Twitter Sentiment Analysis Surrounding Sports Events

EECS 4415

Abdullah Basulaib
basulaib@my.yorku.ca
215971716
basulaib

Edwin Gonzalez Dos Santos
edwin96@my.yorku.ca
214158893
edwin96

Paul Kim
pyjaekim@my.yorku.ca
211702917
pyjaekim

Amir Younesi
kxn@my.yorku.ca
214782304
kxn

ABSTRACT

The purpose of this project is to gauge an understanding of social sentiment across sports games, this is achieved by viewing how fans and viewers react to events live over the course of a game through Twitter [4]. Sentiment analysis allows us to identify whether the writer's post towards a match is positive, negative, or neutral.

Averaging the sentiment value over the course of the game gives an estimation of how viewers are reacting to the game. This data can then be used for multiple analytics such as: social behaviour, ad placements, pinpointing where the changes of attitude may occur and betting predictions.

In order to get the tweets that relate to the sports event, we used Twitter's API [4] to search for specific tweets containing hashtags relating to the given event and is categorized on 3 sets: a home team, away team and a neutral non team hashtag used for comparison. With the use of Natural Language Toolkit [2], we were able to convert the tweets into a sentiment value for further analysis and plotting of data.

Once we have aggregated our data, we can make a graph that shows the average sentiment throughout the game. This graph can help us identify sentiment spikes and fan patterns.

KEYWORDS

Sentiment analysis, data analysis, tweets, sports

1 INTRODUCTION/MOTIVATION

The project is about gathering and analyzing public sentiment over the course of a sports event using Twitter [4]. We consider this subject worthy of rigorous data analysis due to supporter sentiment often being an overlooked statistic in the sports world. This application attempts to better understand if supporter sentiment fluctuations are the only result of major events in the field of play or if they can be used as a predictive measure of events that are yet to come.

The utility of this application can be used to better understand why supporters sentiment changes, and could also be used by the management of a sports team to address negative fluctuations in supporter sentiment and attempt to correct that before it gets too negative, for example.

The data domain for our project will consist of the textual content of a series of tweets containing one or more of the user provided hashtags along with their respective timestamps. We will also be using NLTK [2] to gauge sentiment for the words that are used in the tweets. The goal of our project is to analyze the textual content of tweets that use a hashtag related to a sports event to map out the average sentiment over the course of the sports event. The tweets will be divided into upto 3 groups, those who are supporters of the home team, supporters of the away team and those using a neutral hashtag, that represents the game itself.

This will allow us to judge the sentiment of different sets of supporters independently. Being able to understand public sentiment via word analysis is important because it allows outside forces to better understand a group without having to rely on them filling out a survey or waiting for a groups sentiment become so negative that it begins to demonstrate itself in other ways beyond their word choice in tweets

2 DATA AND DATA ANALYSIS

As mentioned earlier, the data domain is the textual content of a series of tweets that contain the hashtags and the timestamp. Since the data is coming from Twitter, there is no guarantee that the data will be structured, so the data will be unstructured meaning it contains text and possibly pictures or videos.

The volume of the data will vary. We expect that special events, such as rivalries (e.g., Manchester City vs. Manchester United), will attract high volume of data that can be in the gigabytes. Regular games during the season, we expect that will have small volume, up to medium.

The velocity of data will range as well. We expect that during special events, as mentioned above, we will get very high data velocity, as in hundreds of updates per second. We also expect that other games will have high data velocity, as in tens of updates per second. Once the data arrives, the files that contain the tweets are updated in real-time as the data arrives, after removing any irrelevant information from the tweet such as pictures, username, and converting emojis to their unicode strings.

The quality of the data will vary from high quality data, as in tweets that are meaningful and related to the event, to low quality data, tweets that are made by spam bots or tweets that only contain pictures or videos. High quality data can be handled in an automated fashion while low quality data will not provide any results to us.

We expect that the processing time for an event will take the duration of the event plus short processing data time. A soccer game will approximately be 90 minutes for the two halves, plus a 15 minute break between halves and extra stoppage time that varies from one minute to seven minutes.

The tweets will be arriving at real time during the course of the event. Once a tweet is captured, there is a cleaning and transformation process before it is saved. The tweet JSON object we retrieve from the Twitter API [4] using Tweepy [8] contains a lot of fields that we don't have use for such as location, username, number of retweets, number of likes, etc. We clean those fields out and we only capture the text of the tweet, the timestamp of the tweet and hashtags in the tweet.

After cleaning the data, we transform the data into a schema that is used across the project to ensure consistency when dealing with data. The schema has three fields that we need, the timestamp, the text and the hashtag. Each field is separated by a tab, starting with the timestamp and ending with the hashtag. The transformed data is stored in this schema.

As mentioned earlier, we are performing sentiment analysis on the data we collect. Therefore, we have employed NLTK's [2] sentiment analyzer. This type of analysis fits our data and our problem. Since we are looking for the average sentiment, NLTK [2] provides us with a very useful tool to achieve this. Along with the data cleaning and transformation necessary, we are able to answer the questions we have.

3 ARCHITECTURE OF PROPOSED SOLUTION

The application starts by allowing users to select the ID of the league from which they will be capturing games once the league has been chosen they need to choose the home and away team by team ID. That information is used to query the Sports API [6] that will return some details about the match that will be used later on such as the start time and the fixture number. The user will also be prompted to enter 3 twitter hashtags that will be analyzed the first is the hashtag that is being used by the home team, the second is the hashtag that is being used by the away team and the third is a neutral hashtag that is not connected to any team in particular. For example during the match played on 12-15-2019 between Manchester United and Everton the the home team Manchester United used the hashtag #MUFC, Everton used the hashtag #EFC and the neutral hashtag was #MUNEVE. Once the user has entered all the data needed to collect information from a game they will have the option to select another game or start collecting tweets.

Figure 1 shows the overall architecture of this project. In order to get the tweets relating to the sports event we used Twitter's API [4] to capture tweets as the are posted over the course of the game using at least 1 of the previously provided hashtags. Every 5 minutes after the start of the game all of the textual content of the tweets tweets captured in that 5 minute window will be saved as a json file within a folder specific to that game named using the neutral hashtag. Since the application can capture multiple games at once each file is placed in its respective folder. 15 minutes after the conclusion of the game all of the captured tweets are grouped into one csv file for further processing. A second script is run to breakup that csv file into 3 csv files one per hashtag listing the time they are posted and the content of tweet.

Once that is completed, the content of each file is run through a language analyzer to get the sentimental value of the text. This will allow us to assign a value from -10 to 10 for each tweet. All of the tweets over the course of a 5 minute window will then be averaged out to get the average sentiment within a time window per each hashtag. Those results are placed in a csv file that will be used to generate the final graph.

Before the final script is run another call is made to the API to get all of the events for the now completed match and that result in combination with the csv file is used to generate the graph that will be produced as the final output. The final graph is presented as a line graph showing the fluctuations in sentimental value of the tweets that are posted within 5 minute intervals over the course of the game. To provide context to the fluctuations they are accompanied by makers in the table that show when events take place. Those events include goals, yellow cards, red cards, when the first and second half start and end.

Some of the limitations with our application are that the Sports API [6] does not list calls which are overturned due to video review. We found that calls of that nature typically correlated very strongly with fluctuations in fan sentiment but we have no way of representing that using the current API. Future iterations could take advantage of a more detailed API that does provide those details. Another limitation that we encountered was that not every person tweeting about the game would use one of the provided hashtags or if they did some would spell them incorrectly meaning that it would turn up in our stream. This could be corrected in a future iteration by capturing some of the more common misspellings of the hashtags, capturing replies to the twitter account of the teams in question, or tweets that used the name of the team or player. Additionally our application does not take into consideration content of any additional media that may be included in a tweet such as videos, .gifs or pictures. Future iterations should look for a way to integrate such media into its analysis because they could help to clarify the sentiment behind ambiguous tweets.

The majority of the challenges that we faced are related to sentiment analysis portion of our program which in some cases struggled to fully understand the sentiment behind a tweet. In some cases that was due to the tweet being ineligible due to multiple spelling errors or due to the tweet using references which require the reader have some background knowledge to understand.

4 EVALUATION/RESULTS

We test our work on the current running Premier League season in soccer, where we run collect data on the weekly fixtures. The results we expect to get is a graph that lists the average sentiment of tweets of the 3 hashtags we are interested in, home, away and neutral. Sentiment will be rated on a scale from -10 to 10 where 10 indicates that a certain hashtag has high positive sentiment and -10 is the opposite.

This graph would also list important points in the game as markers on the graph represented by a vertical line at the time which the event took place. These markers will serve to contextualize any fluctuations in the results. What is considered an important event will vary by sport, for example, in soccer an important event can be a red card or a penalty kick or a video assisted referee call.

The analysis done here to get this chart as mentioned earlier is sentiment analysis on tweets to get the average throughout the game. The analysis was done on the data collected throughout the game. The data collected are tweets we are interested in that meet our criteria, basically if they include the hashtags we are looking for. The dataset, as mentioned earlier, is cleaned and transformed before being saved.

Figure 2 shows a graph of a soccer game between Manchester United (abbreviated as MUN) and Everton (abbreviated as EVE) on the 15th of December, 2019. As it can be observed, in the beginning of the match, MUN fans had higher positive sentiment than EVE fans. This data is quite useful, for example, telemarketers can predict that if this pattern occurs, it would be a good idea to queue an advertisement. Additionally, after the positive spike, it can be observed that a negative spike follows it. Another finding is that teams can use this opportunity to promote a contest or a giveaway to attract more fans and help increase the confidence in the current fans.

Note that this is a regular game during the season and hence, a graph like this is expected since this is a weekly occurrence. As discussed earlier, games during playoffs period or during cups and tournaments (especially the Champions League, the biggest sports cup in Europe, or the World Cup), will attract, we approximate, over three times the amount of fans and viewers. Therefore, the volume and velocity of the data will increase as well. This means that the graph will have much more, possibly, spikes and fluctuations that it could lead to interesting results.

5 CONCLUSIONS

From our results, it seems that our solution is adequate for the analysis we wanted. Of course, improvements could be made – extra data cleaning, more hashtags to encompass each category (home, away, neutral). A possible extension to our hashtag selection method could be calculating the tf-idf (term frequency-inverse document frequency) of words with tweets related to the event, and using words with a high tf-idf value for the corresponding hashtags.

There are various ways we could scale this project. For example, we could combine our analysis with a real-time betting odds API to see how bets sway depending on tweets. An interesting extension to the project could be predictive analytics of betting odds using only sentiment analysis of tweets for the event. For instance, will the increase in positive sentiment to a particular team also increase their betting odds? If the sentiment change is sudden or dramatic, does that scale accordingly with how the bets are being placed?

The beauty of this project is the ability to generalize the sentiment analysis to a wider or different scope—for example, political analysis. There has been a surge in Twitter usage among politicians and their political campaigns. By tracking hashtags and the sentiment surrounding those hashtags, political campaigns could better understand how significant events change affect the sentiment of the public.

This project can also extended to other sports, assuming the proper configuration has been applied (such as appropriate API, time interval, key events, etc). It can also be expanded to include other platforms such as Instagram and Facebook as well.

6 REFERENCES

[1] Python Docs - http://docs.python.org/3/

[2] Natural Language Toolkit (NLTK) - https://www.nltk.org/api/nltk.html

[3] Python Pandas - https://pandas.pydata.org/pandas-docs/stable/

[4] Twitter API - https://developer.twitter.com/en/docs

[5] Tweepy - https://tweepy.readthedocs.io/en/latest/

[6] Sports API - https://www.api-football.com/

[7] Python Matplotlib - https://matplotlib.org/

[8] Streaming with Tweepy - http://docs.tweepy.org/en/latest/streaming_how_to.html

[9] Python JSON - https://docs.python.org/3/library/json.html

[10] Convert Twitter Timestamp to Epoch - https://stackoverflow.com/questions/18604755/twitter-created-at-convert-epoch-time-in-python

7 FIGURES

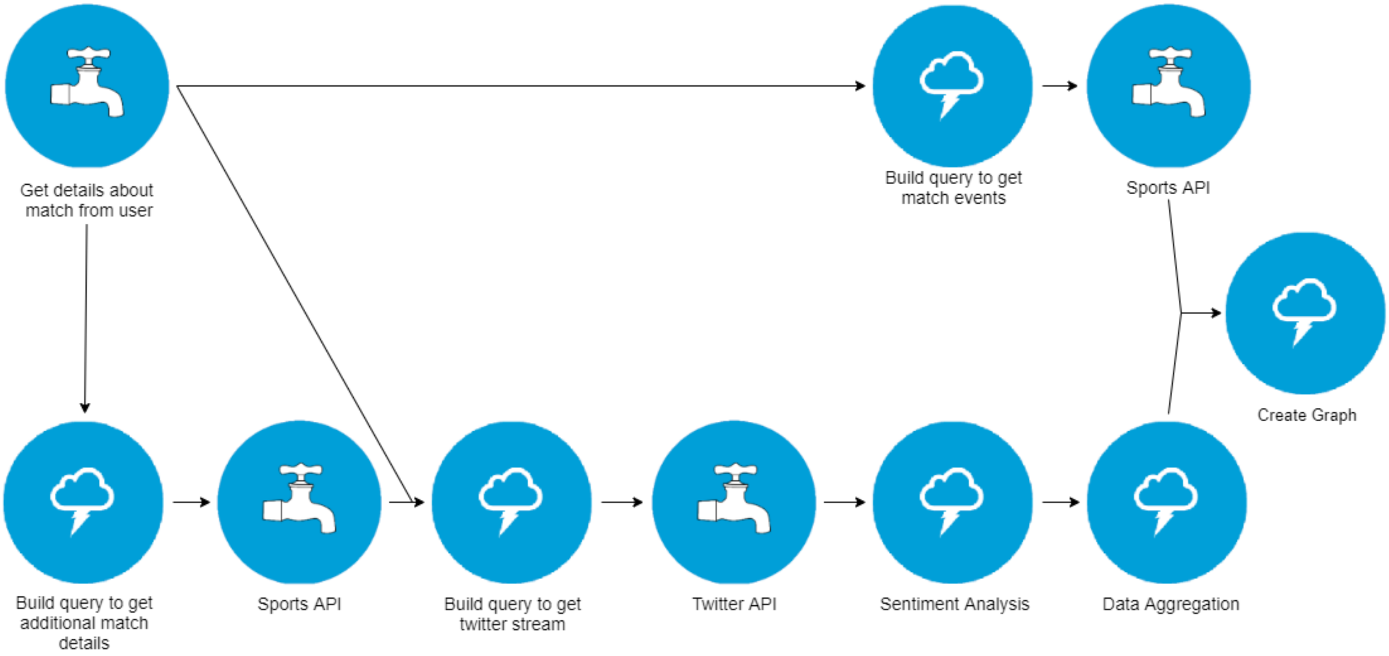
Get details about match from user
Build query to get match events
Sports API
Create Graph
Build query to get additional match details
Sports API
Build query to get twitter stream
Twitter API
Sentiment Analysis
Data Aggregation

Fig. 1

Overall architecture and data flow

Figure 1 shows the overall architecture of this project

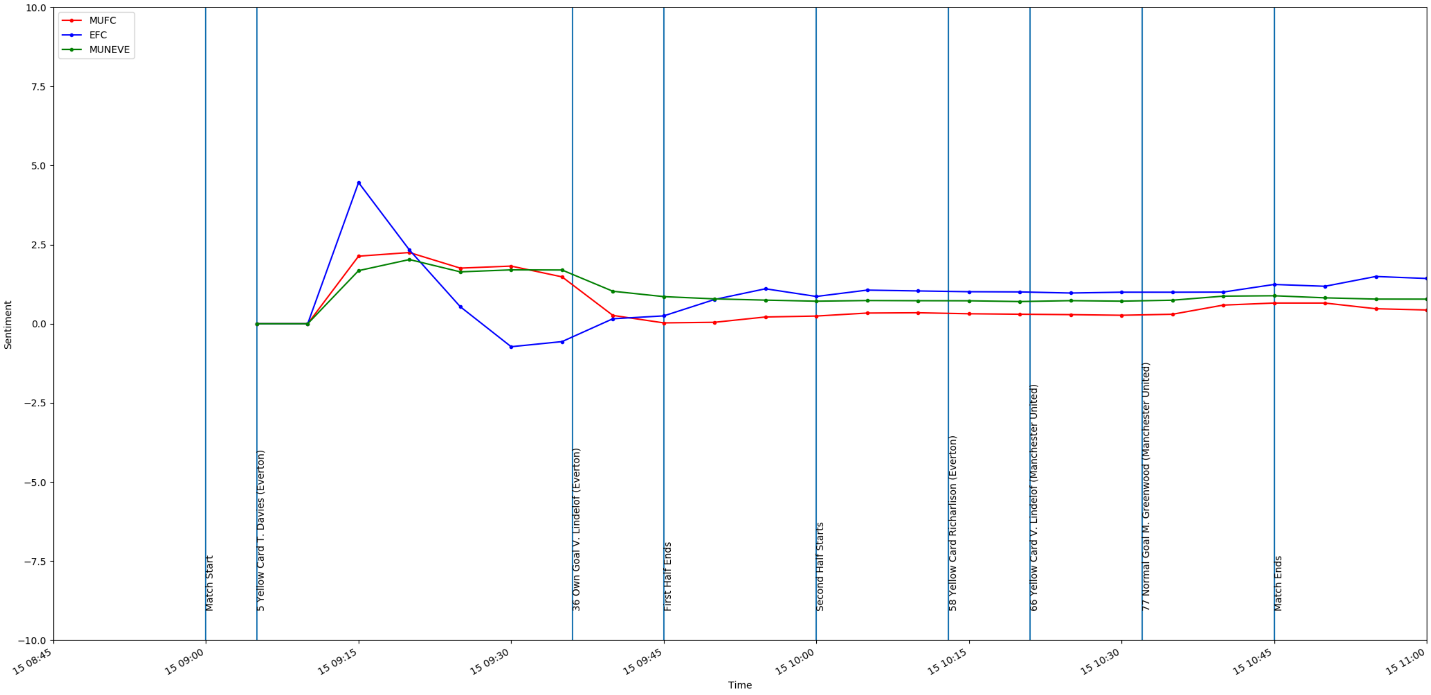

Figure 2 shows a graph of a soccer game between Manchester United (abbreviated as MUN) and Everton (abbreviated as EVE) on the 15th of December, 2019