Project Fletcher - LeBron James: An NLP Exploration

Spencer Tollefson

November 16, 2018

Project Design

reddit.com/r/nba is the subreddit dedicated to the National Basketball Association (NBA). With over 1,420,000 subscribers at the time of writing - its membership dwarfs all other major American sports leagues' subreddits in size and even the 1,217,000 subscriber-strong r/soccer. Subreddit submissions cover everything from potential recruits playing in the NCAA and for international teams, game threads for all NBA games played, breaking NBA news, speculative questions by redditors, light-hearted jokes, and everything in between.

Unsurprisingly, one of the most well-known players currently playing in the NBA, LeBron James, is the direct or an indirect focus of many forum submissions. His popularity and the general NBA fandom's sentiment of him has knowingly ebbed and flowed over time. For example, when he announced he was leaving the Cleveland Cavaliers on a live television special in 2010 to play in Miami, the general sense was people viewed him negatively as it was perceived he was "teaming up" with other good players. On the other hand, he has had moments where the public has viewed him in an overwhelmingly positive light, such as when he contributed to building an elementary school in summer 2018.

The **goal** of this project is to perform NLP techniques on the reddit comments to glean interesting insights. Specifically:

- Topic Modeling - derive topics from the comments
- Sentiment Analysis - sense the polarity of the comments
- Time Series - overlay time series with the initial two goals to see any changes over time and how they map to real-life events

Tools

- Python:
  - Data cleaning: Pandas, Joblib
  - Data analysis: Numpy, Pandas, Scikit-learn, Jupyter
  - NLP tools: NLTK, Scikit-learn
  - Presentation: Matplotlib, Seaborn
  - API: Reddit API, PSAW for Pushshift, PRAW
- Google Slides

Data

Reddit has an API I intended to use to gather all my comments. PRAW is a popular wrapper that makes accessing the API simple. Shortly after using PRAW, I learned that reddit only allows the most recent 1000 posts from a subreddit to be obtained. As I hoped to reach back to 2008 and potentially 10,000s or 100,000s of posts, this would not do.

I identified two possible solutions. One was using Google's BigQuery to obtain the data I wanted. There is a maintainer who collects and stores all reddit submissions and comments in a SQL database available there.

I chose my second option, which was using Pushshift.io. Pushshift is a service that catalogs more reddit info than the reddit API uses. Combining a Python Library for interacting with Pushshift called PSAW, I was able to download the all the comments I wanted from 2011-2018. Unsure why I could not go back to 2008, but the 45,000 comments I collected was enough data to work with.

comment_text	created_time	reddit_score	comment_author	parent_submission_text	parent_author
string	datetime	int	string	float	float

Results

I preprocessed the reddit comments by standardizing punctuation and case, tokenization, reducing to English language words, removing stop words, and stemming the remaining words. Afterward, I created a CountVectorizer matrix from the pre-processed text and performed Latent Dirichlict Allocation (LDA) to discover topics for the comments. Eventually I settled on 16 LDA topics. While these were far from perfect - about half of the groups I could not pull meaning from whatsoever - I did identify some clear groups such as "best player ever", "free throws", "greatest of all time comparison / michael jordan / kobe bryant", and "best player in the league".

Additionally, I then performed a KMeans Clustering unsupervised algorithm to the document-topic matrix output by LDA. Based on inertia inflection, the best # of clusters was 17 - quite similar to the number of LDA topics. Upon checking the documents in these clusters, some of the clusters directly matched LDA topics. Notably, the largest cluster was full of jokes/puns/quips.

Finally, I performed a sentiment polarity analysis on the dataset using the NLTK Vader implementation. It showed remarkable consistency across time of the comments not tending to carry much positive nor negative feelings. However, in June 2012 there was a noticeable uptick in positive feelings; this also is the time when LeBron James won his first NBA championship.

What I would do differently

I would change my dataset constraints. The first change would be to extract all highly up voted comments from r/nba that contained LeBron or a variation of his name. Secondly, I would require all comments selected have at least 25 or so words. I believe this would filter out many of the jokes/puns/quips/one-liners that occupied much of my dataset and for my use case essentially amounted to unwanted noise.

With this new data, I would want to improve upon the topic modeling. By either improving LDA results or finding better topics from NMF or other algorithms my comments would be better assigned to like categories.

Also, I would like to spend more time mapping real-life events to a surge in comments within specific categories. This would be an interesting experiment to see how well the NLP techniques capture the real world events being discussed in r/nba.

Finally, think more about a use for a well functioning NLP pipeline with this data. Nothing came to mind during this project, except I was exploring a topic I like very much.