How does the social media popularity of NBA players correlate to their season leader ranking?

Anthony Qiu Chu
Binghamton University
New York, United States

Jesse Xie
Binghamton University
New York, United States

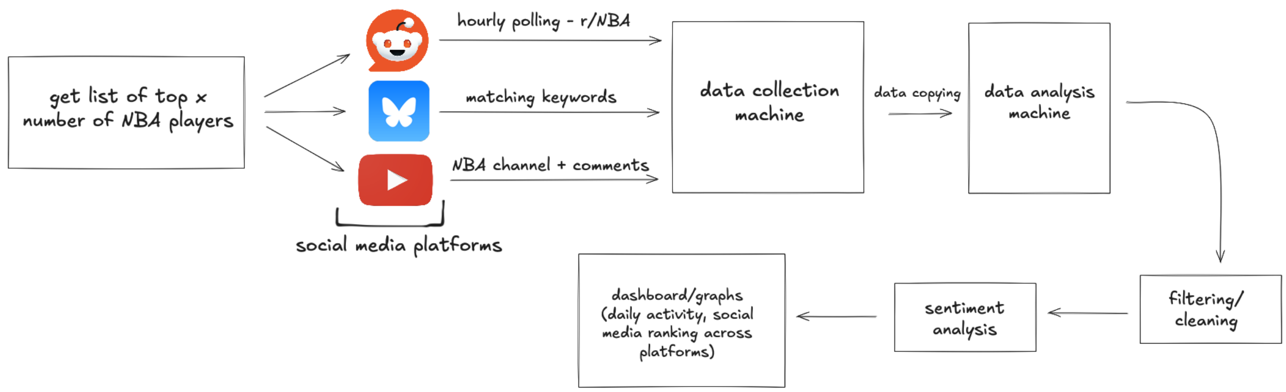

Figure 1: Data collection pipeline

Abstract

Within the 21st century, the increase in the use of social media platforms has enabled a higher level of accessibility for discussion that was not possible in the past. While much of this discussion is grounded in opinion, the NBA, especially in recent decades, has provided more advanced statistics tracking for more objective analysis of players and teams. For our project, we have implemented social media post scrapers for Reddit, Bluesky, and Youtube. These scrapers collect data about posts made of a select group of NBA players. Over the course of the last 2 months, we have collected a large amount of data totalling around 250,000 posts, which we have labeled for toxicity and briefly analyzed. We have also formed three research questions based on how the data we collected relates to the performance of these players according to the ESPN season leader rankings.

We will also statistically analyze the results/data from the posts that we have collected. We will compare factors of post volume, sentiment score, and toxicity levels to determine if there is a tangible correlation to player rankings and performance. This will be visualized through the creation of various graphs, figures, and tables. The overall data collection process will also be reviewed in terms of efficiency, consistency, and volume. Finally, we will compare the results of our analysis to our hypotheses for the experiments that we wish to conduct. We will ultimately show that in our data, we can observe slight correlations between sentiment shared online and the statistical performances of players.

Our analysis and figures are accompanied with an online tool that can contains a dashboard and can be used to filter and parse posts from our dataset. The tool also allows for various interactive figures to be visualized and that support our conclusions.

ACM Reference Format:
Anthony Qiu Chu and Jesse Xie. 2025. How does the social media popularity of NBA players correlate to their season leader ranking?. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Background and Related Work

Our research focuses on statistics involving players within the Nation Basketball Association, the largest basketball league in the world with 30 teams. As a result of its influence, the league and its players have large social media presences across multiple platforms. Our research focuses on analyzing online posts about players within the league.

1.1 Player Statistics

For decades, since the 1970s, the NBA has been keeping track of various statistics related to player performance, such as points scored, rebounds, and assists. By combining a variety of these statistics through different equations, various metrics have been developed to represent player performance with a single value. For the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference acronym 'XX, New York, NY
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

purposes of our experiments, we have elected to utilize the ESPN rating value calculated by ESPN as apart of their season leader ranking at this website: https://www.espn.com/nba/seasonleaders. This value is obtained through this formula: ESPN RATING = PTS + REB + 1.4*AST + STL + 1.4*BLK -.7*TO + FGM + .5*TGM -.8*(FGA-FGM) + .25*FTM - .8*(FTA-FTM), in which the values are the average across the games that each player has played throughout the season. For a player to rank high on the season leader list, they must average collectively a high amount of the aforementioned statistics while minimizing others like turnovers and shots missed. Though how talented or strong a player is cannot be simply represented with a single number, the ESPN rating provides a general score for how well a player is playing for the reason and fits the need of our research.

1.2 Player Performance on Social Media

Similar research has been done in the past, but on a much smaller scale. Li et al. [2] conducted a study on sentiment scores from posts made on Twitter regarding players participating in the 2019 NBA Finals, determining that there is a positive relationship between player performance and public opinion. Their experiment also used the same sentimental analysis library in TextBlob, indicating that such data can be analyzed. However, their group used different metrics from what our project utilizes (in terms of player performance score and a modified sentiment score). Research conducted by Gong et al. [1] suggests a relationship in the opposite direction - poor performances may incur negative messages and sentiment online. Through further our understanding of social media, we can refine our understanding of NBA fans and players.

2 Research Questions

Through our site and in the report, we will provide answers to two of the research questions we had before:
RQ1: How does sentiment score (average) correlate to season rankings?
RQ2: How does sentiment score (volume) correlate to season rankings?

3 Database Implementation

Before proceeding with our data collection pipeline, we first compiled a list of the NBA players that we wanted data on into a text file. Though a majority of the players on the list are not MVP (most valuable player) contenders, these 50 players represent 50 of the best/most promising players in the modern NBA era (many of whom have been selected as an all-star at least once). Within each line contains the name of each player as they are known within the league, as well as any widely used names/nicknames that can be used as search parameters for social media platforms. As an example, player Anthony Davis of the Dallas Mavericks is widely known as "AD", but is not typically referred to by his first name, "Anthony". To reduce clutter and erroneous player matches within our data collection, we have included such names that would be unique to that individual. Specific nicknames also have spacing in the file as to prevent incorrect matching (such as "ad" in the word, "bad").

All of the data that we aim to collect is put into tables created with PostgreSQL. To make use of Postgres in Python, we decided to use the psycopg3 library (the documentation can be found at https://www.psycopg.org/psycopg3/docs/). We have three different files listed under the "database" folder in our implementation repository: create_tables.py, delete_tables.py, and print_tables.py.

In create_tables.py, we connect to our database directly and utilize queries to create three different tables: players, posts, and player_posts. The players table stores the names of each player along with the number and total sentiment score of all the posts associated with that player under a player id. The posts table stores the title, host platform, url, creation time, content, and json data of a post (though some values are truncated, such as title to 100 characters). Finally, the player_posts table serves as a join table that stores relations between players and posts by player_id, post_id pairs.

Delete_tables.py executes simple queries that deletes each of the three tables and the contents within them, requiring create_tables.py to be run to restore them. Print_tables.py executes queries to retrieve the contents of specific tables and prints them to standard output/terminal. As mentioned previously, it also creates csv files for the data to be viewed more easily.

4 Data Collection Implementations

Our implementation of a Reddit data scraper leverages the praw library (Python Reddit API Wrapper, the documentation can be found at https://praw.readthedocs.io/en/stable/) to make requests for posts. By inputting valid credientals, an instance of the Reddit class can be created. Data from the subreddit, r/nba, can be obtained by creating an instance of the subreddit class and retrieving posts made in the last 24 hours with the subreddit.top() function. Though data can be obtained every hour, doing so every 24 hours allows users to make comments on posts, providing us with more data. The function used to fetch posts can be faulty, so it may be run multiple times to ensure that they are properly retrieved.

For each post that is fetched, the comments can be parsed through the submission.comments attribute. Each comment is split into sentences, and each sentence is processed by our sentiment analysis model from TextBlob. To determine associations between player names and comments, the text file containing our player names is iterated through to see if the player names are in the text of the sentence. Occurrences are marked down in a dictionary object, with the player id stored with the polarity and number of occurrences in sentences from the comment.

Comments that are found to contain instances of player names, as stored in the dictionary/map, will be added as a post with relevant attributes such as content and date created supplied as parameters to a PostgreSQL query into our database. The players database is also edited with a query to update values for total sentiment score and the number of posts found. Associations between player id and post id are also added to the player_posts table. The same algorithm is used for the actual content of the post.

Our implementation of the Youtube data scraper relies on the Google Client library to make requests for posts. For our implementation, we decide to scrape videos and comments under the official

NBA YouTube channel between a 24-48 hour time frame. Similar to our reasoning for the Reddit scraper, we do so in such intervals so that comments can fully populate under the video. We fetch the ID for the NBA uploads playlist and scrape for the video ids that appear within that time frame.

With those video ids, we fetch the video title and comment threads.

Videos posted by the NBA channel include highlights of the previous day's games. Our goal is to identify players in these teams that are discussed in the comments and assess their polarity. However, we also noticed that these videos included short clips of specific NBA players, such as a short highlight clip or a training clip. Due to this, the comments all reference the same player without the player's name specifically being stated. For this reason, we also instantiate a default player id, where if we detect a player's name in the title of the video, we will default to associating comments with that name. Otherwise, we will attempt to identify player names explicitly mentioned in the comment itself.

With these methods in mind, the same algorithm from our Reddit scraper can be utilized. We similarly search for player names that appear and split the comment into multiple sentences in the event that there are multiple player names. We calculate the polarity score of each sentence using TextBlob, and we track the average of each player's polarity score using a players_polarity map. After processing a given comment, we update the posts table. After processing all comments, we iterate through the polarity map and update the players and player_posts tables accordingly.

Our implementation of the Bluesky scraper was possible by simply logging in with our account credentials. For Bluesky, we iterate through the player list names and create a query search for each player name. In this search, we only include the player's full name to avoid scenarios where we get irrelevant results: for example, searches for "butler" for player Jimmy Butler of the Golden State Warriors may yield unrelated results about other individuals, locations, or the profession.

From these search results, we generate the URL leading to the post and similarly calculate the polarity score of each post and store in a polarity map. In this scraper, we do not split each sentence since we are guaranteed the player's name occurring. Similarly, after processing a post, we update the posts table. After processing every post, we iterate through the polarity map and update the posts, players, and player_posts tables.

All of the code that was written was pushed to a GitHub repository, and then cloned onto a virtual machine. After setting up PostgreSQL on the machine and installing the required libraries, we created cron jobs to schedule the scrapers to run at approximately 8:00 AM each morning.

4.1 Filtering

A lack of proper filtering would have been problematic for nicknames such as "butler" or "ad". However, specific design choices in the scrapers account for this potential issue.

In particular, Bluesky would have posed the largest problem as we search platform-wide by keyword: however, we have already decided to search solely by full name there. In the other two platforms, YouTube and Reddit, we scrape channels/subreddits related to the NBA, so there we consider the nicknames as they are likely to be directly related to the players that the nicknames are referring to. For players with common first names or surnames such as "Jalen" or "Brown", we simply do not search for these keywords on any platform.

Although there is still the chance that from searching up or finding matches of the unique player names, we find occurrences of other individuals such as referees, coaching staff, broadcasters, or other previous/current players mentioned in NBA-relevant spaces, we feel that the vast majority of such discussion pertains to the NBA players themselves. However, this is still a limitation of our project and is difficult to circumvent without a more advanced model. Furthermore, occurrences of multiple names related to different players are treated with the same sentiment score (so if a single sentence in a post is positive towards one player and negative towards another, a single sentiment score is generated and attributed towards both players).

5 Toxicity Analysis

We decided to use the Perspective API by Google. We requested access to the API via a Google Form and once it was approved, we added a new column to the database called "toxicity_score" with a default value of 0.0 for newly added data, and NULL for existing rows of data around 3 weeks after our data collection started.

Due to the rate limit of 1 request per second (or 60 requests per minute) with Perspective, we ran a cron job to process our existing rows of data in a process that would take a few days. We added a sleep(1) command at the end of every iteration to ensure that the subsequent request processes without the API rate-limiting us. Error handling was also added for posts that could not have a calculated toxicity score: we default their toxicity_score value to 0.01 to ensure it is not processed and gives an error again in the future. After all of the existing data was labeled, a cron job was scheduled to run after the data scraper execution each day to label new posts.

6 Content

To create our tools and displays, we import the data from the Postgres tables that accumulated over the last 2 months. This encompasses around 250000 posts. We created a dashboard that facilitates interactive queries to find individual posts and the sentiment statistics for players. This would include attributes such as player name, date created, platform, and more. Our tools/site will also contain displays for various statistics, such as the number of posts created each day by platform, the average sentiment score of each player, and how season leader ranking relates to sentiment scores and posts made.

More specifically, our figures include:

- Interactive figures for Sentiment Score Over Time Per Player, where you are able to filter by player
- Toxicity Score Over Time Per Player, where you are able to filter by player

- Top 50 Players by Sentiment Score vs ESPN Ranking, where you are able to filter by date and compare to ESPN Ranking Data
- Dashboard: Filter the posts database by toxicity score, platform, and date range

7 Results

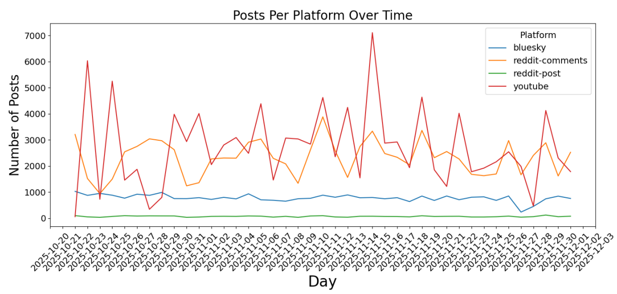

Figure 1: Posts By Each Platform Over Time

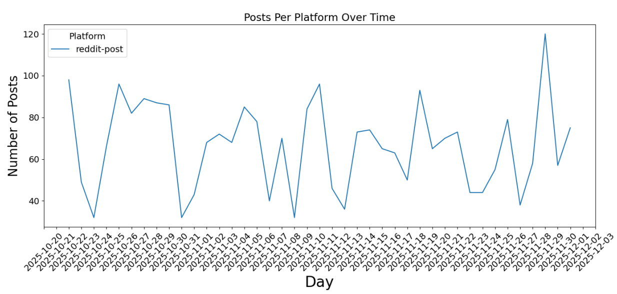

Figure 2: Posts By Each Platform Over Time (Reddit Posts Only)

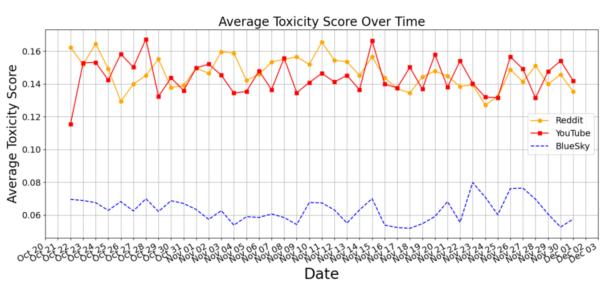

Figure 3: Average Post Toxicity Over Time

Average Sentiment Score of Posts Per Player Over Time

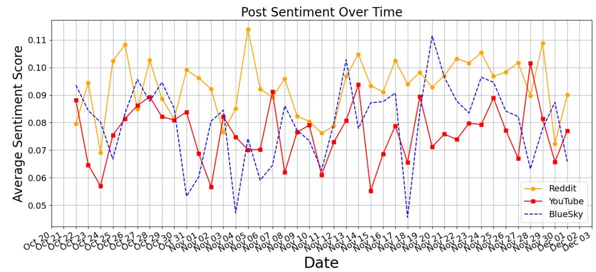

Figure 4: Post Sentiment Over Time

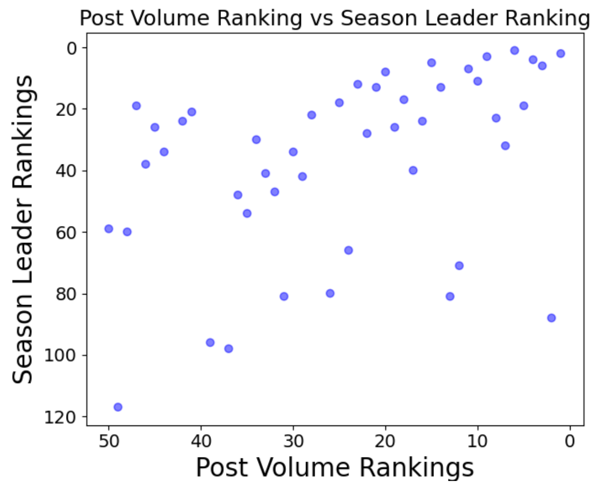

Figure 5: Post Volume Ranking Vs Season Leader Ranking

8 Discussion and Analysis

Looking back at our figures, we notice the most overlap of players within the top 10 range. Players in our top 10 that receive consistent posting such as Luka Doncic, Shai Gilgeous-Alexander, Victor Wembanyama, Giannis Antetokounmpo, and Tyrese Maxey appear in the ESPN ranking's top 10 as well. From the information on the tables and figures, we can identify that YouTube and Reddit have the most post content out of the three platforms. We believe this is due to the fact that there is an established NBA-specific subreddit and official NBA channel with a large amount of users. On the other hand, there is comparatively a lack of presence of NBA related content on Bluesky, and the NBA content across the entire platform is not as unified. We can also see spikes in posts per platform and player sentiment score the day following a game. This can be attributed to their higher visibility and potential highlights after a game has occurred.

8.1 Figures Analysis

A look at figure 1 and 2 shows that post count on each platform spikes up and down at fairly regular intervals. These fluctuations can be linked to the number and quality of NBA games as well as player performances that day. For instance, on Thanksgiving

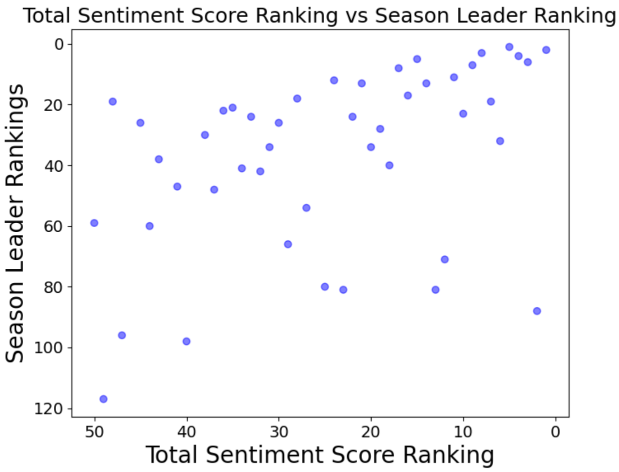

Figure 6: Total Post Sentiment Score Ranking vs Season Leader Ranking

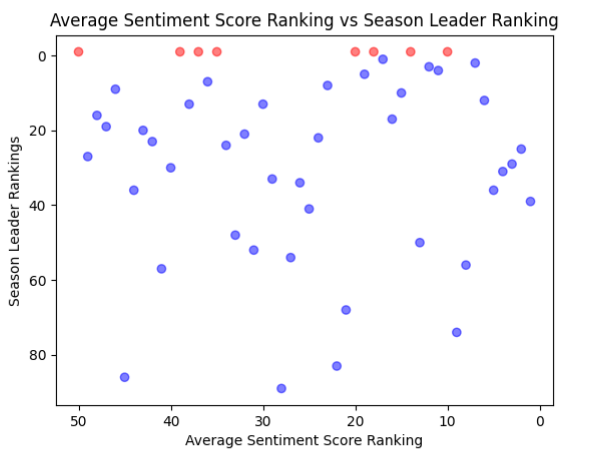

Figure 7: Average Post Sentiment Score Ranking vs Season Leader Ranking

Day 11/27/25, only one Youtube video was uploaded to the NBA channel. Due to the fact that the Youtube scraper looks through the comments made on videos posted within the last 24-48 hours, only comments made on the one video posted that day would have been collected, leading to an incredibly small number. As such, it may have been better practice to collect posts made on videos with a greater range, though our choice was made to prevent potential issues with duplicate posts in our database.

Some additional observations include the difference in toxicity across platforms. For example, in Figure 3, we see that in Reddit and YouTube, the average toxicity score among them is roughly 0.15 while the average on Bluesky is significantly lower, averaging around 0.07. As for post sentiment, we see that Reddit has the

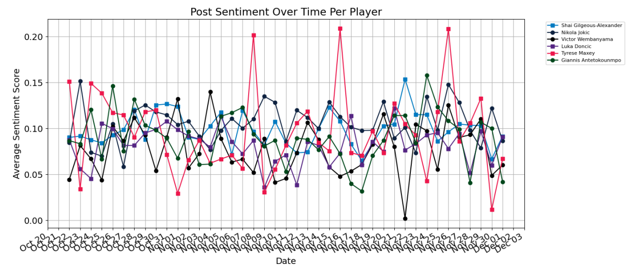

Figure 8: Average Sentiment Score of Posts Per Player Over Time

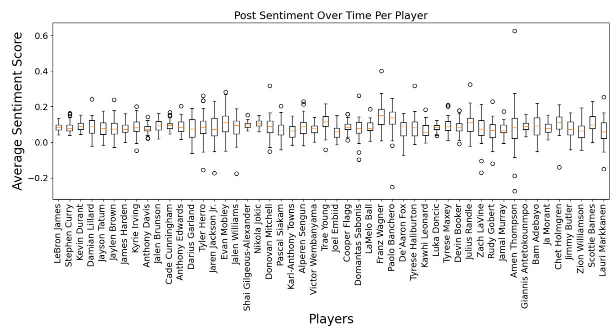

Figure 9: Average Sentiment Score of Posts Per Top 6 Players Over Time

highest sentiment values. This implies that Reddit posts are on average more positive than YouTube and Bluesky.

Correlations that we found between the volume of posts, total post sentiment score, and average post sentiment score as depicted in the figures were also lower than expected, which will be explored in greater detail below.

From plotting Post Volume Ranking against Season Leader Ranking (Figure 5), we can observe a slight positive correlation between the two. This correlation becomes stronger if outliers are removed, especially those with season leader rankings lower than 60. Plotting Total Sentiment Score Ranking against Season Leader Ranking (Figure 6) as well as Average Sentiment Score Ranking against Season Leader Ranking (Figure 7) shows a similar distribution, though the correlation may be weaker.

Figure 8 provides a closer look of the data in Figure 6 with only the average sentiment scores of the top 6 players in ESPN season leader ranking as of the creation of this report. Generally, post sentiment fluctuates up and down every 1-2 days. Higher sentiment scores can, in part, be attributed to the days that a player plays and performs well, while lower sentiment scores can, in part, be attributed to the days that a player does not play (possibly from injury) or does not perform well. As an example, player Tyrese Maxey and his team had a notably good game on 11/8 against the Toronto Raptors, but had a poor showing on 11/9 against the Detroit Pistons.

8.2 Statistical Analysis

The following statistical tests will utilize Pearson's Correlation Coefficient test to determine if there is a correlation between the two variables, as depicted by the points on a scatterplot. The coefficient can be derived from this formula:

$$r = \frac{n \sum xy - (\sum x)(\sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}}$$

From conducting the correlation coefficient test on the relationship between Post Volume Ranking and Season Leader Ranking with the rankings from each player as depicted in Figure 5, we find that there is a correlation coefficient of r = 0.4585. Utilizing the Pearson distribution, we can obtain a p-value of 0.001353. Given an significance level of 0.05, this p-value is statistically significant. This indicates that there is a moderate correlation between the two variables, which may be attributed to the fact that many of the best players are also some of the most popular players. The strength of the correlation may be weighed down by the fact that some players that have built up a reputation and fan base within the league have entered a decline as a result of their age or external factors. Conversely, not as popular, younger players may have also started approaching new highs in their career. Though the correlation coefficient is not particularly high, it still can be concluded that there is some correlation.

Conducting the same test on the relationship between Total Post Sentiment Score Ranking and Season Leader Ranking with the rankings from each player as depicted in Figure 6 yields comparatively, a slightly higher correlation coefficient of r = 0.4794. Utilizing the same Pearson distribution, we can obtain a p-value of 0.000749. Given an significance level of 0.05, this p-value is statistically significant. These values indicate that there is a moderate correlation between the two variables, which may not be surprising due to the fact that Total Post Sentiment Score Ranking and Post Volume Ranking are closely related. On average, the sentiment score of a post leans positive for every player involved in the study, as depicted in Figure 9. As such, with more posts made about a player, the higher their total sentiment score will likely be. However, the higher correlation coefficient from this test compared to the previous may possibly indicate that a better performing player will yield a greater total sentiment score than average.

However, conducting Pearson's Correlation Coefficient test on the relationship between Average Sentiment Score Ranking and Season Leader Ranking as depicted in Figure 7 yields a much lower correlation coefficient, of r = 0.1395. With the Pearson distribution, we can obtain a p-value of 0.355159, a value that is not statistically significant at an alpha value or significance level of 0.05. We believe that this can be attributed majorly to the unreliability of our sentimental analysis model - from taking a closer look at individual posts and the sentiment score associated between these posts and players, we observed inaccuracies from the sentiment scores returned from the Textblob model and the actual sentiment as determined by us. This can be attributed to the fact that the TextBlob library is not familiar with slang used within the online NBA community, or basketball terminology and metrics. As a result of this, the sentiment of many posts may be inaccurate (likely closer to 0 than it should be, or negative when it should be positive). For example, as depicted on figure 8, player Nikola Jokic of the Denver Nuggets with the number one season leader ranking on ESPN had a comparatively low average sentiment score on 11/23 even though he performed well on his game the night before. One of the posts praising his performance that day, a simple comment of "I'll always respect your game Jokic", yields a sentiment score of -0.4 when passed through the sentiment analysis. Inaccuracies like this have likely hurt the strength and accuracy of such statistical analysis. In the future, a model can be trained to more accurately label the sentiment of such posts, though it will likely be slower than the model provided by the TextBlob library.

8.3 Table Analysis

To generate the data for **Table 1. Total Number of Posts Each Week On Each Platform**, we use a SQL Query to sum the posts within the provided date ranges for each week starting on 10/22, with the last date excluded. For example, in Week 1 (10/22-10/29), posts made on 10/29 are excluded.

Similarly, to generate the data for **Table 2. Top 10 Sentiment Score Players vs Actual Ranking**, we use a SQL query to get the top 10 cumulative sentiment scores and retrieve the respective player id with each. Then, we use those player ids to look up the player's name in our players list. To compare with the top 10 actual season rankings, we look back at our scrapped HTML data of the ESPN webpage. We use the ESPN player rankings at the end of the range (for example, 10/29 for Week 1) and manually label each rank. To reiterate, an entry in this table are in the format (Our predicted player for the rank, ESPN's player for that rank).

9 Conclusions

Given our figures, tooling, and statistical analysis, we can observe that there may be some correlation between the popularity of a player and their season leader ranking. We can make sense of this through higher coverage of a player after good performances. These posts are also generally positive afterwards and only cause a player's total sentiment score to become increasingly higher positively. However, we cannot find evidence that correlates a player's average sentiment score to their popularity ranking as it is unlikely for even a high ranking player's sentiment to be consistently positive and higher than the rest of the players.

References

[1] Hua Gong, Nicholas M. Watanabe, Brian P. Soebbing, Matthew T. Brown, and Mark S. Nagel. 2021. Do Consumer Perceptions of Tanking Impact Attendance at National Basketball Association Games? A Sentiment Analysis Approach. *Journal of Sport Management* 35, 3 (June 2021), 254–265. doi:10.1123/jsm.2020-0274

[2] Qiwen Li, Jiarui Zhang, Jiayu Guo, Jiaqing Li, and Chenhao Kang. 2022. Evaluating Performance of NBA Players with Sentiment Analysis on Twitter Messages. *Proceedings of the 2021 European Symposium on Software Engineering* (2022), 150–155. doi:10.1145/3501774.3501796

Table 1: Total Number of Posts Each Week on Each Platform

Week	YouTube	Reddit	Bluesky	Total
Week 1 (10/22-10/29)	15731	15496	6291	37518
Week 2 (10/29-11/5)	19661	15604	5530	40255
Week 3 (11/5-11/12)	16100	15043	4687	35830
Week 4 (11/12-11/19)	22973	15439	4932	43344
Week 5 (11/19-11/26)	17579	15944	5383	38906
Week 6 (11/26-12/3)	15554	17003	4847	37404

Table 2: Top 10 Sentiment Score Players vs Actual Ranking (Weeks 1-3)

Week	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Week 1 (10/22-10/29)	(Victor Wembanyama, Luka Doncic)	(Luka Doncic, Giannis Antetokounmpo)	(Anthony Davis, Victor Wembanyama)	(Shai Gilgeous Alexander, Austin Reaves)	(Lebron James, Tyrese Maxey)
Week 2 (10/29-11/5)	(Luka Doncic, Luka Doncic)	(Victor Wembanyama, Giannis Antetokounmpo)	(Lebron James, Nikola Jokic)	(Giannis Antetokounmpo, Tyrese Maxey)	(Anthony Davis, Shai Gilgeous Alexander)
Week 3 (11/5-11/12)	(Luka Doncic, Nikola Jokic)	(LeBron James, Luka Doncic)	(Victor Wembanyama, Giannis Antetokounmpo)	(Shai Gilgeous Alexander, Shai Gilgeous Alexander)	(Nikola Jokic, Tyrese Maxey)
Week 4 (11/12-11/19)	(Luka Doncic, Nikola Jokic)	(Lebron James, Luka Doncic)	(Nikola Jokic, Giannis Antetokounmpo)	(Shai Gilgeous Alexander, Shai Gilgeous Alexander)	(Victor Wembanyama, Tyrese Maxey)
Week 5 (11/19-11/26)	(Luka Doncic, Nikola Jokic)	(Lebron James, Luka Doncic)	(Nikola Jokic, Giannis Antetokounmpo)	(Shai Gilgeous Alexander, Shai Gilgeous Alexander)	(Victor Wembanyama, Tyrese Maxey)
Week 6 (11/26-12/3)	(Luka Doncic, Nikola Jokic)	(Lebron James, Luka Doncic)	(Shai Gilgeous Alexander, Giannis Antetokounmpo)	(Nikola Jokic, Shai Gilgeous Alexander)	(Victor Wembanyama, Tyrese Maxey)

Week	Rank 6	Rank 7	Rank 8	Rank 9	Rank 10	Overlap %
Week 1 (10/22-10/29)	(Kevin Durant, Nikola Jokic)	(Anthony Edwards, Shai Gilgeous Alexander)	(Zach LaVine, Lauri Markkanen)	(Giannis Antetokounmpo, LaMelo Ball)	(Nikola Jokic, Jamal Murray)	0%
Week 2 (10/29-11/5)	(Shai Gilgeous Alexander, Victor Wembanyama)	(Ja Morant, Austin Reaves)	(Kevin Durant, Devin Booker)	(Nikola Jokic, Josh Giddey)	(Stephen Curry, Julius Randle)	10%
Week 3 (11/5-11/12)	(Kevin Durant, Austin Reaves)	(Nikola Jokic, Victor Wembanyama)	(Anthony Davis, Cade Cunningham)	(Ja Morant, Donovan Mitchell)	(Cade Cunningham, LaMelo Ball)	10%
Week 4 (11/12-11/19)	(Stephen Curry, Victor Wembanyama)	(Kevin Durant, Donovan Mitchell)	(Giannis Antetokounmpos, Cade Cunningham)	(Cade Cunningham, Austin Reaves)	(Ja Morant, James Harden)	10%
Week 5 (11/19-11/26)	(Stephen Curry, Victor Wembanyama)	(Kevin Durant, Cade Cunningham)	(Giannis Antetokounmpos, Donovan Mitchell)	(James Harden, James Harden)	(Cade Cunningham, Austin Reaves)	20%
Week 6 (11/26-12/3)	(Kevin Durant, Victor Wembanyama)	(Stephen Curry, Cade Cunningham)	(Giannis Antetokounmpo, Donovan Mitchell)	(Cade Cunningham, Austin Reaves)	(James Harden, Jalen Johnson)	0%