Who Wins? Utilizing Facebook Comments for Football Match Predictions

Pawit Wangsuekul*
School of Electrical Engineering, KAIST
pawit_w@kaist.ac.kr

Guntitat Sawadwuthikul*
School of Computing, KAIST
guntitats@kaist.ac.kr

Abstract

Predicting a football match has never been an easy task. A wide range of factors from recent match statistics to player injuries collectively decide the winner of the game. Instead of intensive feature engineering, we propose a novel *a posteriori* approach that infers match-related information from public Facebook comments before the match starts. Using *EPC20-21* which we collected and made publicly available, we evaluate our proposal on the dataset, achieving 61.29% accuracy without deep learning and 74.19% with deep learning. Our implementation and the *EPC20-21* dataset are available at https://github.com/heartpawit/epl-predictor.

1 Introduction

For football team improvements, strategic analysis and match prediction become potentially beneficial to gain advantages over the opponent team and to assist the coach by adaptively adjusting the team's strategy (Zhang et al., 2021) and by performing analysis on individual performances (Park et al., 2017; Oytun et al., 2020). The benefits of match prediction do not limit to guiding performance improvement but for entertainment (e.g. gambling and fantasy prediction[1]) as well.

Recent studies utilize different approaches to solve the football match prediction task. The approaches vary from using statistical networks and traditional machine learning techniques (Joseph et al., 2006) to modern deep learning networks (Danisik et al., 2018; Bunker and Thabtah, 2019). In addition, due to a large number of indirect factors, chosen features differ between studies. Match features (Li, 2020), external features such as player characteristics and betting odds (Stübinger et al., 2019; Tax and Yme Joustra, 2015), and news articles (Park et al., 2017) have been alternatively used to improve the prediction model.

Despite the advancement of natural language processing and deep learning techniques, the human brain is arguably more than capable of understanding a language that the human is familiar with. Based on the acquired knowledge, humans can further perform critical analysis and predict probable consequences. Upon this fact, we borrow the human capability in combination with natural language processing to predict a specific outcome. An illustrative task is football match prediction, which becomes the main focus of our study.

In this paper, we propose a novel approach to predict football match results that depends only on Facebook public comments and does not require previous statistics that may include expensive access or require intensive feature engineering. We also publish the *EPC20-21* dataset which contains public Facebook comments in line-up updates from each English Premier League team to the public.

2 Dataset

2.1 Data Collection

To collect insightful opinions from the football fans before the match, we choose to scrape the Facebook posts that notice the starting line-up from each team. The commenters of these posts do not only have prior knowledge about the two teams but also the latest possible updates on the team status.

Facebook posts with comments are scraped by using the facebook_scraper[2] library. The most challenging task is to find the exact line-up update posts for each match. With automated preliminary scraping and some techniques including keyword search and match date comparison, we extract the post IDs of the line-up updates for each match and scrape the post comments respectively.

[*] These authors contributed equally to this work

[1] https://fantasy.premierleague.com/

[2] https://github.com/kevinzg/facebook-scraper

Label	Example Comments
Win	Come on Burnley good starting 11. I would say that this is our strongest team.
Lose	Nelson should have started ahead of Willian. Yeah we buy defenders but we wont play them.

Table 1: A capture from our *EPC20-21* dataset showing a few comment samples and their corresponding labels.

2.2 *EPC20-21* Dataset

The dataset, which is used as the training set, contains public Facebook comments in the line-up updates for both teams from all of the English Premier League 2020-21 matches. Each comment is labeled as either "win", "draw", or "lose", where the label indicates the actual result of that team in the corresponding match.

The comments are scraped from 405 Facebook posts across 16 teams and 301 matches. Each post is limited to a maximum of 150 comments to prevent data imbalance. The total of 53132 comments, categorized according to their labels, consists of 19392, 13352, and 20388 comments labelled with "win", "draw", and "lose", respectively. Example comments are displayed in Table 1.

3 Methodology

We first split the EPC20-21 dataset into two classes depending on the result of the corresponding match. Both groups of comments are treated as a training set and input to the encoder (Section 3.1), which vectorizes the comment text. The encoded vectors are consequently passed to the predictor (Section 3.2) which decides the winning chance of a specific comment. This study experiments on a variety of encoders and predictors as described in the following subsections.

3.1 Encoder

The encoder preprocesses the comment texts into a numerical form, which is more interpretable for the machine. We handpick three commonly used techniques and two pretrained deep neural networks and categorize them into two subtypes: frequency-based and semantic-based encoders.

3.1.1 Frequency-based Encoder

Frequency-based encoder translates the training set based on the frequency distribution of the tokens without considering the semantics behind the text. The three encoders include:

- *Unigram* tokenizes each post into a list of words, then it is converted to a frequency distribution of tokens.
- *Bigram*, similar to *Unigram*, captures pairs of adjacent words in each post, which are later transformed into a frequency distribution.
- Term frequency-inverse document frequency, or *TF-IDF*, compares any specific word of a single comment to the entire training set and vectorizes the comment into a sparse vector according to the relevances of the words.

3.1.2 Semantic-based Encoder

We extend the ability of the frequency-based encoder to not only capture the word distribution but the semantics of each comment. To maximize the performance of semantic encoding, two state-of-the-art deep text encoders are introduced: Universal Sentence Encoder (Cer et al., 2018) and Sentence-BERT (Reimers and Gurevych, 2019), instead of non-deep learning approaches.

3.2 Predictor

The predictor receives the output from the encoder as an input and predicts its label. We propose three predictors, which are the following:

- *Spearman correlation* $S(f, f')$: We compare the unigram/bigram frequency distributions of a post, $F(p)$, with that of the winning set, $F(W)$, and the losing set, $F(L)$, by computing the Spearman rank correlation of each pair and predict the label as the set with higher rank shown in Equation 1.

$$\underset{X=W,L}{\arg\max}\, S(F(p), F(X)) \tag{1}$$

- *Cosine similarity*: For each post p, after each comment is encoded into a vector $\mathbf{c_p}$, we compare the encoded post comments with the encoded set comments ($\mathbf{c_W}$ and $\mathbf{c_L}$) pairwise using mean cosine similarity and predict the label as shown in Equation 2.

$$\underset{X=W,L}{\arg\max}\, \frac{1}{|p||X|} \sum_{\mathbf{c_p} \in p} \sum_{\mathbf{c_x} \in X} \frac{\mathbf{c_p} \cdot \mathbf{c_x}}{\|\mathbf{c_p}\| \|\mathbf{c_x}\|} \tag{2}$$

- *Truncated SVD + Gradient boosting classifier*: We use truncated SVD to decompose the encoded vectors into 128-dimension embeddings to eliminate redundancy. Then, we train

Method	Single Evaluation					Double Evaluation				
	TL	FW	FL	TW	Acc	TL	FW	FL	TW	Acc
TF-IDF / Cosine Similarity	**16**	**15**	**9**	**22**	**61.29**	9	6	7	9	58.06
TF-IDF / TSVD + GBC	30	1	28	3	53.23	13	2	14	2	48.39
Unigram / Spearman	2	29	1	30	51.61	6	9	9	7	41.94
Bigram / Spearman	29	2	26	5	54.84	7	7	8	9	51.61
USE / Cosine Similarity	9	22	3	28	59.68	10	5	5	11	67.74
USE / TSVD + GBC	28	3	27	4	51.61	13	2	12	4	54.84
SentBERT-a / Cosine Similarity	24	7	22	9	53.23	**11**	**4**	**4**	**12**	**74.19**
SentBERT-a / TSVD + GBC	28	3	28	3	50.00	13	2	14	2	48.39
SentBERT-b / Cosine Similarity	20	11	20	11	50.00	10	5	7	9	61.29

Table 2: Performances evaluated on different combinations of encoders and predictors. The metrics are reported in accuracy along with the confusion metrics (TW: true win, FW: false win, TL: true lose, and FL: false lose). The upper group shows feature-based encoders while the lower group includes only semantic-based encoders. Two distinct backbones have been used in Sentence-BERT, namely a) all-MiniLM-L6-v2 and b) stsb-roberta-large.

the gradient boosting classifier, which predicts the label of the input using an ensemble of decision trees.

4 Experiments

Using the encoders and predictors mentioned previously, we compare the prediction performance between nine different combinations of encoders and predictors. Two types of evaluations have been carried out:

- *Single evaluation*: Using the result from the predictor, we predict the match result of each team, post by post, without needing to know about the predicted result of its opponent.
- *Double evaluation*: Since there cannot be two winners or two losers in one match, instead of immediately predicting the match result of each team using only its post, we combine the results from both teams. The team with a higher winning chance will be regarded as the winner and the other as the loser.

After we have the predicted match result, either from a single or double evaluation, we compare it with the ground truth. The results are shown in Table 2.

5 Discussion

The results show the importance of semantics in distinguishing win and lose comments. Due to the high similarity of word frequency distributions, involving the semantics results in a better performance than using the frequency-based encoders. Similarly, bigram captures the surrounding context of each word, yielding a greater accuracy than unigram.

However, using a large number of features does not guarantee performance improvement. In the case of using large language models, it also increases the training time.

The cosine similarity induces less bias towards either win or lose class than using a classifier, as the numbers of predictions are more evenly distributed among win and lose.

The results indicate that with frequency-based encoder, single evaluation performs better than double evaluation. In contrast, double evaluation leads to a better performance in semantic-based methods.

In this study, we exclude all the matches that result in a draw from both training and evaluation, since the draw comments introduce ambiguity to the prediction task. This is partly because the comments are usually positive (win) or negative (lose) rather than neutral, which causes the performance to be significantly worse. Reconsidering the inclusion of the draw comments might be an intereseting task to improve our predictor's capability in the future.

6 Conclusion

Our novel *a posteriori* approach, using only public Facebook comments, shows that it performs better than blind guessing. Consisting of only a pair of encoder and predictor, the proposed framework achieves 61.29% without deep learning and improves to 74.19% with deep learning to help extract the semantics.

References

Rory P. Bunker and Fadi Thabtah. 2019. A machine learning framework for sport result prediction. *Applied Computing and Informatics*, 15(1):27–33.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics.

Norbert Danisik, Peter Lacko, and Michal Farkas. 2018. Football match prediction using players attributes. In *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)*. IEEE.

A. Joseph, N.E. Fenton, and M. Neil. 2006. Predicting football results using bayesian nets and other machine learning techniques. *Knowledge-Based Systems*, 19(7):544–553.

Hang Li. 2020. Analysis on the construction of sports match prediction model using neural network. *Soft Computing*, 24(11):8343–8353.

Musa Oytun, Cevdet Tinazci, Boran Sekeroglu, Caner Acikada, and Hasan Ulas Yavuz. 2020. Performance prediction and evaluation in female handball players using machine learning models. *IEEE Access*, 8:116321–116335.

Young Joon Park, Hyung Seok Kim, Donghwa Kim, Hankyu Lee, Seoung Bum Kim, and Pilsung Kang. 2017. A deep learning-based sports player evaluation model based on game statistics and news articles. *Knowledge-Based Systems*, 138:15–26.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics.

Johannes Stübinger, Benedikt Mangold, and Julian Knoll. 2019. Machine learning in football betting: Prediction of match results based on player characteristics. *Applied Sciences*, 10(1):46.

Niek Tax and Yme Joustra. 2015. Predicting the dutch football competition using public data: A machine learning approach.

Qiyun Zhang, Xuyun Zhang, Hongsheng Hu, Caizhong Li, Yinping Lin, and Rui Ma. 2021. Sports match prediction model for training and exercise using attention-based LSTM network. *Digital Communications and Networks*.