NBA Twitter Correlation

Andres Fandino
University of Central Florida
Orlando Florida USA
andres.fandino@knights.ucf.edu

Josh Harkness
University of Central Florida
Orlando Florida USA
joshua.harkness@knights.ucf.edu

Mina Sonbol
University of Central Florida
Orlando Florida USA
mina.sobol@knights.ucf.edu

ABSTRACT

Prediction methodologies using machine learning increased drastically in the last few years. Several studies were done where the public mood was used to predict economic markets and other important public events such as public office elections[1][2][3][5][6]. This study identified basketball player performance as an area where Twitter feed sentiment analysis could be used as a predictor. Unlike other similar studies, this study used the players' tweets to analyze the mood of the tweet and predict the player's performance.

Identifying the most relevant parameters in the sentiment analysis was perhaps the most relevant feature of this problem. This study aimed to predict one or more basketball statistics collected for each player. Compared to other team sports, basketball provides a wealth of statistics collected for each player. The study envisioned a model where FGM (field goals made) positively correlates with positive sentiment analysis but negatively when the tweet sentiment is negative. This assumption was based on other studies that demonstrated negative sentiment was a better predictor of downward moves in a firm value.

This study is important for basketball team organizations and fans. Coaches can use the analysis to support starting lineup decisions and fans can decide whether to attend a game and pay the high-ticket price for a good player performance or stay home and watch the game on the television when the performance is predicted to be poor. In the aggregate, these types of predictions can give insight to basketball organizations during contract negotiations and possibly obtain a better ROI on the player. It also furthers the understanding of athletic performance as it relates to the athlete's mood. Players in team sports can gain insight on upcoming matches and, in the case of basketball, give input to coaching staff on how many minutes should be played. Lastly, in localities where sports gambling is legal, this tool can be valuable to gamblers in placing better bets.

The idea for this project came from research done by J. Bollen, H. Mao, and X. Zeng, titled "Twitter Mood Predicts the StockMarket"[1] and published in the Journal of Computational Science in 2011. Similar research was conducted by Mittal et al. [6] in 2012. The researchers of these studies wanted to find if there was a correlation between sentiments collected from large Twitter feeds and the stock market as measured by Dow Jones Industrial.

There are two major differences between our project and the aforementioned research. To begin with, there is a considerable difference in data size. While the researchers in [1] used 9.7 million tweets collected from 2.7 million users, this study focused on the tweets made by the top players over five seasons. For example, in the 2019-20 season, the top 15 players by total points score had a total of 1500 tweets between them.

Another difference is the scope. The researchers focused only on tweets that explicitly mentioned their author's mood, as this aligned with what they were trying to achieve; to find whether public mood can predict the performance of the stock market. This study included all tweets made by the top-scoring players. The focus was on whether the tweets can shed a light on their mood in the days leading up to any given game and whether that has any discernible effect on their performances.

The study used several tools to solve the problem. For data collection, the NBA API collected performance statistics, and the snscrape API collected tweets.

For sentiment analysis, VADER was used to predict positive, negative, or neutral sentiments and the NRC lexicon captured other emotions, i.e. fear, anger, etc. An embedded analysis was also used. These sentiments were the main features, but in the future other features such as emojis, mentions, and hashtags could be useful.

Once the dataset was collected, different models were used to test their predictive power. The target variable will be the total points or accuracy per game above the average for that player.

The specific models chosen were linear regression, decision trees for regression, and multilayer perceptron (MLP).

The evaluations reflected the study's primary focus of how the sentiment of tweets correlates with the points scored by a top player. Each analysis compared the tweets leading up to a game with the points scored by the player in that game. Simply comparing the scores would not be enough though, because even amongst the best players, each person averages a different point total for each game.

Instead, the study took each player's season points scored, obtained the average, and compared it against each individual game. The tweets' sentiments were analyzed alongside the point differential with the average to determine if a correlation existed. The expected result was a positive correlation would exist where positive sentiments would lead to more games with points scored above the average.

The study also took shot percentage into account. A player scoring fewer points may not have to do with them, but rather the team they are up against or the player's teammates not getting them the ball. To account for this potential offset, the shot percentages could be used instead of points. The idea for this was even if the player doesn't get the ball much, if they still make the most out of their shots (the shot percentage is at or above average), a low scoring game should not count as a negative result.

Our Experiment

A natural experiment approach was used to examine the available Twitter data from the top NBA scorers. We made the basic assumption that positive tweets would correlate with better than average performance and the inverse with negative tweets. The software we implemented followed the architecture of data capture, data cleansing, entity assembly, sentiment analysis, model generation, and lastly data visualization. The software flow diagram is shown in figure 1.

Our data acquisition approach involved three separate data loaders each with a specialized purpose. The NBA data loader's main purpose was to retrieve a player's season statistics using the NBA API. The Twitter data loader retrieved a player's tweets and grouped them using date-time information based on the games played during the seasons analyzed. The Twitter handler loader module loaded the selected top players in which the analysis was going to be performed.

The data cleansing step was divided into two modules. An NBA and Twitter data cleanser. The NBA data cleanser input was the data produced by the NBA data loader. This cleanser removed the unnecessary data columns, calculated an average used for the modeling, and aggregated the player's points and accuracy made across different categories. A decision to combine all the player's points into one total to simplify the experiment was made. The scoring statistics were reported in different categories separating the three point and regular two-point shots into

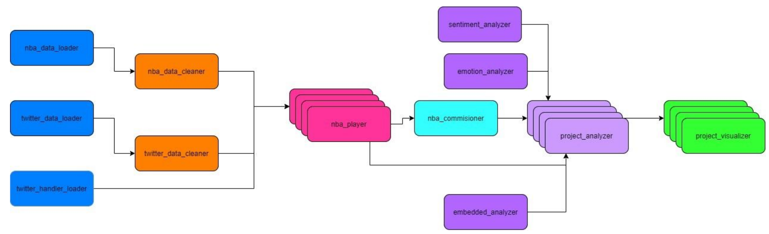

Figure 1: Software wire diagram

independent buckets.

The NBA player class created player objects which called the required functions from previous steps and the output was the data required to perform the experiment. For instance, the player object has a method "getEverySeasonStats" that retrieves all the stats available for a given player. It also has a method "getEverySeasonTweets" that retrieves the player's tweets for all available seasons.

The NBA commissioner module read in all the player information from a pickled file to save on processing time and reduce the number of calls made to the different APIs. Another benefit of the NBA Commissioner module was it provided a central location to access all player information.

Two modules for the sentiment and emotion analysis of the data were implemented. The emotion analyzer module used the NRCLex package to generate emotion frequencies for text. Values between zero and one were returned for emotions such as fear, anger, trust, surprise, sadness, disgust, joy, and anticipation. The sentiment analyzer module imported the VADER library to provide a positive, negative or neutral score for the observations.

1. Method

The software designed followed the classical approach of system modularity. The problem was decomposed into smaller parts and python modules or classes were created with standardized interfaces. This approach made the python code more efficient and easily maintainable. The loosely coupled modules also increased the flexibility in the software. If a problem was encountered in a specific function of the code, it was easily replaced without affecting other parts of the system. Another important principle applied to the software produced was the adherence to the PEP 8 “Style Guide for Python Code”. The python code defined internal and external methods as prescribed in the guide improving the collaboration between developers working in different modules.

The tweet data was extracted from Twitter using their API. The Twitter loader class was designed to collect and format the Twitter data into a suitable data structure. For each tweet collected, three parameters were passed in to specify a time interval and the Twitter handle of a player. The Twitter data was returned in the form of a python dictionary with the tweet ID as a key. This python dictionary object was one of the basic data elements used in the other system modules. The second main data element for the experiment was the player’s scoring statistics. This data was extracted using a third-party API (nba_api). The NBA data loader class implemented several methods to retrieve the player’s season statistics and season ranges. The returned object from this class was a dataframe with rows representing the observations per player in a single game and the columns representing the scoring statistics.

2. Challenges

Some of the issues encountered while doing this investigation were in the data preparation step. Existing data mining packages do not take into consideration the idiom used in Twitter. For example, the acronym “lol”, which is common knowledge to be interpreted as “laughing out loud”, was not taken into consideration when performing text mining until VADER was introduced. In addition, an incorrectly spelled word in the tweets deteriorated the model. Sarcasm was also widely used in the tweets making the interpretation of tweets problematic for VADER. Another problem noticed with many of the tweets was the intended usage. Numerous players used their Twitter account to promote products that are unrelated to their state of mind and in some cases, their accounts are managed by third parties. In the latter case, emotions were completely removed from how the player was feeling the day of the game. Because the model depends on the availability of tweets for each individual player, we encounter problems with the size of the datasets to produce consistent results. In future work a solution for the issues listed can be implemented.

3. Tools

We selected VADER (Valence Aware Dictionary for sEntiment Reasoning) to create our sentiment analysis labels. VADER is a rule-based model for general sentiment analysis specifically designed for micro-like contexts like Twitter. VADER uses a valence-aware lexicon to provide a score for a sentiment label.

Although VADER is specifically designed for social media contexts it did not classify the language in the tweets as expected. The largest percentage of tweets were classified as neutral making the sentiment analysis less informative.

The VADER lexicon is a human-curated gold standard that uses rules to assign values to keywords and return a numerical sentiment measure for the phrase. Figure 2 shows the proportion of positive/negative/neutral tweets in our data.

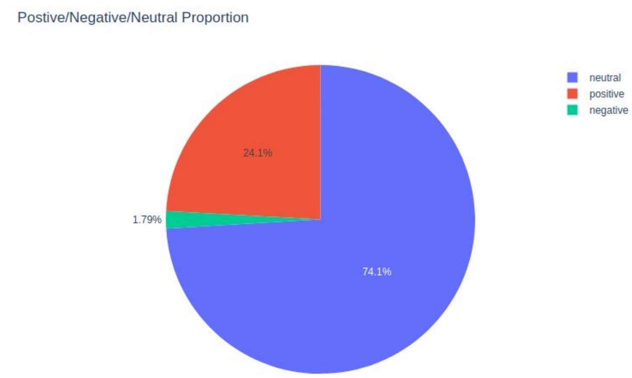

Figure 2: Proportion of positive/negative/neutral tweets

In the software, the NRCLex library was imported to produce emotion analysis from the tweets. This is another lexicon that can be used to produce scores for emotions including fear, anger, anticipation, trust, surprise, positive, negative, sadness, disgust and joy. Figure 3 shows the emotions through time for one player.

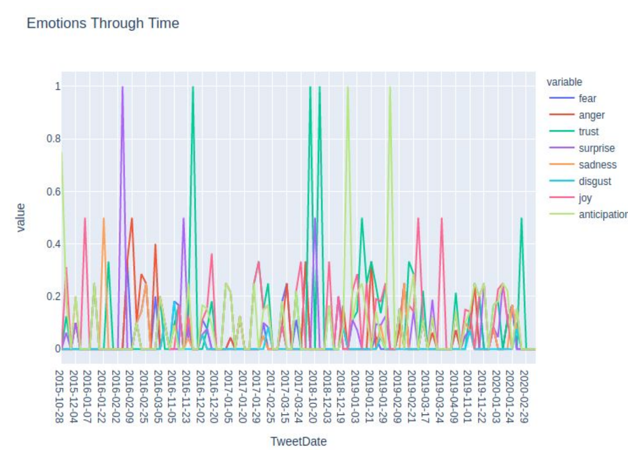

Figure 3: Emotions through time for one player

Another tool used to extract features from the input text was the universal sentence encoder[7], which takes plain text as input and produces a high dimensional vector that captures the semantic significance of the sentence as output. These vector embeddings can then be used as features in their own right or coupled with other features to reinforce the meaning of the sentence. Figure 4 shows the average tweet length vs. the size of embedding (512-dimensional vector) per tweet.

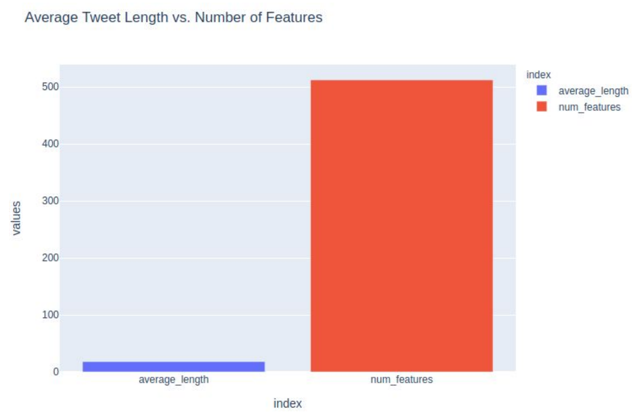

Figure 4: Average tweet length vs. the size of embedding (512-dimensional vector) per tweet

To generate predictions and construct a model, three libraries from scikit-learn [4] were used. Linear regression, MLP regressor, and the Random forest regressor were imported into the project. The results of the models and performance metrics are discussed in the results section.

The MLP regressor provided us the advantage of learning non-linear models and implementing the analysis using a neural network model. The random forest regressor is a decision tree-based model and is a stable algorithm that is minimally affected by bias.

The GitHub platform was utilized to produce the software. Using git and the pull requests feature kept the code in line with coding standards and allowed us to provide feedback on the code. Github Actions was also used as a ticket system to help identify what each person worked on and when.

4. Results

Twelve experiments were conducted to determine whether a correlation exists between a player's emotional state in the days leading up to a game and the player's performance during that game. Representations of their results are shown in figure 5. All the results are shown in table 1. Each experiment tested one analysis using one regression model. There were four different types of analyses: sentiment, emotion, embedding, combinations, and three different regression models: linear regression, MLP, random forests. When predicting points, Linear regression with sentiment features reported the lowest mean squared error value, with a score of 69.1938, while MLP with the sentence embedded features reported the highest mean squared error value, with a score of 103.7417. On the other hand, when the models were trained to predict accuracy, Linear regression with emotion features reported the lowest mean squared error value, with a score of 0.0136, while MLP with the combined features reported the highest mean squared error value, with a score of 0.0205. However, in both cases, the best performing models were very conservative in their predictions, showing low variance but very high bias. More importantly, most models had a negative r-squared score. In other words, the predictions made by these models were worse than predicting a constant value. This can be attributed to several factors. One of these factors was the small size of the training dataset. Focusing on the top 15 players per season only put a limit on the amount of tweets that can be collected for the study. Also, it should also be highlighted that many players use account managers to curate their online presence, which in some cases dilutes the emotional attribute of their tweets, as evidenced by the high number of tweets with a neutral sentiment (figure 2). These are some of the factors that could be dealt with in future work, by including more players in the analysis, using text attribution models to identify the real author of the text, and using deep learning models to detect sarcasm in text. In addition, more strict data cleaning mechanisms could be implemented to exclude tweets that have no meaning, or convey low emotional/sentimental attributes. However, there are other factors that can also contribute to the bad prediction scores, such as behavioral factors that could be manifested by players who react in different ways when dealing with the same emotion. For example, some players, when angry, may play better while some may play worse. Another factor that could play a role here is the fact that these are professional athletes who can maintain high-performance levels while exhibiting negative emotions. Such factors could be harder to quantify.

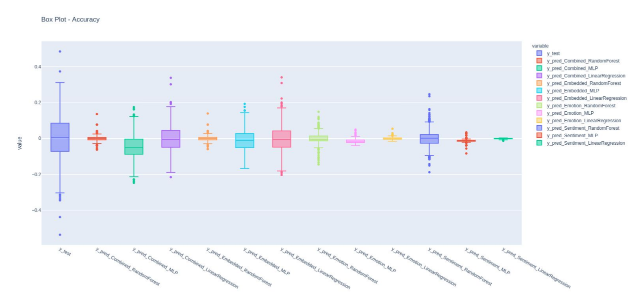

Figure 5.a: Boxplot - Real and predicted +/- accuracy scores

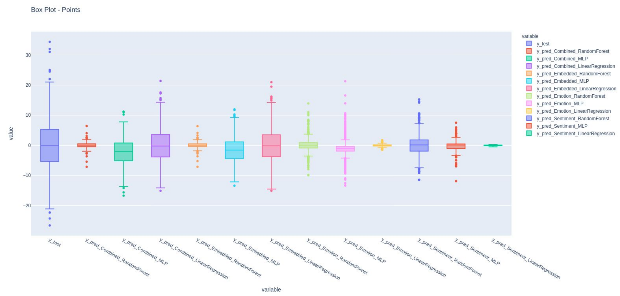

Figure 5.b: Boxplot - Real and predicted +/- point scores

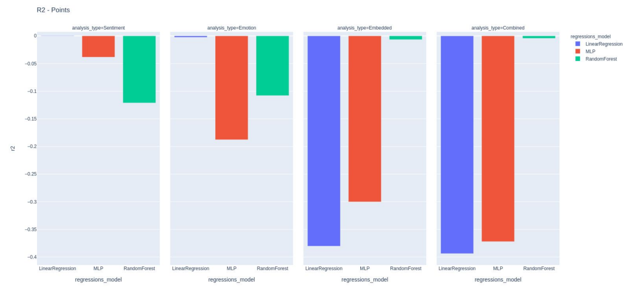

Figure 5.f: R-squared scores for models predicting +/- points

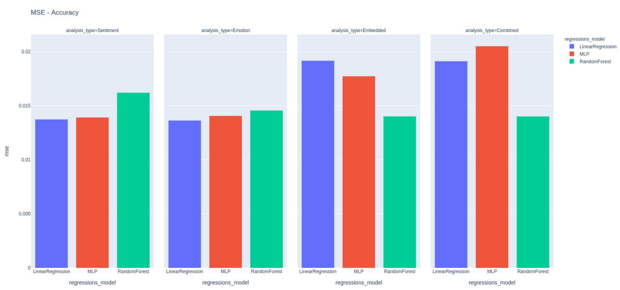

Figure 5.c: MSE scores for models predicting +/- accuracy

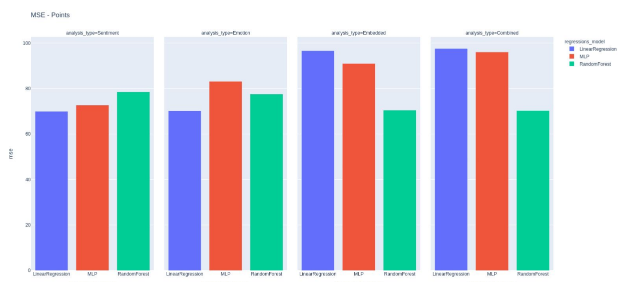

Figure 5.d: MSE scores for models predicting +/- points

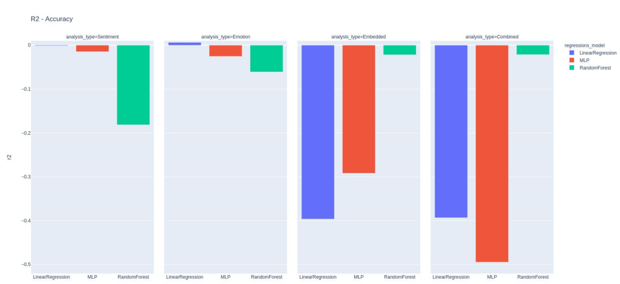

Figure 5.e: R-squared scores for models predicting +/- accuracy

REFERENCES

[1] Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. Journal of Computational Science 2, 1 (March 2011), 1–8. DOI:https://doi.org/10.1016/j.jocs.2010.12.007

[2] CJ Hutto and Eric Gilbert. 2014. VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Association for the Advancement of Artificial Intelligence (2014).

[2] Brendan O'connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. 2010. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. Proceedings of the International AAAI Conference on Weblogs and Social Media (May 2010), 1–8.

[3] Robert P. Schumaker, A. Tomasz Jarmoszko, and Chester S. Labedz. 2016. Predicting wins and spread in the Premier League using a sentiment analysis of twitter. Decision Support Systems 88, (August 2016), 76–84. DOI:https://doi.org/10.1016/j.dss.2016.05.010

[4]G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller. 2015. Scikit-learn. GetMobile: Mobile Computing and Communications 19, 1 (June 2015), 29–33. DOI:https://doi.org/10.1145/2786984.2786995

[5] Yang Yu and Xiao Wang. 2015. World Cup 2014 in Twitter Word: A big data analysis of sentiments in U.S. sports fans tweets. Computers in Human Behavior 48 (February 2015), 392–400.

[6] Mittal and Goel, 2012. Mittal, A. & Goel, A. (2012). Stock prediction using twitter sentiment analysis. Stanford University Working Paper.Google Scholar

[7] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua,Nicole Limtiaco, Rhomni St John, Noah Constant,Mario Guajardo-Cespedes, Steve Yuan, Chris Tar,et al. 2018.Universal sentence encoder.arXivpreprintarXiv:1803.11175.

ANALYSIS	MODEL	R2	MSE
Emotion	LinearRegression	0.006248187786148	0.013649536269069
Sentiment	LinearRegression	-0.001257783679918	0.013752633469498
Sentiment	MLP	-0.014730682739989	0.013937688552779
Combined	RandomForest	-0.021280423409362	0.014027651581496
Embedded	RandomForest	-0.021748544054577	0.014034081385846
Emotion	MLP	-0.025322968504823	0.014083177383031
Emotion	RandomForest	-0.060800755193749	0.014570477461587
Sentiment	RandomForest	-0.181416270215119	0.016227174663707
Embedded	MLP	-0.291866267192957	0.017744244842738
Combined	LinearRegression	-0.39314538428057	0.01913534970916
Embedded	LinearRegression	-0.396224264571372	0.019177639230227
Combined	MLP	-0.494469119392222	0.020527067420087

Table 1.a: MSE and R-squared scores for models predicting PlusMinusAccuracy

ANALYSIS	MODEL	R2	MSE
Sentiment	LinearRegression	0.000520572665997	69.9488174748632
Emotion	LinearRegression	-0.002470963172298	70.1581808580401
Combined	RandomForest	-0.004247653943518	70.2825230056288
Embedded	RandomForest	-0.006550511995866	70.4436891018674
Sentiment	MLP	-0.038362583416584	72.6700648695414
Emotion	RandomForest	-0.107854198799196	77.5334529368651
Sentiment	RandomForest	-0.121162910640748	78.4648664607166
Emotion	MLP	-0.187752635645566	83.1251650048174
Embedded	MLP	-0.30004962224951	90.9842976733995
Combined	MLP	-0.371978760822317	96.0182764102846
Embedded	LinearRegression	-0.380195255499422	96.5933098434239
Combined	LinearRegression	-0.393680260038495	97.537061255764

Table 1.b: MSE and R-squared scores for models predicting PlusMinusPoints