of various parts of speech and dependencies in an arbitrary sentence.

In this figure, it can be seen that "Saints" is identified by SpaCy as a "TEAM", as well as the various linguistic dependencies in the sentence. However, fig. 4 shows a new pattern that must be addressed.

A further NER was trained to recognise various verbs indicating whether the Twitter user was predicting a team to win or to lose. Often, such a prediction does not contain a score and so was stored as merely a relative ranking between the two teams, to be used in section VI-E. This is shown in Pattern 4. Pattern 5 seeks to capture predictions in which a user refers to the team they support as "we". If a Tweet matches Pattern 5, we inspected the Twitter user's public biography. If this contains a single Premier League team, it was assumed that this is the team the user is affiliated with and so the standard team name replaces "we". Otherwise, the prediction was disregarded.

- Pattern 4 - W/LVERB, TEAM, TEAM, DIGIT?, DIGIT?
- Pattern 5 - "WE", W/LVERB, TEAM, DIGIT?, DIGIT?

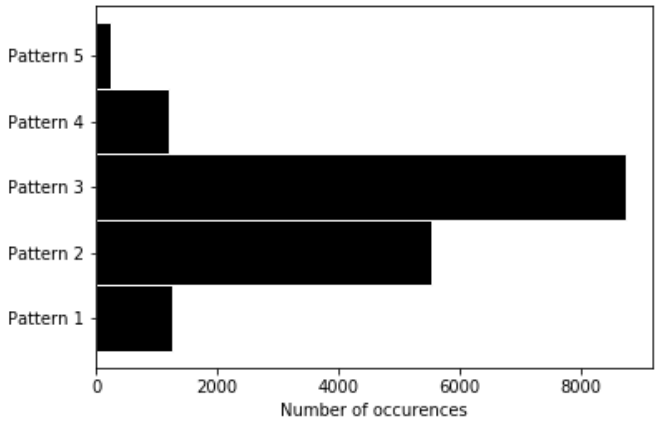

Fig. 5: Graph showing the number of each pattern in the dataset of $16,844$ predictions, extracted from the complete dataset.

Care was taken to ensure that no pattern overlaps with another and that the data was extracted in the order above, such that the first three patterns were located and extracted from the dataset, then the final two were extracted. These precautions prevented a Tweet that reads "Arsenal 3 2 Bournemouth Cardiff 2 4 Liverpool" matching as Pattern 2 (Arsenal 3 2 Bournemouth), Pattern 3 (Bournemouth Cardiff 2 4), Pattern 2 (Cardiff 2 4 Liverpool), and instead correctly being recorded as 2 occurrences of Pattern 2.

Using these patterns, data can be extracted from Twitter, however, much of is not useful for predictions. For instance, many Tweets predicted implausible scorelines, such as "0-12", therefore, any Tweet containing a prediction of more than 6 goals for one team was excluded from the dataset.

fig. 5 shows the number of different patterns within the dataset. It can be seen that the modal pattern is Pattern 3, while Pattern 5 only occurs 251 times in the dataset of $16,884$ predictions extracted from the dataset of $53,293$. This reveals a preference in the population toward presenting predictions in the same format as Pattern 3.

B. Reddit Data Wrangling

We attempted to build a similar lexer algorithm to parse predictions from Reddit comments. However, it became apparent that these contain a far greater diversity of prediction patterns than exist in the tweets, some of which are visible in fig. 3.

Parsing this content would necessitate a more complex lexer architecture. Additionally, rather than simply predicting scores or results as on Twitter, Reddit users tended to express their beliefs as odds in various formats (American, Europe and and decimal), so it was likely to have proven difficult to integrate the two data sources into a common analysis pipeline. For these reasons, we decided to stop pursuing the Reddit predictions. Since the size of this dataset was far smaller than

that obtained from Twitter, this is not considered a major loss to the project.

C. The Guardian Data Wrangling

Data from webpages is typically semi-structured in a DOM tree format [10]. Therefore, we needed to pre-process the data obtained from The Guardian. Primarily, we used the BEAUTIFULTSOUP Python library to search the Tag, Class in the DOM tree. However, each Tag still contained a large amount of unnecessary information. In order to get the specific results and odds information in the Tag we used Regex to find keyword patterns, as shown in Figure6. Moreover, we needed to transform match dates from text format to time object format. After extracting all necessary information for each match, we stored an entry in our database.

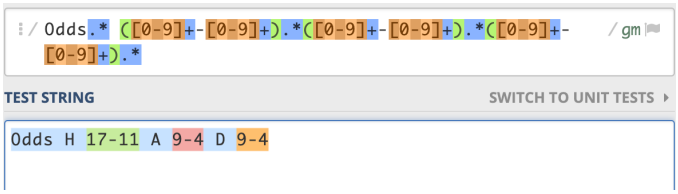

Fig. 6: Using Regex to extract odds from The Guardian content.

D. Timestamp-based Fusion

After wrangling the datasets from both Twitter and The Guardian, we were able to perform the data fusion process to integrate them into a single structure. The DATETIME Python module played a crucial role here, as it allowed us to correspond every prediction to a particular Premier League match based on the date of creation of the containing tweet. To account for people making predictions several days in advance, we allowed for a 7 day search window for predictions made prior to a match. The fusion process yielded $16,884$ predictions, linked to to 299 matches.

Predictions and true results were combined in JSON format, which is extremely well-adapted to the Python working environment, since its structure bears similarity to the dictionary data structure in Python. An example of one match, and the first Twitter prediction collected for that match, is shown in fig. 7.

V. Data Exploration

The unified JSON data structure contained all required information for exploration and analysis work, in a form that could easily be loaded into Python functions and further restructured depending on the specific analysis viewpoint (e.g. focusing on individual matches, teams, or Twitter user affiliations). At various points, we found it prudent to discuss predictions, and their relationships with the match odds and true results, in terms of several alternative metrics:


```
{   ((Fulham,Wolverhampton Wanderers),2018-12-26,(1,1)): [
    {   "user": "Bob_Smith",
        "tweet_time": "2018-12-26 12:33:47",
        "likes": 0,
        "retweets": 1,
        "sentiment": {"neg": 0.0,"neu": 0.887,"pos": 0.113,"compound": 0.4215},
        "score": (1,2)
        "allegiance": "Liverpool" },

    {   "user": "Sally_Jones",
        ...
```


Fig. 7: Unified JSON data structure containing both predictions and true results.

- *Match scoreline*: goals scored by each of the two teams in a given match.
- *Team goal difference*: goals scored minus goals conceded for a single team in a given match.
- *Match result*: either home win, draw or away win.
- *Team result*: either win, draw or loss (independent of whether the team is playing home or away).

It was important not to inadvertently conflate any two of these during analysis. We will not discuss further details of the data exploration methodology here, instead mentioning the important points in parallel with the presentation of results in the following section.

VI. Results

A. Match-Level Analysis

The first target of our investigation was the overall per-match prediction accuracy. We chose to explore this in a way that would retain the maximum information: visualising the distribution of match scoreline predictions as a heat map. For the vast majority of matches, the score predictions were found to be well-approximated by a 2D Guassian, so distributions were fitted using the SCIPY.STATS Python module. The Gaussian distributions served as surrogates for modelling the likelihood of all scores (of up to 6 goals per team), though it is important to acknowledge that such models were not perfect since they operate on continuous data while scores are discrete. Heat maps and fitted Gaussians for a selection of matches are shown in fig. 8. The true match scorelines are plotted in blue.

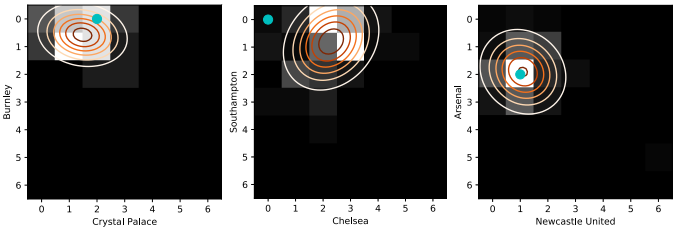

Fig. 8: Scoreline prediction distributions for three matches.

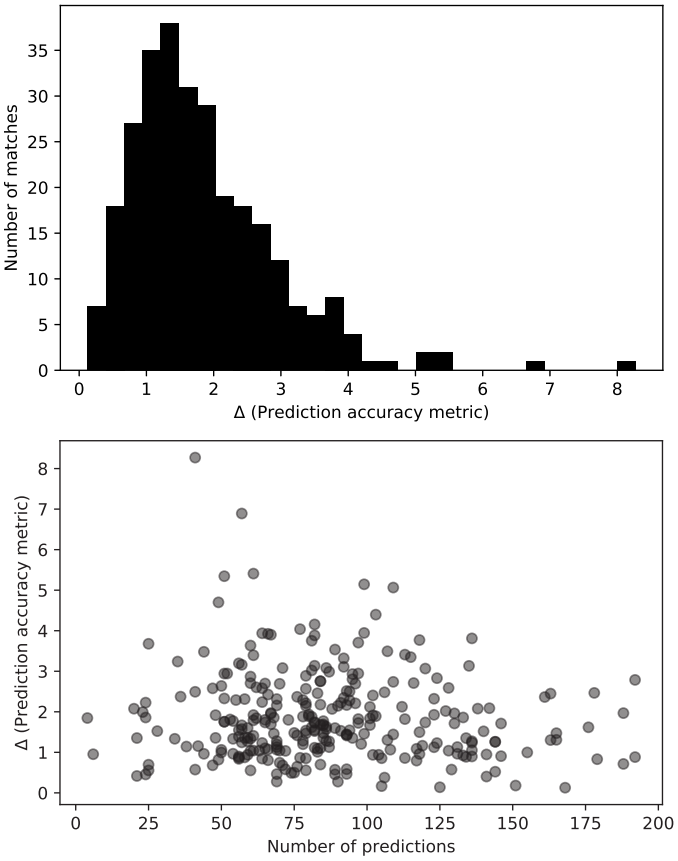

Fig. 9: Distribution of Δ values for all 299 matches in the dataset, and plot relating these values to the number of predictions made about each match.

It is clear from visual inspection the accuracy with respect to the true scoreline varies between matches. We sought to quantify this phenomenon in a metric Δ by taking the distance between the Gaussian mean μ and the true scoreline s in each dimension, scaling by the corresponding component of the standard deviation σ, and computing the L^2 norm. Note that **smaller** values of Δ indicate **better** predictions.

$$\Delta = \sqrt{\left(\frac{\mu_{home} - s_{home}}{\sigma_{home}}\right)^2 + \left(\frac{\mu_{away} - s_{away}}{\sigma_{away}}\right)^2}$$

According to this metric, the single best-predicted match was BURNLEY 1-1 SOUTHAMPTON ($\Delta = 0.13$, 168 tweets), while the worst-predicted was WEST HAM UNITED 4-3 HUDDERSFIELD ($\Delta = 8.27$, 41 tweets). The full distribution is shown in fig. 9. Also shown is a plot relating Δ to the number of predictions made about each match. While no incontrovertible relationship is visible, those matches that received the most predictions all have comparatively low Δ values. This may be evidence of a wisdom of the crowd effect.

B. Comparison to Odds

An advantage of fitting Gaussians to the prediction sets was that they provided a mechanism for inferring match result odds: integrating the distribution for each match across the regions corresponding to a home win, away win and draw. This yielded a probability distribution $P_T \in [0, 1]^3$.

A comparable format had to be obtained from the 235 Guardian pre-match articles. As on most UK betting websites, odds on The Guardian are given in a fractional form: a/b. For every b units wagered, a units will be won if the prediction is correct. The conversion into a probability distribution P_g was achieved by the formula $P_G = \frac{a}{a+b}$. This gave us a full list of football games and their associated odds from two separate sources. The next logical step was to perform a mathematically-rigorous comparison between these values and the true results.

The *Jensen-Shannon divergence* (JSD) is a non-parametric method of measuring the statistical similarity between two probability distributions. It is a smoothed and symmetric version of the Kullback-Leibler (KL) divergence. For two distributions P and Q, the JSD between them is computed via:

$$JSD(P||Q) = \frac{1}{2}D(P||M) + \frac{1}{2}D(Q||M)$$

where $M = \frac{1}{2}(P + Q)$ and D is the KL divergence:

$$D_{KL}(P||Q) = \int_{\infty}^{\infty} p(x) log(\frac{p(x)}{q(x)}) dx$$

We represented each game's true result as a one-hot vector (e.g. $[1, 0, 0]$ for home win, $[0, 1, 0]$ for draw), and computed the JSD between this and the corresponding P_T and P_G distribution. Heat maps for all 235 games for which odds were available are shown in fig. 10. Note that in these graphics, 1 minus the JSD is plotted so that the colours represent the degree of agreement.

There is a visible correlation between the two sets of JSD values, indicating that some matches are inherently more predictable than others, both in terms of odds and casual predictions. Between the two prediction sources, $1 - JSD = 0.973$.

However, an important finding is that Twitter predictions are marginally more predictive of true Premier League results than are the odds listed on The Guardian. The $1 - JSD$ value for Twitter is 0.765 compared with 0.752 for the odds. This result suggests that it may be possible to construct a profitable betting strategy by aggregating predictions that have been freely shared by others on Twitter. It must be noted that the standard deviation of the divergence values are 0.181 for Twitter and 0.122 for the odds, showing that the accuracy of Twitter predictions is somewhat more inconsistent than odds, so such a betting strategy would carry significant risk.

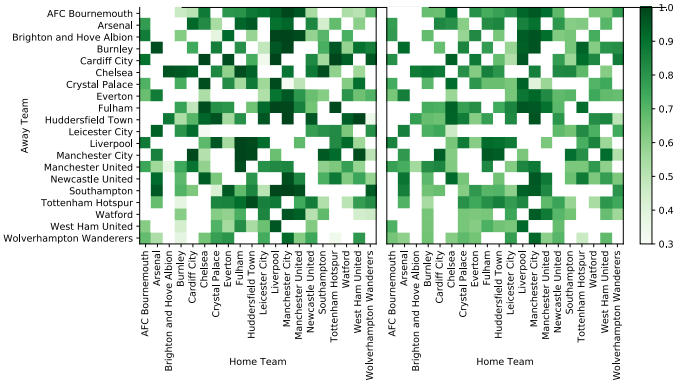

Fig. 10: Left: heat map of $1-JSD$ between Twitter predictions and true result vectors. Right: heat map of $1-JSD$ between odds from The Guardian and true result vectors. In both plots, white cells indicate games that are missing from the dataset, or are yet to take place.

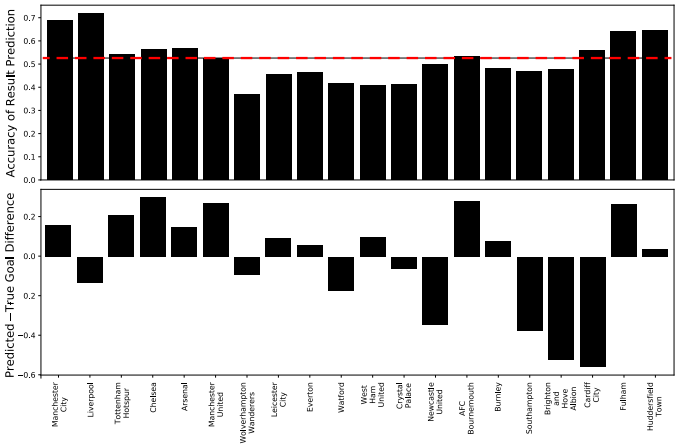

Fig. 11: Top: mean result accuracy for all Premier League teams in league table order. Global mean result accuracy of 52.6% plotted as a dotted red line. Bottom: meal goal difference error for all Premier League teams in league table order. Note: All teams received at least 1500 predictions.

It is also interesting to observe which matches are most accurately forecasted by both prediction sources, namely the home matches of the league's top-performing teams (e.g. Liverpool, Manchester City) and encounters between these teams and those at the opposite end of the table. In such matches, the dominant team very often wins.

C. Team-Level Analysis

For a football team seeking to place highly in the league table come the end of the season, what matters is not precise scorelines, but rather whether they win, draw or lose each match, and the goal difference they accumulate whilst doing so. Therefore, for team-level analysis, we primarily used the metrics of team result and team goal difference.

We first investigated the overall predictive accuracy of tweets about each team across all matches. fig. 11 shows the mean result accuracy (*what proportion of predictions correctly forecast the team result?*) and mean goal difference error (*how did the predictions err on the team goal difference?*) for all 20 teams, ordered left-to-right by their positions in the Premier League table as of 30/04/19.

In the result accuracy graph, a clear trend is visible: mid-table teams are less well predicted than those at either end. The four most predictable teams – and the only ones with accuracy above 60% – are the top two and bottom two in the league. Result predictions about ten mid-table teams are correct less than half the time, and for Wolverhampton Wanderers, the accuracy is a remarkable 37.0%, just 3.7% higher than would be attained through uniform random prediction.

In the goal difference error graph, there appears to be a tendency to over-predict the goal difference of strong teams (as is the case for five of the top six), and under-predict for weak teams (notwithstanding the bottom two). Despite low result prediction accuracies, mid-table teams see small mean goal

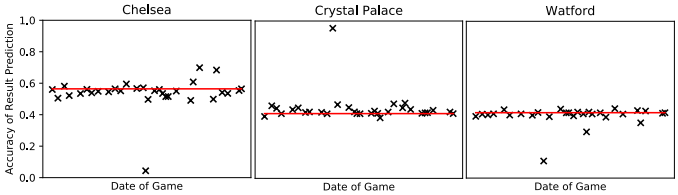

Fig. 12: Per-match result prediction accuracy for three teams, plotted as a function of match date. Mean accuracy for each team (as per fig. 11 plotted as a red line.

difference errors, which points to a compelling conclusion: top and bottom teams are correctly predicted to win and lose respectively, but generally by bigger margins than occur in reality. For intermediate teams, the many inaccurate predictions are equally likely to be over-estimations as under-estimations, so approximately cancel out over the season.

We then integrated temporal information into the analysis, by plotting result prediction accuracy against match date for each team. Results for three teams are shown in fig. 12.

For most teams, the result prediction accuracy is remarkably consistent over the course of the season aside from a handful of outliers. It may be reasonable to consider this value a kind of *property* of a football team, indicating their degree of erraticism.

Similar plots were created to assess temporal variations in goal difference error. These have far higher match-to-match variability, so care had to be taken not to seek trends where none existed. However, for several teams, there are discernible patterns in the data. Four examples are shown in fig. 13.

These trends may be tentatively interpreted as follows. Everton have been through several phases of over- and under-performance relative to expectation, while predictions about

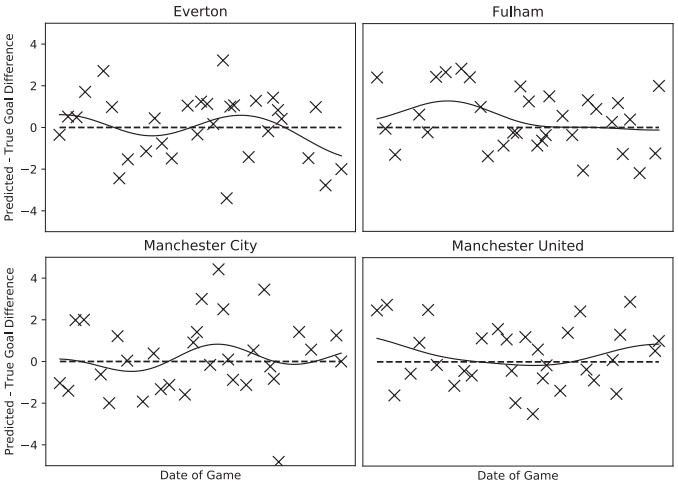

Fig. 13: Four teams with visible temporal trends in the goal difference error metric. Curves fitted using Gaussian process regression (RBF kernel); maximum likelihood solution shown.

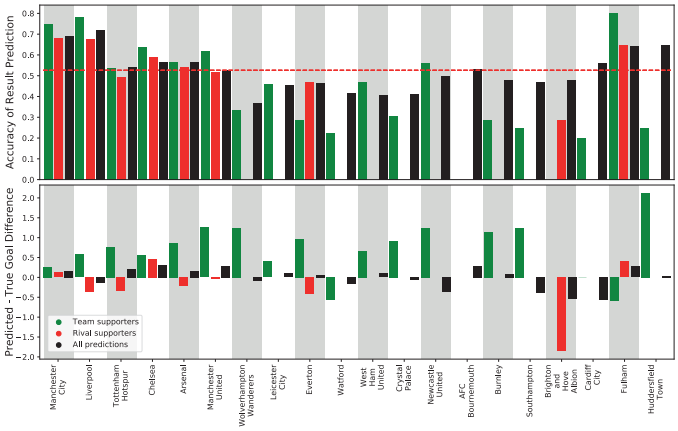

Fig. 14: Top: mean result accuracy, separated by affiliation, for all Premier League teams in league table order. Global mean result accuracy of 52.6% plotted as a dotted red line. Bottom: meal goal difference error, separated by affiliation, for all Premier League teams in league table order.

Fulham consistently over-estimated their goal difference during the first half of the season, before equilibriating to a more accurate average in the second half (this team has had a poor season and is guaranteed relegation). For Manchester City, goal difference errors have centred around zero aside from a mid-season surge (coinciding with a series of losses in December), and predictions about Manchester United have a U-profile, with over-estimation towards the start and end, and performance only meeting expectation in mid-season (the arrival time of new manager Ole Gunnar Solskjaer).

D. Bias and Popularity Analysis

We then sought to assess the prevalence of various biases in the dataset, which would reveal information about the reasoning of football supporters, as well as the relationship between a tweet's popularity and its predictive accuracy.

First, we made use of the team affiliations gathered from Twitter profiles to assess whether result accuracy and goal difference error change when a person makes a prediction about either their own team or a close rival. Rivalry relationships were defined according to the information on the Wikipedia page *List of sports rivalries in the United Kingdom* [11], since in this particular context, this source was deemed likely to be relatively up-to-date and authoritative. fig. 14 presents the results, in the same league-ordered format as fig. 11. A minimum of five predictions was set for each category, hence the absence of some bars.

Of the 19 teams for which sufficient supporter predictions are available, eight receive more accurate result predictions from their own fans than from the general population, four of which are in the top six (who also have the largest fanbases and most supporter predictions). Of the nine teams with sufficient rival predictions, three are better predicted by those rivals. Most interestingly, supporters of eight mid- and low-table are so poor at predicting their own team's result that they would attain higher accuracy through uniform random guessing.

Supporter biases are clearer in the goal difference error metric. Of the 19 teams with sufficient data, 17 sets of fans over-predict their own team's goal difference on average, in six cases by more than a goal per match. For bottom-placed Huddersfield, the value is in excess of two goals per match which, when combined with the low result prediction accuracy, suggests this team's fanbase has been consistently engaged in near-heroic levels of wishful thinking. Six of the nine sets of rival fans under-predict their rivals' goal difference, but the error magnitudes are generally smaller, indicating that the negative bias induced by rivalry is less severe than the positive bias towards one's own team.

Despite both metrics ultimately being functions of goals scored on the football field, the two graphs appear to be relatively uncorrelated. This is likely an artefact of the fact that a single goal is sufficient to flip the result of a match, so small goal difference errors can accumulate to large result errors.

The second type of bias analysed was that towards or against the three possible match results – home win, away win and draw. The results are shown in fig. 15.

Home wins are better predicted than away wins, with 64.2% and 53.8% accuracy respectively. This aligns with intuition, since a home win is often viewed as the default result in football, and away victories are more frequently considered upsets. Draws are only correctly predicted 22.8% of the time, revealing a major form of bias. It might naively be hypothesised that fans simply do not *want* a match to end in a draw, as it is less interesting than a definitive result. However, ignoring their correctness, the marginal probabilities of home win, away win and draw predictions are 49.0%, 32.6% and 18.4% respectively, while across the 299-match

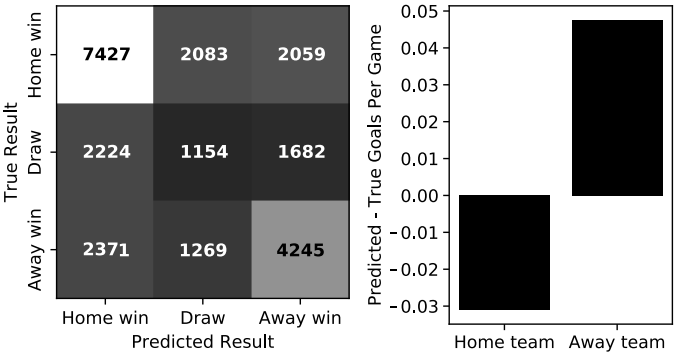

Fig. 15: Left: Confusion matrix of result predictions for all matches. Right: Mean per-match goal difference errors for home and away teams.

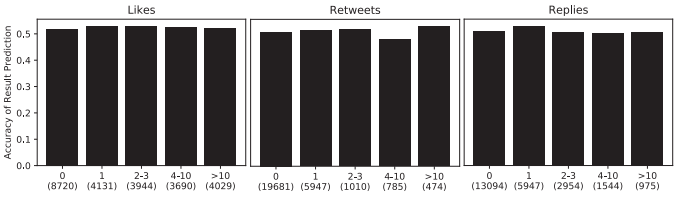

Fig. 16: Histograms relating result prediction accuracy to like, retweet and reply counts. Number of predictions in each bin given in brackets.

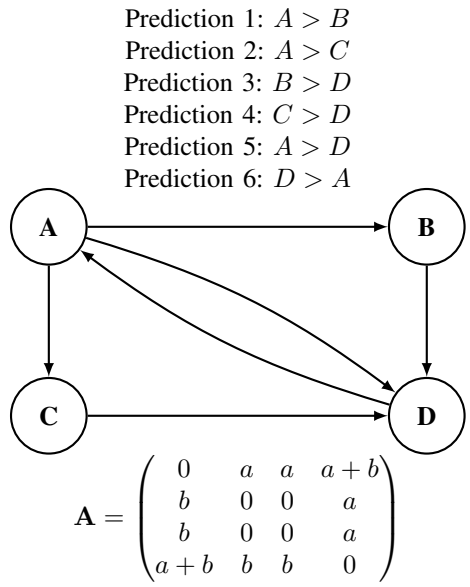

Fig. 17: An example of the Peer Ranking method [12]. Each prediction is translated into a directed link in the graph, with a as the tip-weighting and b as the tail-weighting. The adjacency matrix of this directed graph can be computed, as shown at the bottom.

dataset the true ratios are 49.2%, 33.1% and 17.7%. Football fans' predictions are extremely well-calibrated with respect to the overall frequency of each result, meaning the low draw prediction accuracy must have an alternative cause. This may lie in the dynamics of football itself: draws could be inherently difficult to foresee in advance using the available data.

The second chart in fig. 15 indicates a slight under-estimation of the goals scored by home teams, and over-estimation for away teams, but the magnitudes of these biases are small, both equating to less than one goal in 20 matches.

Finally, we investigated the relationship between result prediction accuracy and the like, retweet and reply counts of the containing tweet. The purpose of this was to evaluate whether users with high engagement in their predictions – more likely to be perceived experts or influencers – exhibit improved predictive performance over those who see little or no attention. Histograms for each variable are shown in fig. 16. It is quite evident that no significant trends exist in these graphs. This indicates that popular Twitter users are no better or worse at predicting results than unpopular ones, and any perceived expertise is therefore likely to be illusory.

E. League Table Reconstruction

It is possible to analyse Twitter predictions to rank each team in the Premier League, reconstructing an approximation of the league table. This can be done using techniques drawn from methods for fairly distributing marks in the assessment of group projects [12]. The Peer Ranking (PR) method describes group members ranking their peers in order of their relative contribution to the overall project.

In the context of this project, the PR method considers a prediction of one team to win against another a pairwise ranking between the two. Every tweet in the database, whether or not it expresses an explicit prediction of the score, contains such a comparison. These predictions can be aggregated to form a hierarchical ranking.

To analyse these predictions, let us consider each pairwise ranking to be a directed link in a graph $G(V, E)$. Let V be the set of all teams in the league and E be the set of predictions between two teams. Each prediction forms a directed link from the predicted loser to the predicted winner as shown in fig. 17. Each link carries with it a weighting, a for the tip of the link and $b = 1 - a$ for the tail.

The 20×20 adjacency matrix of this weighted, directed network is formed, with each entry corresponding to the weighting of links between pairs of teams, with $tr(\mathbf{A}) = 0$ as no team is ranked against itself. Taking $det(\mathbf{A} - \lambda\mathbf{I}) = 0$ yields 20 eigenvalues, the largest of which is referred to as the leading eigenvalue. As in adaptions of the PageRank algorithm [13], the eigenvector that corresponds to the leading eigenvalue reveals the relative importance of each node in the graph and hence can be used to approximate the ranking of each team.

The i^{th} element of the leading eigenvector is related to the

Real Table			Predicted Table		
Place	Team	Points	Team	Points	
1	Man City	92	Liverpool	92	▲ 1
2	Liverpool	91	Man City	78	▼ 1
3	Tottenham	70	Chelsea	75	▲ 1
4	Chelsea	68	Arsenal	68	▲ 1
5	Arsenal	66	Man United	62	▲ 1
6	Man United	65	Tottenham	62	▼ 3
7	Wolves	54	Wolves	54	▶ 0
8	Leicester City	51	Everton	43	▲ 1
9	Everton	50	West Ham	41	▲ 2
10	Watford	50	Leicester City	40	▼ 2
11	West Ham	46	Watford	39	▼ 1
12	Crystal Palace	43	Bournemouth	34	▲ 2
13	Newcastle	42	Burnley FC	32	▲ 2
14	Bournemouth	42	Crystal Palace	31	▼ 2
15	Burnley FC	40	Southampton	31	▲ 1
16	Southampton	38	Newcastle	29	▼ 3
17	Brighton	35	Fulham	26	▲ 2
18	Cardiff City	31	Brighton	26	▼ 1
19	Fulham	26	Cardiff City	21	▼ 1
20	Huddersfield	14	Huddersfield	17	▶ 0

Fig. 18: The predicted table shown in relation to the actual league table as of 27/04/19. Here, $a = 0.99$. The right-most column shows the number of places between the prediction and the actual ranking of each team.

i^{th} team. Therefore, to construct the predicted Premier League table, the list of teams is ordered by the entries in the leading eigenvector. A further prediction can be made for the number of points by scaling the points according to the entries in the leading eigenvector v. This is given by

$$points_{pred,team} = \frac{points_{max}}{v_{max}} \times v_{team}.$$

For a simple example, consider a league of 4 teams. If the leading eigenvector were $[0.4, 0.3, 0.2, 0.1]$, and the top team, Team A, had 20 points, then the points would be $[20, 15, 10, 5]$.

The results of this can be seen in fig. 18.

It is apparent that this table is very rarely precisely correct. However, it is equally rarely very wrong. The maximum number of places it is wrong by is never greater than 3. One such team is Newcastle. This team is ranked 3 places lower than it performs in reality, possibly due to the negative sentiment toward the owner of the club biasing the predictions against the club. However, this is only one of many possibilities.

The accuracy of our table prediction can be contextualised by considering how it compares to guessing at random. The sum of positional errors in the table in fig. 18 is 28. A simple simulation consisting of randomly permuting the digits 1 to 20 for 50 million iterations yielded no instances where the summed error was the same or lower.

VII. Web Application Deployment

Given the uniqueness and practical salience of this aspect of our analysis, we chose to build a web application to dynamically visualise the reconstructed Premier League table over the course of the season, as produced by the method in the previous section. A combination of the Django Python library and React were used to create this application. First, a prediction model was created with 6 fields:

- tweet_time
- team1
- team2
- score1
- score2
- type

Secondly, a Fixture model was created in order to track the true league table over time. This model has 6 fields

- date
- hometeam
- awayteam
- homescore
- awayscore
- victor

To construct both the predicted table and the actual table at any point in time, a query is submitted to the database containing the date desired by the user. This returns two QuerySet's containing all the predictions and fixtures prior to the date in question. To reconstruct the true League Table, each a function iterates through every element in the Fixture QuerySet, assigning three points to the victor, and one point apiece in the case of a draw.

An endpoint is created that will search the database and perform the analysis of section VI-E, returning data as a csv. This endpoint is then used by the landing page of the application to query the database, before rendering the csv data into a table. This is done using D3.js, a Javascript framework designed to facilitate the visualisation of wide varieties of data.

Figure 19 shows landing page of this application, with the table evaluated on the 1st of January 2019. This is a prototype and so is slow to load, however it would be simple (though time-consuming) to address this. We would need to adapt the application to regularly extract more Twitter Data and process it into weekly tables that could be stored in the database. At present, the data is processed after each query to the database, hence the slow loading times for more recent dates. This application is accessible at *http://ads.aecmix.com*.

VIII. Summary of Findings

The following are the key findings of our investigation into Twitter users' predictions of Premier League matches:

- Individual predictions from Twitter users contain signal: 53% get the match result correct.
- Fitting Gaussian distributions predictions for each match enables the construction of probability distributions over

POSITION	TEAM	POINTS	TEAM PREDICTION	POINTS PREDICTION
1	Chelsea	12	Manchester City	12
2	Liverpool	12	Chelsea	11
3	Watford	12	Liverpool	10
4	Manchester City	10	Tottenham Hotspur	9
5	Tottenham Hotspur	9	Arsenal	8
6	AFC Bournemouth	7	Manchester United	8
7	Arsenal	6	Everton	7
8	Everton	6	Wolverhampton Wanderers	7
9	Leicester City	6	Leicester City	6
10	Manchester United	6	Burnley	5
11	Wolverhampton Wanderers	5	Fulham	5
12	Brighton and Hove Albion	4	Watford	4
13	Fulham	4	Newcastle United	4
14	Southampton	4	Crystal Palace	4
15	Crystal Palace	3	West Ham United	3
16	Cardiff City	2	Southampton	3
17	Huddersfield Town	2	AFC Bournemouth	3
18	Burnley	1	Brighton and Hove Albion	2
19	Newcastle United	1	Huddersfield Town	2
20	West Ham United	0	Cardiff City	1

Fig. 19: A screenshot of the table rendered by the application. Here, this table has been evaluated on the 1st of January 2019.

the possible results, which we have found to be more predictive of true outcomes than the odds listed on The Guardian's sports website, according to the metric of Jensen-Shannon divergence.

- Result predictability is highest for the strongest- and weakest-performing teams in the Premier League.
- Some temporal trends are visible in time-series analysis, providing insight into how a team's performance compares with public expectation over time.
- Bias and allegiance is a visible factor in predictive accuracy, particularly in terms of the goal difference metric. Fans over-predict the goal difference of their own team, and under-predict rivals, though the latter is a smaller effect.
- There is no correlation between the online attention that predictions or predictors enjoy, and their level of accuracy. Experts or influencers appear to be no more prescient than fans with no following.
- By treating match predictions as expressions of pairwise ranking, a remarkably accurate reconstruction of the Premier League table is possible.

IX. Conclusion

There are still many prospects to consider for our current, working model. Time constraints aside, there is a lot more room for improvement. For example, now that we have established a data extraction pipeline, a potential implementation would be an additional feature that allows for real-time, live update of the data stream. This would streamline the data ingress process tremendously, allowing us to adapt the model for current football data.

Another path of progress would be to incorporate additional social media platforms for even more perspectives. As it stands, YouTube and Facebook may all be possible venues for further exploration.

Finally, it goes without saying in understanding football predictions, we would hope to derive betting strategies to assess our performance. It would be satisfying to pit our model against the betting market, and most importantly, it may prove to be financially rewarding.

REFERENCES

[1] David G Schwartz. Roll the bones: The history of gambling. 2013.

[2] Contexts Magazine. English soccer's mysterious worldwide popularity.

[3] Littlewoods' John Moores, the father of home shopping, Mar 2010.

[4] "The Littlewoods Organisation PLC.". The littlewoods organisation plc, 2019.

[5] Andrew Perrin. Social media usage: 2005-2015. 2015.

[6] Francis Galton. Vox populi (the wisdom of crowds). *Nature*, 75(7):450–451, 1907.

[7] Christian Wagner and Tom Vinaimont. Evaluating the wisdom of crowds. *Proceedings of Issues in Information Systems*, 11(1):724–732, 2010.

[8] Most popular social networks worldwide as of april 2019, ranked by number of active users (in millions), 2019. [Online; accessed 01-May-2019].

[9] Brian William Locke. Named entity recognition: Adapting to microblogging. 2009.

[10] W. Li, Y. Dong, R. Wang, and H. Tian. Information extraction from semi-structured web page based on dom tree and its application in scientific literature statistical analysis system. In *2009 IITA International Conference on Services Science, Management and Engineering*, pages 124–127, July 2009.

[11] Wikipedia contributors. List of sports rivalries in the united kingdom, 2019. [Online; accessed 01-May-2019].

[12] Hugh Harvey, James Keen, Chester Robinson, James Roff, and Thilo Gross. Quantitative analysis of approaches to group marking. *Assessment & Evaluation in Higher Education*, pages 1–15, 2019.

[13] Peteris Daugulis. A note on a generalization of eigenvector centrality for bipartite graphs and applications. *networks*, 59(2):261–264, 2012.