

Lindsay Kotouch, Craig Beach, Brian Taylor

IST 736: Text Mining | Sunday 5 pm

Final Report

Fantasy Football Anyone?

# Introduction



Wings, Beer, and a Big screen TV? Sounds like a football party! Where did the craze for the “Greatest Game Ever Played” start?

American or “Gridiron” Football started in the 1870’s as a cross between rugby and soccer as an intercollegiate sport with the first recognized game being between Princeton and Rutgers. It started migrating towards using rugby style rules within a few years with every region having their own version of rules before quickly transitioning to the game much more familiar to us today due to Walter Camp, who is known as the “Father of American Football” and was a major influence in the second American Intercollegiate Football Association.

1892 marked the first step towards making football a professional sport when the first player was paid $500 dollars to play a game. This quickly led to the first professional and fully paid game in 1895, and the first professional team in 1897. Popularity continued to grow and eventually the NFL or National Football League was established in 1922. They knew they had it made when the 1958 Championship game between the Baltimore Colts and New York Giants netted a whopping 45 million viewers. Over the last 60 years, that popularity has continued to grow.

Along with the growing obsession with sports and football, came the advent of fantasy sports. Not surprisingly it began with baseball as one of the more statistically driven sports. For football, it started in a hotel in 1962 as a way for the owner of the Oakland Raiders tried to think of a way to make watching a horrendous team more exciting. These days, Fantasy Football not only makes the sport more interesting, but it also fosters camaraderie, and depending on the league, can be quite profitable.

In fantasy football, “Players” build their own teams using actual players, and they score when the players score or make plays in the real games. Each week the player with the highest points when matched up against each other wins, leading up to a league champion, just as in the real game. Participants create their teams before the beginning of the season. This is a great time for statistical analysis of previous years, overall performance, and other factors can be useful in building a team. However, each week, teams can be rearranged, players can be traded, and other players benched which requires a strategic approach to picking a team each week. With money on the line, every disadvantage, such as injuries or weather, or advantages, such as great coaching and many other factors can be critical knowledge. Many of these topics are regularly mentioned in social media. If that’s the case, can an examination of social media help build the winning team?

# Analysis and Models

### About the Data

*Positions:*

*QB = Quarterback*

*TE = Tight End*

*RB = Running Back*

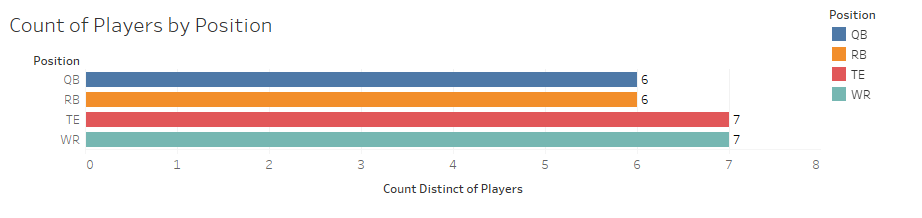
*WR = Wide Receiver*

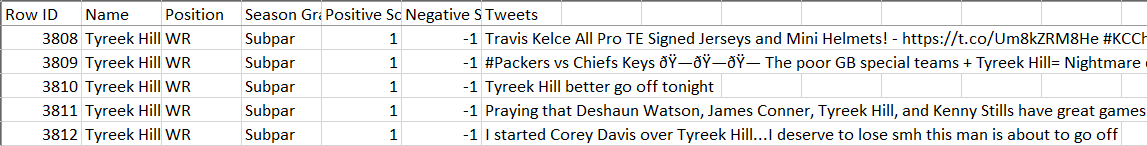
*Season Grade:*

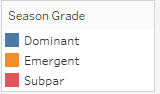
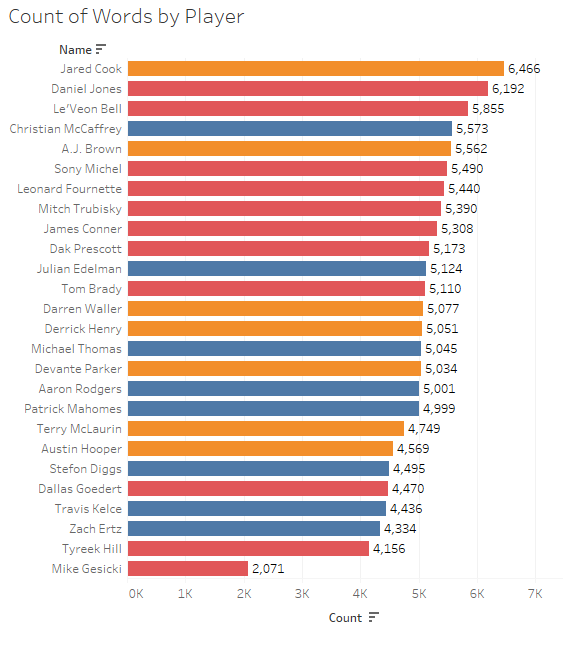
*Dominant = player that is having a top tier fantasy season AND has had one before.*

*Emergent = player that is having a top tier fantasy season AND has NOT had one before.*

*Subpar = player that is having a fantasy season below standards/projected season.*

 For the data, 26 players who fall across the spectrum of talent, were chosen with there being 6-7 players for each of 4 positions.

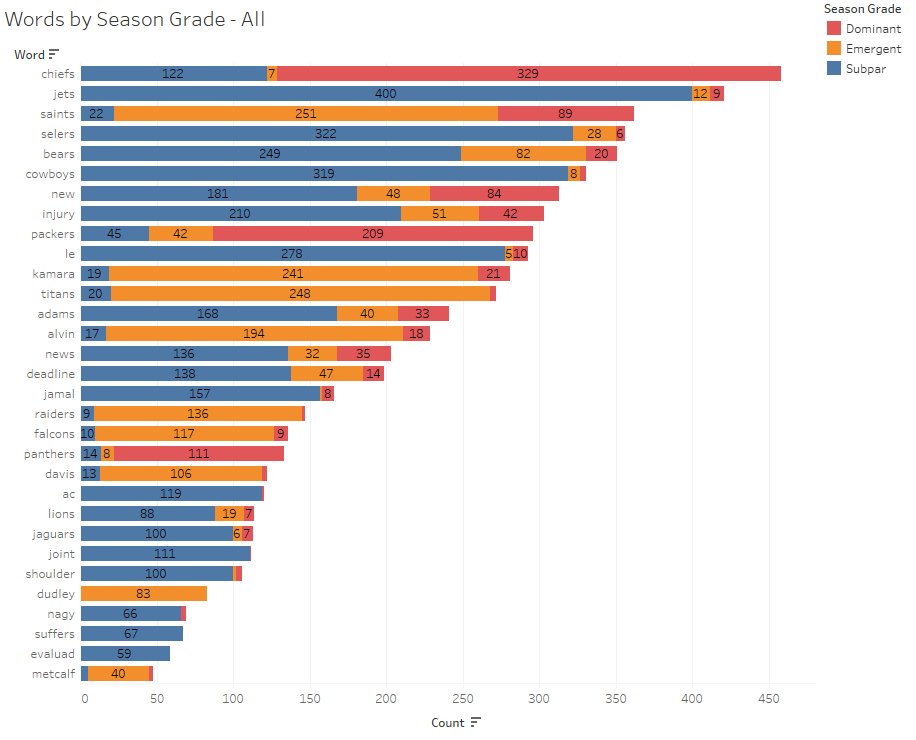
The players were labeled with their position, a Season grade, defined to the right, and 500 tweets were pulled for each player with each tweet given a sentiment score for positive and negative sentiment using SentiStrength. Before cleaning or tokenizing, the dataset looks like the below picture with 7 columns and 12613 rows as shown here:



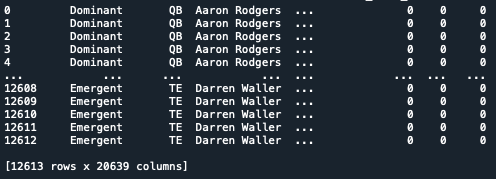
The data was then analyzed for basic statistics and information to determine trends.

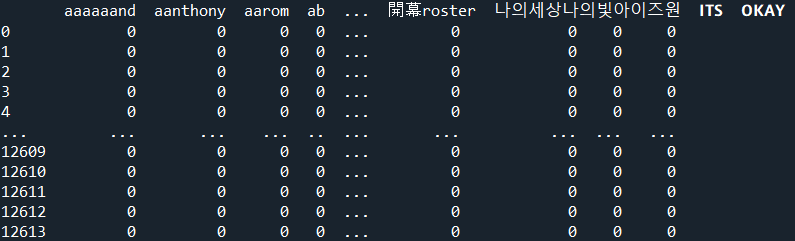
Here is a breakdown of the number of words per player as well as the Season Grade category they fell into.

It appears that Subpar and Emergent players are talked about the most in Social Media, though it is an uneven dataset for the most part as some players are just discussed more than others.

Looking at the most frequent words, while some words appear in all grade areas, there are many words that are more likely to appear in one grade more than others, mostly in subpar then emergent. Teams names seem prominent, while there are also other words such as injury, new, deadline, and joint which may be indicative of terms that people use to predict how well players will perform or what may affect fantasy football scores.

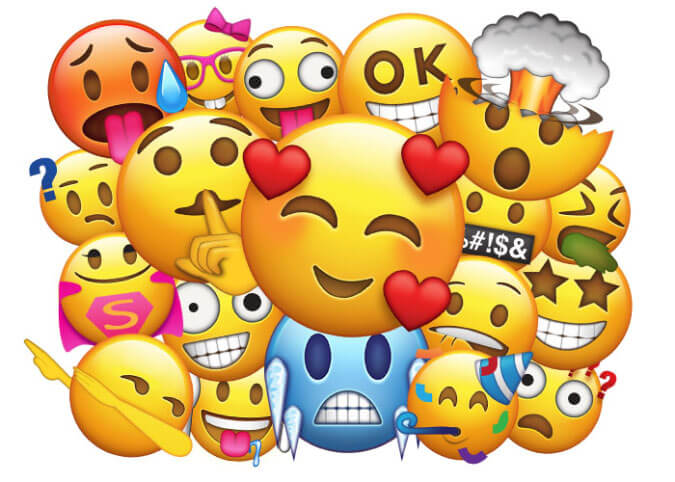
The dataset was then cleaned removing emojis, random punctuation caused from pulling Twitter data, Players names (the ones selected), as well as other random symbols and icons. The data was then run through a count vectorizer with no lemmatization or stemming for all models. For Naïve Bayes, the data was also run through a TFIDF vectorizer and another vectorizer with binary for Bernoulli NB.

Here is the cleaned and vectorized dataset for the Naïve Bayes and SVM models. It includes the word features and the labels of Player names, positions and Season Grade.

The data was run through the same cleaning and vectorizing process for the LDA topic modelling, however, the only column pulled was the tweets and no labels were included.

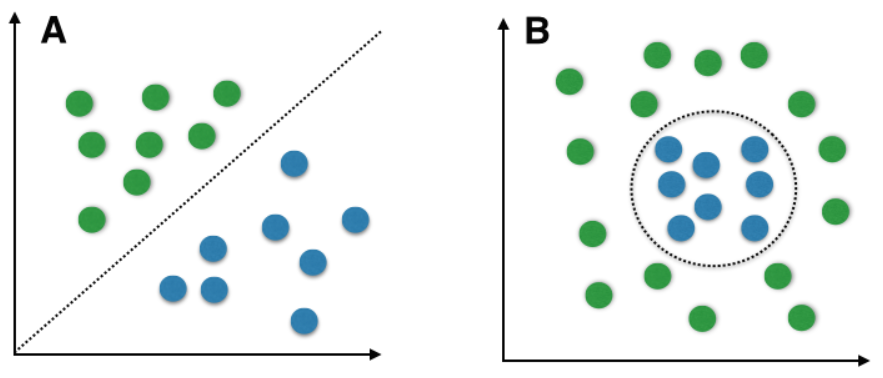
### Models

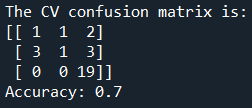
#### SentiStrength for Sentiment Analysis

 SentiStrength is a lexicon and sentiment analysis tool that was specifically developed to determine sentiment in short social media posts such as Twitter. It gives both an indication of the strength of positivity and negativity of a post by assigning a score from between -4 to 4, which allows for more precise measurement of sentiment that compensates for some of the ambiguity and subtlety that can be found in any language. As it has already been trained on all varieties of social media data, it does not require extra training data and can be used directly on new social media data with accuracy ratings like those of English speakers’ hand labeling the data. The overall scores are derived by using a dictionary with words already assigned positive or negative scores based on their usual usage in everyday language. The scores of each word are then calculating an overall positive score and negative score of between -4 to 4 as seen here:

#### Multinomial Naïve Bayes

Multinomial Naïve Bayes was used to analyze the three data frames with Count Vectorizer, Term Frequency Inverse Document Frequency (TF-IDF) vectorizers. The Multinomial Naïve Bayes Model is a basic classification model that assigns probabilities that something will be classified in a certain way based on known probabilities. It works best as a linear model for data that looks like A below that can easily be separated by a line and is clearly defined instead of data like in B that requires a more complicated model.

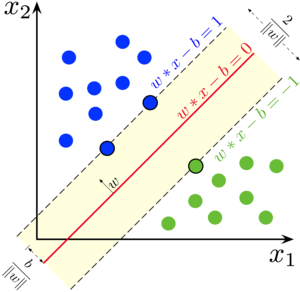


 Using the sklearn Multinomial Naïve Bayes classifier, the model was defined, and the three training sets of data were applied. The test sets of data were then run through the trained models and the predicted labels were then compared to the actual labels to get confusion matrices and accuracy ratings for each model.

#### Bernoulli Naïve Bayes

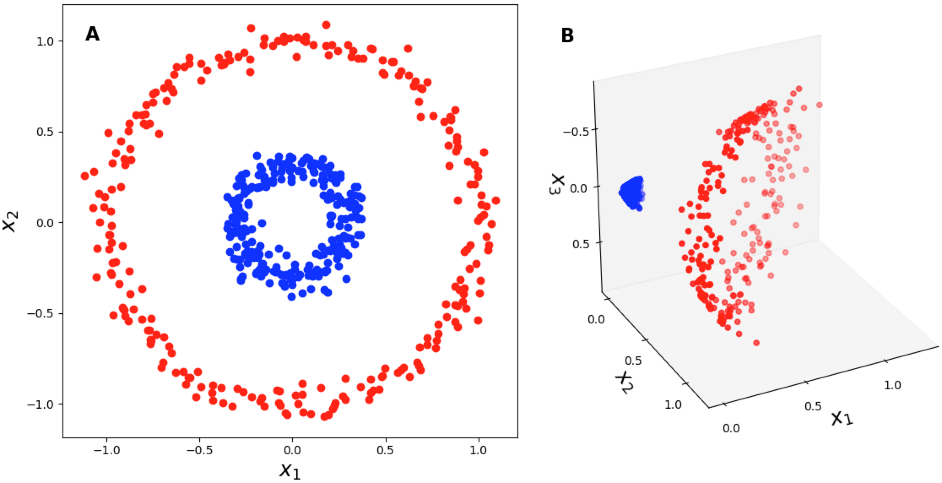
The same thing was done with the last dataset, though it was run using the Bernoulli model. The Bernoulli model is in the Naïve Bayes family and is run very similarly, however it only has 2 outcomes, P(X=1)=p or P(X=0)=1-p, or in other words, it only looks to see if a term is present or not and does not take the frequency of the features into account. This can make it much simpler as it only looks at 0’s for absence and 1’s for presence without needed to count up the frequency, making things faster for much larger datasets. The downside though is it may have a much harder time with ambiguous data. A word that may regularly be used in a positive way, could be present if used sarcastically in a negative way, which can throw off any model, but especially this model if it’s weighting words equally based on presence.

#### Support Vector Machine (SVM)

 Like Naïve Bayes, SVM’s are a supervised learning model that requires training and test data, however, unlike Naïve Bayes, is just used the points closest to the line and finds a line that has the widest gap or margin possible. SVM algorithms can change the shape of the data to try and find a better fit if the data does not have a nice linear shape. It does this with kernels.

Linear Kernel

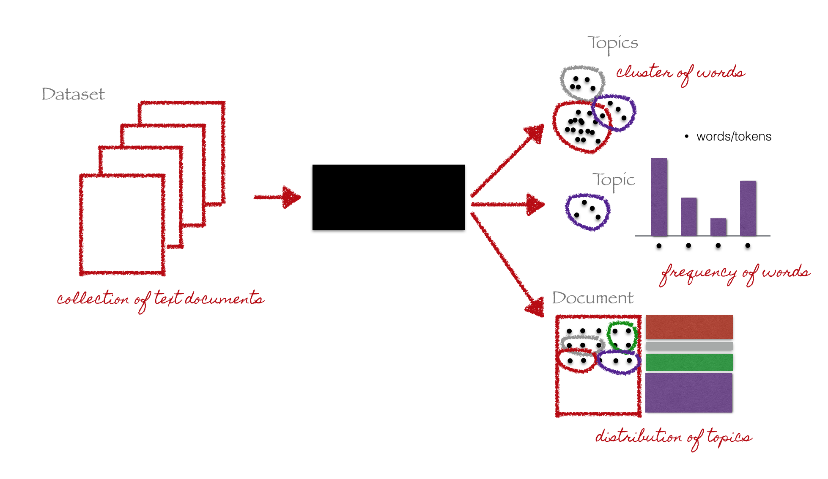
The linear kernel is used when data can be separated linearly and would also work best with the same form of data that works with Naïve Bayes data.



Polynomial Kernel

This kernel transforms the data into polynomial shapes that would allow the data to be separated by a plane after it’s been folded or curved.

#### LDA Topic Modeling

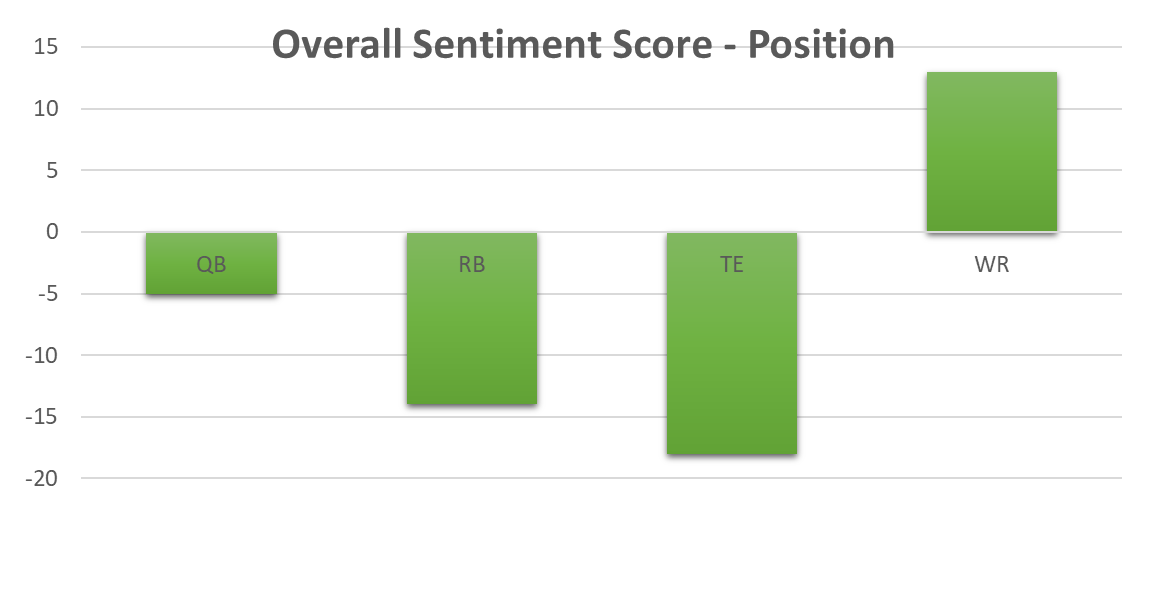
 LDA or latent Dirichlet allocation (LDA) is an unsupervised learning cluster model. It takes a bag of words and assumes there are k topics across the documents and gives each word a topic, that it assumes that every other topic assignment is wrong, then probabilistically groups words together based on how many times they were assigned a particular topic. These word groups are assumed to form a particular topic; however, the model doesn’t list an actual topic itself. That responsibility falls on the user to interpret the results and determine what the overall topic is for each cluster. The issue with this, is that any particular grouping can actually have multiple interpretations, especially when the entire topic is more focused or the terms used are too general. It may also be an issue if there are multiple clusters with similar vocabulary that most likely end up being the same or very similar topics, which may mean the number of clusters may need to be reduced.

The dataset was first cleaned and vectorized as done previously before being run through the LDA model. It was first run for 10 clusters, before reducing the count to 7, 4, and finally 3.

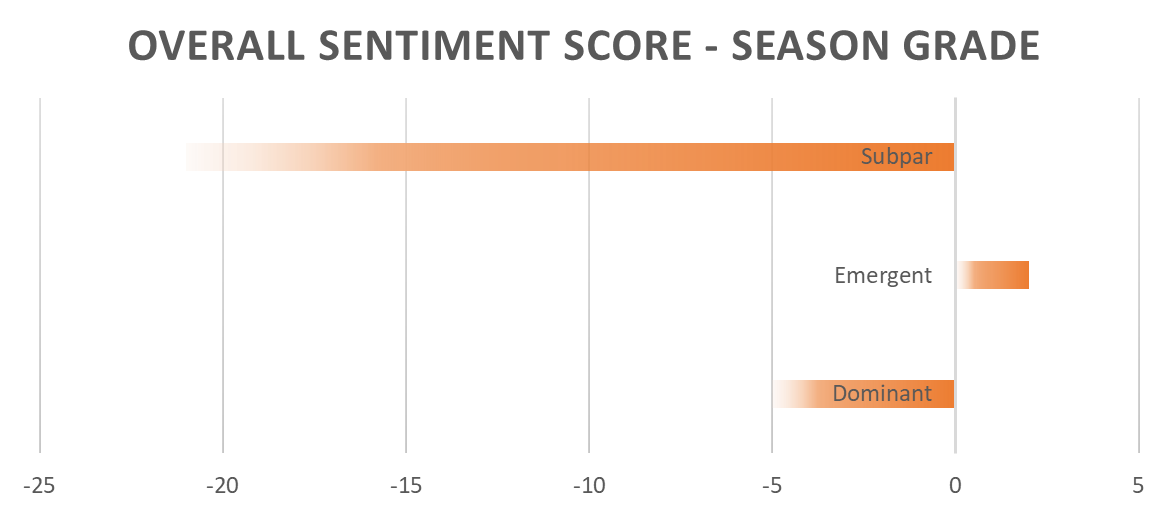
# Results

#### SentiStrength

After using SentiStrength to assign positive and negative sentiment scores to the data, basic data analytics were performed to determine if sentiment may be a good indicator of Season Grade or Position. The interesting thing was that while most of the positions had an overall negative sentiment, tight ends have an abnormally negative social media presence, while wide receivers are the only position to have an overall positive score, even for the subpar players. When it comes to Season Grade, while there is a difference between the 3 grades, there is enough of a negative slant to the dominant grade, most likely from the tight ends and low positive dominant scores for the other positions that it did not seem to be a feasible option for further analysis currently.



|  |  |
| --- | --- |
| Position/Grade | Sum of Total Score |
| **QB** | **-5** |
| Dominant | 1 |
| Subpar | -6 |
| **RB** | **-14** |
| Dominant | 3 |
| Emergent | 1 |
| Subpar | -18 |
| **TE** | **-18** |
| Dominant | -13 |
| Emergent | -5 |
| Subpar | 0 |
| **WR** | **13** |
| Dominant | 4 |
| Emergent | 6 |
| Subpar | 3 |



#### Naïve Bayes vs. SVM

Accuracy for the Naïve Bayes models generally outperformed the Support Vector Machine models. The primary dependent variable, Season Strength, was predictable with an accuracy of at least 63%. The classifier made a slight difference in the results, with a basic Count Vectorizer used with Multinomial NB achieving 66% accuracy. In the case of predicting names, a TF-IDF vectorizer with NB scored 52%. Given 26 possible names, that model performed favorably. In all, although the accuracy scores are lower than expected, the resulting models are useful. For instance, identifying the correct position of a player out of 4 possibilities just from tweets has an expected probability of about 25%. Achieving 68% accuracy in that task is a relatively large improvement. And while the polynomial SVM performed almost or as well as the linear SVM model, all of the Poly SVM models had one category where they performed perfectly with the others being perfectly incorrect as seen in the plot below.

When looking the number of correct vs. incorrect, the success rate matches with the initial look at the data for the players. The players with the most words associated with them were in the subpar category and that also ended up being the category with the most correct predictions for each of the models as they were better trained on that data category.

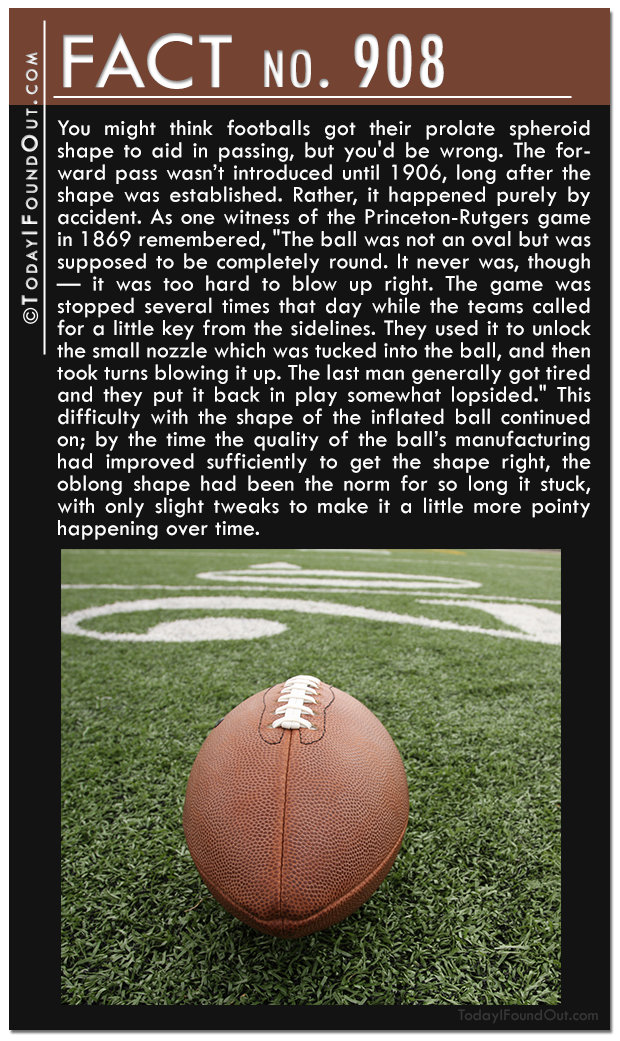


#### LDA Topic Modeling

 After running the data through the model for 10 clusters, it was quickly apparent that there were several topics that were very hard to distinguish. The number of clusters was then reduced to 7. There was at least one distinct cluster that was dropped, however, there were still 3-4 clusters that were too general to differentiate. After reducing the number of clusters to 4, there were 3 distinct clusters with one that was just general topic terms that could also be tied to a cluster that seems to be about fantasy football and its predictors, or the elements people reference when determining how they should play their team. After running the model for 3 clusters, the following topics were identified:

Topic 0 appears to be about injuries and how that affects games, Topic 1 seems to be the fantasy football/predictors topic, and the last one, Topic 3 seems to be about the Saints and some of their games they played. The main theme of the data searched is what mainly appeared while big games, news, or teams that trended during the time frame pulled form the other clusters with this particular data.

# Conclusions

 Social media can help to build a winning team. When choosing players for a fantasy football team, it is very helpful to know as much about a potential “teammate” as possible. Social media, particularly Twitter, is a trove of information beyond that of just performance. Sometimes, external factors off the field are as important as performance on the field. Sometimes, discussion of physical health and injuries are discussed with more detail on social media than in typical sports reporting. Player availability is a primary factor, if not the primary factor in selecting players for fantasy teams.

Text mining was able to produce identify three different categories of performance for a player, their name, and their positions. Although not perfect, the models achieved useful results. For a hobby that can have large amounts of money on the line every week, every competitive advantage must be maximized. Utilizing technology to improve performance is a necessity.

While this process yielded interesting and useful results, it was just a beginning that showed that social media could be useful in predicting the general trend a player’s season took based on a month of data. Further development using weekly data compared to weekly fantasy football scores would give a much deeper look into the viability of the process as well as possibly be helpful in knowing who to play or bench each week.

In addition, expanding the source of information to include sports news and commentary may help improve or even provide better sources of text data for the models. For the use in commercial applications, it may even be prudent to develop a football specific sentiment analyzer to further see if sentiment may play a bigger roll than shown here in predicting how players will perform, as there are many instances of football specific slang or positive use of negative words that appear in social media for this context.

There are many avenues to explore that look very promising as well as very fascinating.