Sentiment Analysis of Twitter and How it Relates to Illini Football

STAT 385 FA2018 - Team Poisseidon

Joe Broton - jbroton2
Braden Gebavi - gebavi2
Brad Gibbons - btgibbo2
Tim Soo - soo3

November 13, 2018

Abstract

The purpose of this project is to explore sentiment analysis of twitter and how it relates to the illini football team. We will be analyzing the tweets from 12/07 to 12/17 with keywords that we thought depited tweets that a Fighting Illini Fan would include. This is in order to see how fans and students view the illini football team and their opinions on Lovie Smith. Lovie Smith has a 6-year contract with the Illini and after 3 years we have failed to make a bowl game.

Contents

1 Introduction

The topic we are addressing is sentiment analysis on live Fighting Illini Tweets. The idea is to analyze live tweets with sentiment analysis in order to further understand people's reactions to the Fighting Illini and Coach Lovie's performance. This particular topic arised naturally from our school and personal enviroments. Given his previous success with the Chicago Bears up north, we hoped Lovie Smith would be the change needed. However, after the Fighting Illini's defeat against the Iowa Hawkeyes, 63-0 and the last game's loss against Northwestern 24 - 14, we wanted to test our suspicions if students/fans were as displeased with the team on Twitter as they are in the conversations we hear around campus. Initially, there seemed to be a buffer period in the first few seasons where the fans gave him the benefit of the doubt, as naturally, people believed it was just going to take some time getting used to the college scene, but we have a feeling that period may be over and fans' patience are wearing thin. We wanted to see fans sentiment on twitter after news of receiving Lovie Smith has been extended two more years.

Our data is a collection of 908 live non-retweeted tweets from Fighting Illini Fans ranging from 12/07 to 12/22. We will be analyzing the data through sentiment analysis. Sentiment analysis requires an understanding of Regex. The reason for this, is to identify patterns, and using these patterns to extract specific portions of tweets that we would like to analyze. Furthermore, Regex helps us also to split and mutate certain tweets, that can further be compared to our positive, negative, and neutral words.

We also will be applying more recent topics, such as RShiny, that allows interactivity with user input. Our RShiny app allows for the user to view our original data set, along with the ability for the user to extract twitter data using their own api access tokens.}

2 Related Work

Noel Bambrick extracted 2.2 million tweets from super 51 and ran sentiment analysis.. It is a very similar idea to ours as it monitors tweets focusing on the volume, sentiment, and team specific fan reactions. The difference with our idea is that we are only analysing the tweets for one team. In this study, he wanted to find out who twitter thought was going to win the game based off of real time tweets throughout the game but in our study we are just focused on the Fighting Illini fans opinions of the team and Lovie Smith on Twitter. He used the AYLIEN Text Analysis API which you have to pay for. Our access from the standard free Twitter API is 250 tweets per query. Our We are using ggplot2 to visualize our data and Noel used Tableau.

We also looked at a sentiment analysis project wherein tweets regarding the Colorado Floods were analyzed. We used this as a baseline for how a good sentiment analysis project should look. In accordance with that, we looked at multiple articles describing the general process of collecting tweets through R and Twitter's API.

We also used information from the "Text Mining with R" book in which different general-purpose lexicons categorized words under different sentiments, giving us more precise data than if we created our own lexicon.

3 Methods

In order to run sentiment analysis, we will first need to collect the data set. To do this, we need to use the 'searchTwitter' function using the 'twitteR' (Gentry 2015) package. Using this function allows us to to search through tweets with keywords that we feel Fighting Illini fans would include within their 140 character limit tweet. The key query terms we searched for were 'lovie smith', 'illini football', 'loviesmith', 'illinifootball'. The search function when you include a space, makes sure that both words are included in the tweet. By removing the space between the keywords, it allows us to be able to search for that extract phrase which means we pulled tweets with that extract phrase and those with the hash tags associated with the phrase (i.e. #loviesmith, #illinifootball).

Our way to approach the word summary was to split each tweet into individual words. In order to clean the data we had to use the stringr (Wickham 2018) package's function of str.split in order to clean up the data by removing unneccesary special characters, and links by various regex patterns. The sentiment analysis is based on all works appeared in all tweet. This was accomplished by the tidytext::unnest_tokens function (Silge and Robinson 2016). Then the words are merged with the selected sentiment lecixons in the sentiments dataset supplied by the tidytext (Silge and Robinson 2016). To run this process smoothly, we used dyplr (**???**)package's function of inner_join with the associated selected sentiment lecixons on the RShiny app interace. The top positive and negative words that were captured using the Bing sentiment lexicon are displayed in the Bing Words section of the graphs.

In order to add interactivy to the RShiny app, we display the data captured within the 'AFINN' sentiment lexicon. This allows for Users to change the positive and negative values (-5,5) for words they may feel have more weight than what the lexicon states. For example, the data we collected stated that both good and great appeared 38 times, while both scores in the lexicons contribute 3 units. If the user believes great should hold more weight than good, a user can click on new score column, and change great to the value of 4 units. This is shown respectively at the Score Distribution section and the User Score Distribution updates as a user changes score values.

As the user clicks on sentiment lexicon dictionary choices, the rshiny updates with the Sentiment Lexicons graph of each selected choice(s). The graph shows the total amount of scores per tweet.

In the "By Tweet" section it displayed the most positive and negative tweets. A score is calculated for each tweet. We must first call the NRC sentiment dictionary to calculate the presenece of eight different emotions and their corresponding valence, "anger", "antipication", "digust", "fear", "joy", "sadness", "surprise", "trust", "negative", and "positive". The sentiment values are then assigned with the relevant functions of syuzhet::get_nrc_sentiment, and syuzhet::gent_sentiment. Once the sentiment values are determined, we then get a measure of the overall emoitonal valence in the tweet.

All our plots were created using the ggplot2 packages (Wickham 2016)

4 Results

5 Discussion

6 Conclusion

References

Gentry, Jeff. 2015. *TwitteR: R Based Twitter Client.* http://lists.hexdump.org/listinfo.cgi/twitter-users-hexdump.org.

Silge, Julia, and David Robinson. 2016. "Tidytext: Text Mining and Analysis Using Tidy Data Principles in R." *JOSS* 1 (3). The Open Journal. https://doi.org/10.21105/joss.00037.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. http://ggplot2.org.

———. 2018. *Stringr: Simple, Consistent Wrappers for Common String Operations.*