Sentiment Analysis on Tweets about the Ballon d'Or

Joseph Busacca (SUID - 676866487)

Talal Hakki (SUID - 432378419)

Jonathan Furhman (SUID - 992638538)

Austin Johnson (SUID - 3978397812)

Evan Garvey (SUID - 582777089)

Social Media and Data Mining (CIS 400)

Syracuse University

May 2023

Table of Contents

Abstract

The goal of this project is to analyze sentiment among popular soccer players to determine the probability of their chance to win the Ballon d'Or. Twitter was a great place to start, due to its API and ease of use. Functionality was a major point in our decision to analyze tweets from Twitter rather than other social media. This is due to Twitter and its content, because it is based mostly on text and statements from users, rather than other social medias like Instagram which utilize pictures, or YouTube which utilizes video. With experience gained throughout this course, it is simplest (yet effective) to analyze text rather than other forms of media. Required elements to complete this task include the Twitter API, Python packages such as NLTK, and more.

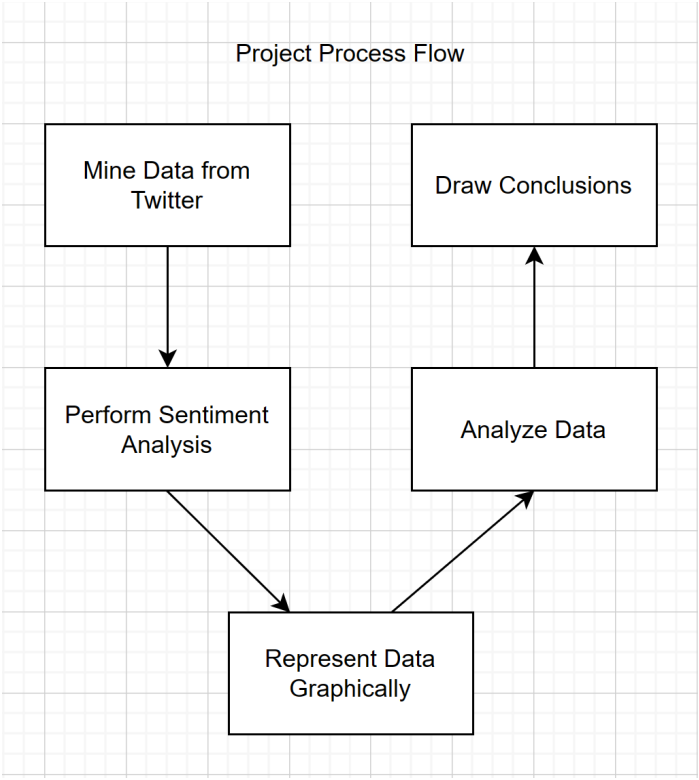
1. Introduction

Twitter and other social media sites have developed into a platform for people to express their thoughts and ideas on a variety of subjects, including sports. One of the most coveted individual awards in football, the Ballon d'Or, always generates a lot of discussion and excitement before the winner is revealed. Twitter is a fantastic opportunity to gather feelings and ideas about potential Ballon d'Or winners in 2023 because of its large user base and constant stream of information.

The goal of this project is to perform a sentiment analysis on a large volume of tweets related to the Ballon d'Or 2023 five top candidates, 300 tweets per candidate to be exact. This way, we will be able to perform an analysis on which candidate people all over the world are supporting, or think they will win the Ballon d'Or, and with all this data, we can make our own predictions.

Overall, this project has the potential to shed light on the public's perception of the top Ballon d'Or 2023 candidates and provide valuable insights into their chances of winning the prestigious award. It will be an exciting opportunity to explore the power of sentiment analysis and showcase the potential of natural language processing techniques in the field of sports.

2. Process Flow



3. All About Data

The Twitter Data API is a set of programming endpoints that can be used to find, retrieve, or engage with, user's tweets. Furthermore, having a twitter developer account, it enables the programmer to gain the power of Twitter's open, global, real-time and historical platform within your own applications. Twitter API makes use of OAuth authentication, which allows the programmer to make API requests on behalf of your Twitter developer App.

First, we will be inputting data to send to the twitter API (What are we looking for? What are the properties of our searches?).
Then, we will proceed with our received data, and perform sentiment analysis on it using the python NTLK library.
Finally, we will utilize MatPlotLib to take our data and visualize our findings.
(obv expand upon this as we go on)

3.1 Calling the API

Below are some necessities for requesting data from this API:

1. Create a Twitter account: To use the Twitter API, you must first create a Twitter account if you don't already have one.
2. Apply for developer account: You need to apply for a developer account from the Twitter Developer Dashboard by filling in the necessary details and agreeing to the terms of service.
3. Create a Twitter app: Once your developer account has been approved, create a Twitter app in the developer dashboard. This involves filling in details such as the app name, description, and website URL.
4. Generate API keys and tokens: After creating the app, Twitter will provide you with API keys and tokens that allow you to access the Twitter API. the API keys consist of 4 parts:
    a. consumer key
    b. consumer secret
    c. access token
    d. access token secret.
5. Choose an API endpoint: The Twitter API has several endpoints that allow you to access different types of data, such as tweets, user information, and search results. Choose the endpoint that you want to use to request data.
6. Construct API request: To request data from the Twitter API, you need to construct an HTTP request that specifies the API endpoint, parameters, and authentication credentials.

3.2 More about the API

Twitter API has a rate limit in place to maintain stability and reliability of the API for all users. The limits vary depending on the type of endpoint and authentication. For example, a standard user-authentication could make up to 900 requests every 15 minutes, while an application authenticated with a bearer token could make up to 5000 requests every 15 minutes.

3.3 API Methods Used

3.3.1 Search Tweets

This method is used to retrieve the most popular tweets for each candidate from Twitter's advanced search functionality. This API method takes in a query search term and tweets count as input. We use this method to query each item in the list of search terms for each candidate to extract. The tweets are fetched based on the most recent tweets and sorted from most popular to least popular (by retweets). Now that we have the metadata of tweets, we can extract the tweet text for the current and next search results by index splicing using the statuses field. To extract an adequate number of tweets, this method is used to extract 300 tweets for each of the 5 players -

totaling 1500 tweets. By default, this method retrieves 100 tweets per function call, so this is accomplished by iteratively calling this function based on the next search results with varying queries until the tweets count is achieved.

3.3.2 Retweets

Upon searching for the tweets, we need to extract the number of retweets for each tweet. This will aid the analysis of tweets to determine the most likely candidate to win the Ballon d'Or. Retweets are taken into account to predict the winner because the number of retweets a Twitter post has correlates to the likelihood of the mentioned player winning - this could be a positive, negative, or neutral standpoint. Whereby the number of retweets for each tweet is stored to ensure that the prediction accuracy is maximized. By default, the search function retrieves retweets which are extracted from the tweet metadata dictionary that will add to the total number of tweets collected.

3.3.3 Unquote Tweets

We use this function to convert the next search results to clean the Twitter text into formatted readable text. Using this we can create and store the cleaned tweets into a dictionary that all have the same format. This simplifies the extraction of Twitter text to assemble keyword arguments.

3.4 Where does the data go?

The data - tweets, retweets, and tweet count - collected for each candidate is stored in a text file, where the tweets are sorted in order of the most recent for each candidate. Using a text file to store all tweets fetched allows new tweets to be added, modified, and removed in a simple manner. This is accomplished by opening a file for writing before gathering the tweets, where batches of 100 tweets are written to the text file at a time. This will eliminate the loss of tweets gathered upon stopping the program while tweets are extracted. Moreover, this aids the extraction of tweets for analysis and modification due to the simplicity of plain text formats that allow for rewriting and appending to the file. Furthermore, to ensure that rate limit expectations are handled, the program will sleep when the rate limit is reached and will resume the fetching of tweets when the rate limit is reset.

4. Sentiment Analysis

To analyze sentiment among the candidates, the Natural Language Toolkit (NLTK) package was utilized to produce results. Some preliminary changes on the tweets need to be done first; the NLTK library works best when sentences are put through several filters to remove extraneous or redundant words.

4.1 Preprocessing

The first of the preprocessing tasks that needs to be accomplished is tokenizing, or splitting, the tweets into their individual words. From there, basic filtering is accomplished by removing ‘stopwords’ - commonly used and non-emotional words that can easily be ignored by the toolkit, such as ‘a’, ‘an’, ‘of’, or ‘in’. After the stopwords are removed, a lemmatization filter is applied to the remaining words. Lemmatization involves taking several synonymous terms, or the same term in different tenses, and resolving them as a more basic, and thus understandable by the toolkit, form. Some examples include ‘worse’ to ‘bad’, or ‘guitars’ to ‘guitar’. Once lemmatization is complete, the words are finished with preprocessing tasks, and can then be analyzed for positive, neutral, or negative sentiment.

4.2 NLTK Analysis

Sentiment analysis in NLTK is rather simple compared to the multi-step process in turning raw tweets into their optimal form for analysis. The NLTK.Sentiment.Vader sublibrary provides a versatile sentiment intensity analyzer - one that can properly parse text and provide insight on emotional connection to the words within the string. The preprocessed tweets are run through this analyzer, utilizing the NLTK process polarity_scores. This process takes in a single string argument and returns four metrics; a positive score, a neutral score, a negative score, and a cumulative score. Out of these four scores, cumulative score is the most useful, since it takes into account all three possible emotional metrics, rather than focusing on one singular result.

4.3 Sentiment Parsing

One arguable limitation of the NLTK library is that while it provides string metrics for sentiment analysis, it is up to the developer to decide how to handle the transformed data. As the metric being used of the four provided is the cumulative score, all tweets can be scored from a range of -1 (completely negative) to 1 (completely positive), with a 0 denoting a neutral tone throughout the tweet. There are several ways to analyze this metric:

4.3.1 Sentiment Averages

For each player analyzed, the tweets are already separated into concise groups, and thus the average cumulative score for one individual group (sum of every cumulative tweet score / number of tweets) is one of the simplest and most effective methods of tracking a generalized public opinion of any given player. A higher number might indicate a player has a higher favorability, and a lower number might indicate the opposite. However, a large collection of neutral tweets might skew the results in an otherwise biased dataset, and thus, additional forms of parsing are required.

4.3.2 Sentiment Percentages

As each player's sentiments are included in one list, then sorting each sentiment by its value (negative, neutral, positive) is another method of parsing the data NLTK offers. For smaller datasets, in our case that of Vinicius Jr.'s, a higher favorability can be seen, and the percentage of neutral tweets regarding him are much lower than that of the other players, save for Kylian Mbappe (Section 6, 'Sentiment Distribution for Vini Jr.'). Inversely, Messi's lower score can be tied to his neutral tweets being at a higher rate than any other player listed, with Karim Benzema's tweets coming in second place (Section 6, 'Sentiment Distribution for Messi'). However, the nature of these tweets is still somewhat vague, and thus there are additional ways to analyze the datasets.

4.3.3 Unique Tweets

Twitter's nature allows for users to repost a tweet sent out by another person, making it effectively appear on their own page - a 'Retweet'. In the dataset obtained by the Twitter API, Retweets make up a portion of the tweets; where some tweets begin with an '@[username]', other tweets are displayed as a 'RT: @[username]'. The percentage of unique tweets in this dataset ranged between twelve and twenty-one percent, with an average between all five players hovering around roughly fifteen percent. This might give additional insight into why some players might have polarity scores and metrics unique to them - many of the Retweets for Mbappe and Vinicius Jr. could have been positive tweets, while Messi, Benzema, and to a lesser extent Haaland might have received much more Retweets for emotionally neutral messages.

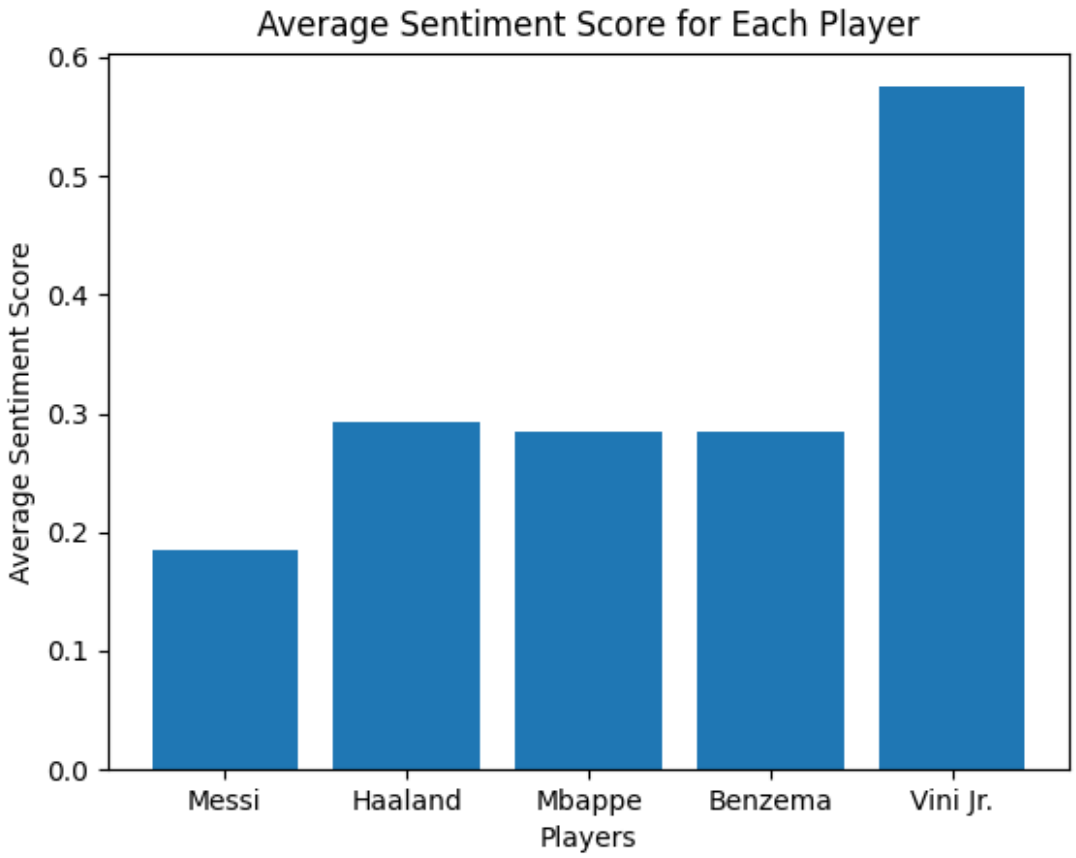
4.4 Data Handling for Visualization

The data is put in multiple concise arrays for visualization. The most crucial of these are the sentiment averages and tweet totals, and other metrics such as tweet sentiment counters (how many tweets are positive, neutral, or negative) and number of unique tweets are also provided. Each of these arrays are mutable and completely programmatic; as in they can be created to

match any number of datasets and are not dependent on any hard-coding. After this process is complete, data visualization can begin.

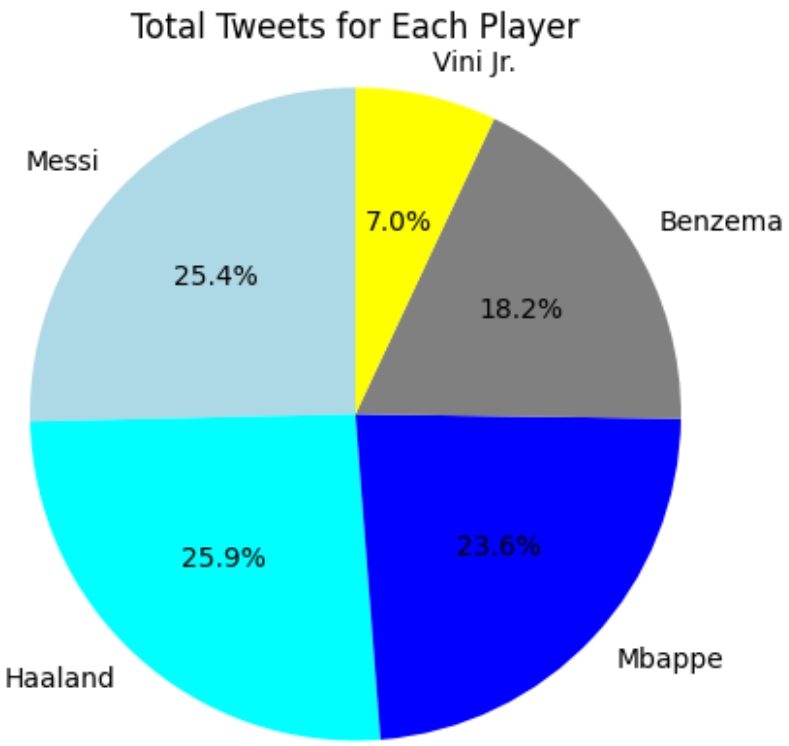
5. Data Visualization

We collected over 1,000 tweets about the ballon d'or race. With our focus on Messi, Haaland, Mbappe, Benzema and Vinicius Jr, we scraped together these tweets and then ran sentiment analysis on them. We analyze and sort the tweets into a couple of different categories. Once we have our tweets sorted, we use MatPlotLib to create pie charts and bar charts to compare and represent our data.

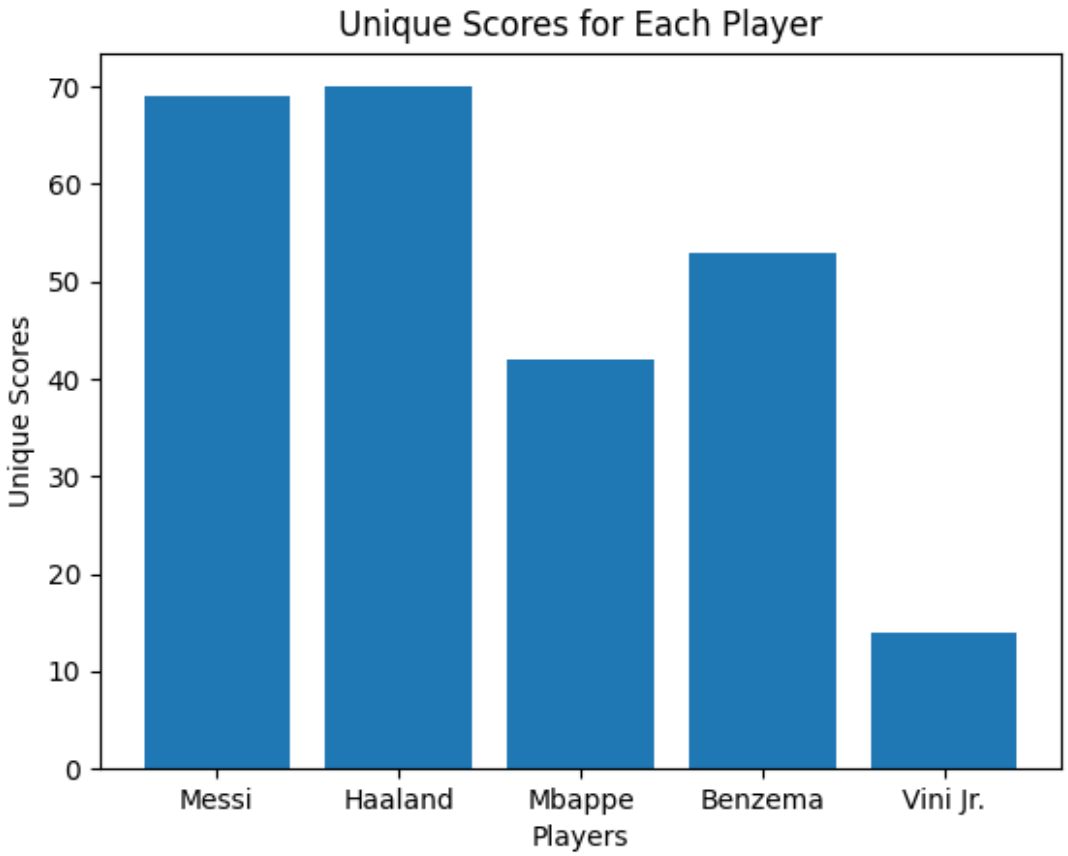


In this bar chart, it represents the average sentiment score for each player. A sentiment score closer to 1 suggests the tweets about them were more positive, while closer to -1 suggests the tweets are more negative. As we can see, all 5 of these players are good candidates as they are closer to 1 than -1, showing they are generally supported by twitter users. Messi has the lowest

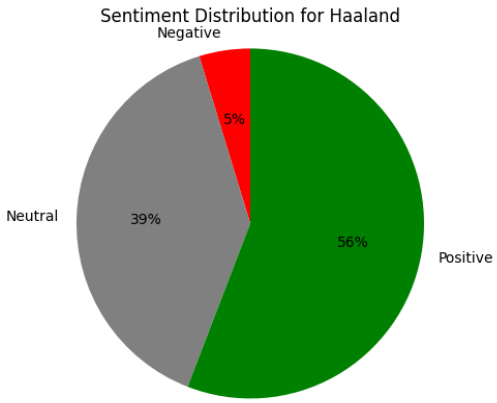
average sentiment score, while Vinicius Jr has the clear best average sentiment score. The other three players have very even scores. Although Vinicius Jr has the highest average sentiment score, we need to look deeper into the data to make an informed decision on who should win the ballon d'or.

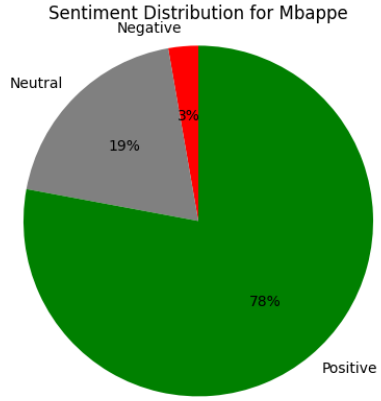


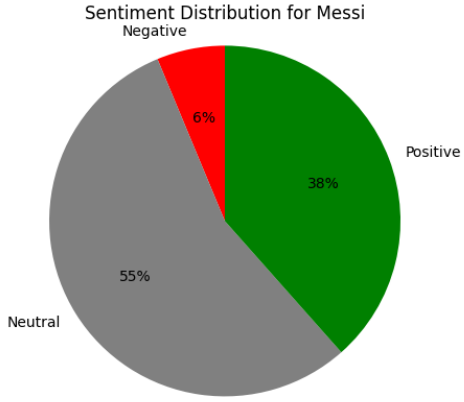
This pie chart represents the total tweets for each player. As I said when analyzing the previous bar graph, we need to look deeper into the data to make informed decisions. Here we can see that although Vinicius Jr had a very good sentiment score, he doesn't have the popularity to win the award, as not nearly as many people are talking about him as these other players. We can see that Mbappe, Haaland and Messi have very similar popularity rankings. Comparing this graph with the previous bar graph, we see that our current top 2 candidates are Haaland and Mbappe. They both have a good average sentiment score that is very close to each other, and then they each had about a quarter of the total tweets. If Benzema had a larger number of tweets with his average sentiment score staying the same, he would be in the same tier as Mbappe and Haaland.

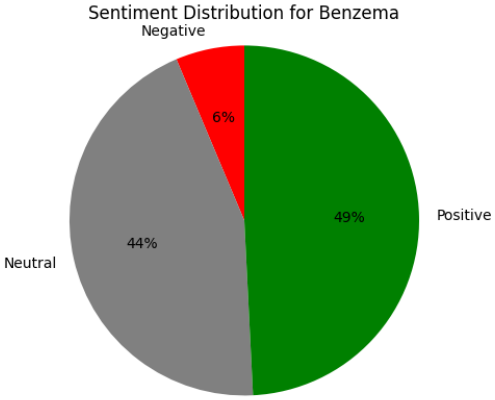


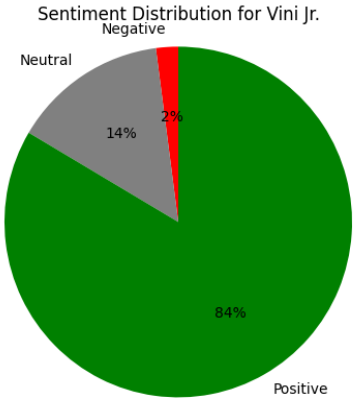
This bar chart represents the uniqueness of the tweets for the players. Having a unique score near zero, means that the tweets about the player are similar. With a higher unique score, it shows the conversations being had about the players is more diverse. This may mean that a player with a high sentiment score has people talking positively and negatively about them, or that their positive or negative tweets are not centered on one subject. Analyzing and comparing the unique scores of the players doesn't give us a clear indication on who should win the ballon d'or but it does give us a deeper insight and we can help use this information along with our other graphs to come to a conclusion.


Sentiment Distribution for Haaland
Negative
5%
Neutral
39%
56%
Positive


Sentiment Distribution for Mbappe
Negative
Neutral
3%
19%
78%
Positive


Sentiment Distribution for Messi
Negative
6%
Positive
38%
55%
Neutral





These five pie graphs are representative of the sentiment distribution for each player. So, this shows the percent of positive to negative tweets, including neutral tweets, about each player. Comparing this data with all of the information we have gathered, we can begin to make informed decisions on who should win the ballon d'or. We can see that under half of the tweets about Messi and Benzema are positive. With Vinicius Jr and Mbappe with over 75% of their tweets holding positive sentiment. Based on only these graphs, you may think Vinicius Jr or Mbappe are the expected winners for the ballon d'or. However, cross-referencing this data with our previous graphs, we see that Vinicius Jr doesn't have the popularity to take the trophy. This is why it's important to have multiple points of data to compare to make an informed decision. Having only looked at some of these graphs, it may be easy to make an ill informed conclusion.

Conclusion

In conclusion, our analysis of over 1,000 tweets about the Ballon d'Or race using the Twitter API, NLTK library, and MatPlotLib allowed us to gain insights into the sentiment and popularity of five top candidates: Messi, Haaland, Mbappe, Benzema, and Vinicius Jr. Our data visualizations included bar charts, pie charts, and sentiment distribution charts to compare and represent our findings.

We found that while Vinicius Jr had the best sentiment score, he lacked the popularity to win the award. Haaland and Mbappe emerged as the top two candidates with similar sentiment scores and total tweet counts. Benzema also showed potential, but he lacked the popularity of the other two. Messi, on the other hand, had a lower sentiment score and a relatively high percentage of negative tweets.

Overall, this project highlights the power of social media in shaping public opinion and its potential to influence major events such as the Ballon d'Or. By utilizing various data analysis tools and techniques, we were able to gain insights into the public sentiment and popularity of these five players and make informed predictions about who might win the coveted award.

So, based on our findings, we would predict that Mbappe would win the Ballon d'Or. We predict that Mbappe will take the trophy over Haaland due to their sentiment distribution. Mbappe had 78% of tweets with positive sentiment compared to Haaland's 56%. Also, they had very similar scores in total tweets and their average sentiment score. With Mbappe as the leader for the Ballon d'Or we will have to see if our data holds up when the trophy is awarded in October.

Future Scope

In this report, we utilized the Twitter API to fetch the tweets of the most popular soccer players to conduct sentiment analysis to predict the winner of the 2023 Ballon d'Or. Nevertheless, to refine the scope of the calculated results, Twitter API v2 can be used to yield further insights on Tweet topics. The context operator will allow the Tweets to be gathered from a single topic which eliminates the chance of gathering tweets that are not aligned with the search fields. Moreover, the development of a function that converts emojis and other non-natural characters to plain text will aid in the precision of analyzing the tweets. Where the context of emojis used may sway the meaning of tweets.

To dive deeper into analyzing the sentiments, the implementation of a dynamic translator will allow the program to fetch tweets in various languages from all regions. This will increase the prediction accuracy of the analysis due to an increased tweet yield from different regions of the world. Additionally, this will eliminate the bias factor of the gathered tweets because a tweet from a user that is from the same region as the candidate will be handled differently in comparison to an unbias tweet.

The program developed in this report follows a softcode ideology. Meaning that the program is not limited to the analysis of soccer players. With minor modifications, the program can perform sentiment analysis on other subjects such as politics, sports, or finance. This broadens the scope of use of the program which allows users to analyze the likelihood of possible outcomes in various topics.

Bibliography

Bird, Steven, Edward Loper and Ewan Klein (2009), *Natural Language Processing with Python*. O'Reilly Media Inc.

Matplotlib Tutorial, https://www.w3schools.com/python/matplotlib_intro.asp.

"NLTK: API Documentation." *NLTK*, NLTK Project, 2023, https://www.nltk.org/api/nltk.html.