Project_4_Part2_Twitter_Data

Project 4 - Part 2 Twitter Data¶

Gorkem Camli (105709280)

Library imports¶

```
[nltk_data] Downloading package vader_lexicon to
[nltk_data]     /Users/gorkemcamli/nltk_data...
[nltk_data]   Package vader_lexicon is already up-to-date!
```

Read Data¶

Question 27¶

Report the following statistics for each hashtag, i.e. each file.

- Average number of tweets per hour
- Average number of followers of users posting the tweets per tweet (to make it simple, we average over the number of tweets; if a users posted twice, we count the user and the user's followers twice as well)
- Average number of retweets per tweet

Since the time to read all files were taking,I solved this issue by reading files line by line and then only saving the columns I need for the questions 27 and 28. Below we can see the shape of each dataframe created for all 6 files:

```
gohawks (169122, 4)
gopatriots (23511, 4)
nfl (233022, 4)
patriots (440621, 4)
sb49 (743649, 4)
superbowl (1213813, 4)
```

Average number of tweets per hour: For the stats questions, the average number of tweets per hour can be defined different ways:

1 - Find number of tweets tweeted all the hours in data set, take their average.

2 - Total number of tweets / (Number of hours within the textfile's min and max date)

For these definitions the results can differ if there are hour timeframes where no tweet has been posted. (2) definition takes into account the hours that have no tweet, (1) does not. For example if the within the data timeframes there are 100 hours but 2 hours have no twitter post, (2) divides the total tweets by 100, however (1) divides by 98. The second approach makes more sense and accurate, and I choose this as definition. But since the question is not clear on which definition we are expected to go I include both of them

Average number of followers of users posting the tweets per tweet: Mean of followers_count column (['author']['followers'])

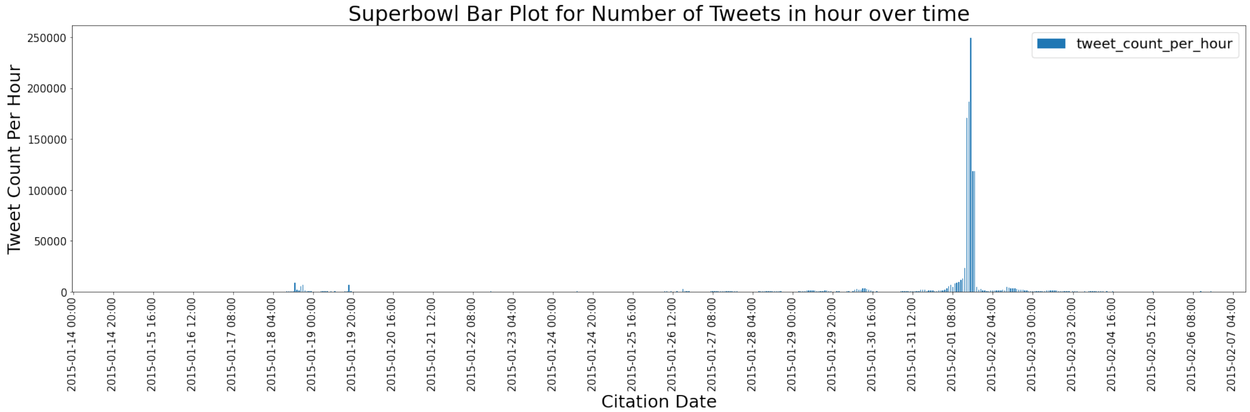
Average number of retweets per tweet: Mean of retweet_count column (['metrics']['citations']['total']) as every row represent a tweet.

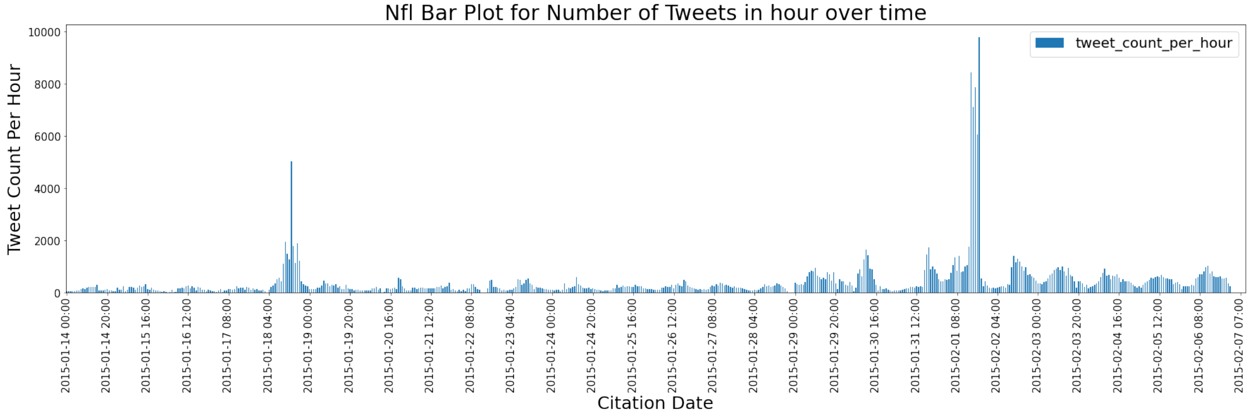
```
--- Stats for #gohawks ---
Average # tweet per hour (definition 1:) 296.705
Average # tweet per hour (definition 2:) 292.488
Average # followers 2217.924
Average # retweet per tweet 2.013


--- Stats for #gopatriots ---
Average # tweet per hour (definition 1:) 53.313
Average # tweet per hour (definition 2:) 40.955
Average # followers 1427.253
Average # retweet per tweet 1.408


--- Stats for #nfl ---
Average # tweet per hour (definition 1:) 399.695
Average # tweet per hour (definition 2:) 397.021
Average # followers 4662.375
Average # retweet per tweet 1.534


--- Stats for #patriots ---
Average # tweet per hour (definition 1:) 750.632
Average # tweet per hour (definition 2:) 750.894
Average # followers 3280.464
Average # retweet per tweet 1.785


--- Stats for #sb49 ---
Average # tweet per hour (definition 1:) 1384.821
Average # tweet per hour (definition 2:) 1276.857
Average # followers 10374.16
Average # retweet per tweet 2.527


--- Stats for #superbowl ---
Average # tweet per hour (definition 1:) 2067.825
Average # tweet per hour (definition 2:) 2072.118
Average # followers 8814.968
Average # retweet per tweet 2.391
```

Question 28¶

Plot "number of tweets in hour" over time for #SuperBowl and #NFL (a bar plot with 1-hour bins). The tweets are stored in separate files for different hashtags and files are named as tweet [#hashtag].txt.

Plots below show the number of tweets in hour for #superbowl and #nfl data in 1-hour bins. X axis ticks shows the corresponding day and hour for that bin. Since there are many bins and corresponding ticks for them to make the plot less crowded, I showed xticks in every 20 hours.





For superbowl hashtag we can see that the peak is in Feb 1st 2015 on the superbowl 49 game day. For nfl the biggest peak is again on superbowl day, Feb 1st but there are also smaller peaks happening on other days, which probably also corresponds game days. For example the first small peak, on Jan 18 2015 corresponds the day where Packers and Seahawks have game.

Question 29¶

Task Explanation¶

Follow the steps outlined below:

- Describe your task.
- Explore the data and any metadata (you can even incorporate additional datasets if you choose).
- Describe the feature engineering process. Implement it with reason: Why are you extracting features this way - why not in any other way?
- Generate baselines for your final ML model.
- A thorough evaluation is necessary.
- Be creative in your task design - use things you have learned in other classes too if you are excited about them!

Intro & Task Description:

In this question, I did character-centric time series tracking and a winner prediction task:

- Character-centric Time-Series Tracking part, intends to understand how tweet emotions changing during the Superbowl 49 game for hawks and patriots, as well as for American Football players playing in the game. So thee question I want to answer "Can we track the average perceived emotion across tweets about each player in the game across time in each fan base? Can we correlate these emotions with the game score in the given time interval?".

- Winner Prediction Task: Given tweet emotions expressed to players within a time window, can we predict who is winning the superbowl 49 in that time window? The emotions about player or the team is aggregated based on 1-minute time intervals. So, given the perceived emotion in any minute in the game, who is leading the game, the answer could be Tie, Patriots or Seahawks. I will do 3 class classification task with different emotion features on 3 models (Logistic Regression, Random Forest, Neural Networks -simple MLP-).

These two tasks are not trivial to solve. In the raw data:

- There is no information on which tweet belongs what player.
- There is no information about players or the game.
- We only have timestamp for tweets but we need to map these to the game time where significant events such as score occurs.
- There is no label for the prediction task. We don't know the game status in the tweets data.

In the next sections Data Exploration and Feature Engineering, I explained in details how I tackled these problems.

Data Used:

- '#gopatriots.csv', '#patriots.csv' and '#gohawks.csv' files.
- Additional data:
  - Players information: I collected team player names, positions, position category information for Patriots and Seahawks team members who played at Superbowl 49.
  - Game information: I used the score plays information to extract relative score times from tweets and label which team is the leading the game (patriots, seahawks, tie) when a tweet posted.
  - Emotion data: I used 4 different emotion feature set extracted from different libraries/models: **Nltk.sentiment.vader, TextBlob, CardiffNLP Roberta and AllenNLP Roberta.** I did both tasks with 4 set of features and compared the feature results as well. I chose nltk.sentiment.vader and Textblob as the baseline features, the features extracted with these methods are rather simple. Nltk vader[1] and Textblob are both rule-based sentiment analysis approaches. Whereas sentiment features extracted from Roberta (Robustly Optimized BERT Pre-training Approach) Models use advanced NLP deep learning models which promises better emotion features. CardiffNLP is especially interesting because it is pretrained on Twitter dataset.

Data Preparation¶

Set GAME_START and GAME_END times¶

```
GAME_START 2015-02-01 15:30:00-08:00
GAME_END 2015-02-01 19:06:00-08:00
```

Helper functions¶ (to prepare data extract features, feature engineering and prepare the dataset to be used in the tasks)

Prepare Twitter Data and Additional data¶ To be able to create the tasks mentioned above, I need to first prepare the data, but some of the informations I need is not available within the Twitter dataset provided.

- Find Extra data:
  - Finding players data for each team
  - Finding game related scores, and significant events

Players Data¶ I used players and american football positions data found from below resources. I also crosschecked the team players info from espn website and added the missing players. Also, I realized that some players such as Dan Connolly didn't play in 2015 superbowl 49 (according to ESPN), so I created a new column espn_team_list to keep track of the players who played/not played in SB49.

References:

- players data: https://www.pro-football-reference.com/boxscores/201502010sea.htm#all_team_stats
- positions data: https://www.rookieroad.com/football/positions/position-abbreviations/

The final combined players table preview:

```
(39, 9)
```

	Player	Pos	Team	espn_team_list	Abbreviation	Position	Category
0	Tom Brady	QB	Patriots	True	QB	Quarterback	Offense
1	Shane Vereen	RB	Patriots	True	RB	Running Back	Offense
2	Brandon LaFell	WR	Patriots	True	WR	Wide Receiver	Offense
3	Julian Edelman	WR	Patriots	True	WR	Wide Receiver	Offense
4	Rob Gronkowski	TE	Patriots	True	TE	Tight End	Offense

Raw Tweet Data¶ Read gohawks, go patriots and patriots data.

```
Creating tweet_dfs2...
gohawks (169122, 13)
gopatriots (23511, 13)
patriots (440621, 13)
```

Language distribution of the tweets for top 8 language: (und:undecided)

`gohawks`

	en	und	es	pt	de	fr	ja	tl
tweet.lang	0.841801	0.119198	0.011104	0.008976	0.003737	0.001963	0.001904	0.001898

`gopatriots`

	en	und	es	pt	de	fr	ht	it
tweet.lang	0.547999	0.18234	0.129131	0.089107	0.00906	0.008166	0.005614	0.004211

`patriots`

	en	und	es	pt	fr	de	ht	in
tweet.lang	0.898008	0.048788	0.029399	0.005792	0.004142	0.002921	0.001441	0.001203

Given that we will do sentiment analysis task, I will only use english language tweets.

Create Project Data¶ We need to incorporate the newly collected data with the tweets data. This is not a trivial step as we need to know (1) if a tweet is about a specific player, (2) mapping game time to real time. (1) is handled in this section and (2) has its own specific part since it is more complex.

In this step I prepared the final raw data to be used in the above described tasks:

Steps followed to prepare this data:

- Select features from raw tweet data
- Filter to have english language only
- Concat gohawks and gopatriots data in a single dataframe, put hashtag column to later use to separate them.
- Handle date time information for citation_date column.
- Game Status: create a column to specify when the tweet is posted: 'pre-game, during game' and 'after game'
- Map tweets and players
- Sentiment Analysis
  - Nltk Vader
  - Textblob
  - CardiffNLP Roberta
  - AllenNLP Roberta

Further details for some of the above steps:

Selected features from raw tweets data:

['title', 'tweet.text', 'citation_date','hashtag', 'tweet.lang']

Date time:

Twitter raw data has citation_date column as the tweet post time. This column values are in the form of UNIX time as a scalar number. I first convert this column to a human-readable format using datetime and converted the timezone to PST time zone. I also used PST time zone for significant times in the game such as game start and end time to be able to easily filter out tweets during game time. I also created additional year, month, day, hour columns for convenience.

Find if a tweet about a specific player:

In order to make a character-centric tracking we need to know whether a tweet is related to a player. There are several ways we can see a tweet could be related to a player: if tweet text mentions the player, uses hashtag about player and use the name of the player.

I used a simple logic with full name and partial name to match to find tweets about players. The results are case insensitive. By doing this I can identify players for 15% of the tweet data with 1 or more than 1 players out of 551K tweets. So, around 83K of tweets are matched with at least 1 player. This could be further improved by finding twitter usernames of the players (for mentions) and specific hashtags used for players. However, most of these already contains either full, first or last name of the players meaning most of these case would be handled already with the current logic. Several columns created: player (list of the players found), player_count (number of players identified in a tweet), player2 (same as player but with team names for tweets no player found).

What will happen to the remaining 85% of the tweets? Well, since I didn't want to discard them directly I assigned team names to the tweets that I couldn't find specific players.

Sentiment Analysis:

For sentiment analysis, we need sentiment information. 4 different approach used to extract sentiment information: nltk, textblob, CardiffNLP Roberta and AllenNLP Roberta. With nltk.vader, I found positive, negative, neutral and compound scores for each tweet. Assigned the labels using compound score. With TextBlob, I extracted polarity and sensitivity information and assigned the label using polarity information. CardiffNLP I extracted positive, neutral, negative scores and label. With AllenNLP, I extracted positive and negative scores as well as label. Since roberta model results don't have a compound score, I used the positive scores as the feature set (lower positive score would correspond negative emotion).

For the character centric analysis, we need to aggregate these emotion values, for each character and time interval. I described those steps in the later sections. I experimented extracting sentiment information with both cleaned and uncleaned tweets, the change in the results were insignificant. I shared them later in the notebook. Also, applying these steps take considerable time, in order to not rerun the same things over and over, I saved the final df and reread when I restart working.

Preview of the project data:

```
Starting posting time datetime features...
(550932, 72)
```

	hashtag	title
0	#gohawks	“@TheDA53: “@nathanSD8: @TheDA53 broncos?! #...
1	#gohawks	Dr. Jim Kurtz & I before Seahawks vs Panth...
2	#gopatriots	NFL PLAYOFFS: Brady throws 3 TDs in win over R...

Data Exploration & Feature Engineering¶

Answer some questions¶

- **Check if data is balanced for each team?**

```
#gopatriots    408565
#gohawks       142367
Name: hashtag, dtype: int64
```

```
#gopatriots    0.741589
#gohawks       0.258411
Name: hashtag, dtype: float64
```

We have highly inbalanced data, gopatriots tweets corresponds to the 74% of the data.

- **Find in how many tweets each player is mentioned?**

	title
player2	
#gopatriots	357785
#gohawks	115377
Tom Brady	32383
Russell Wilson	8710
Rob Gronkowski	5395
Marshawn Lynch	4845
Richard Sherman	4314
Chris Matthews	2753
Julian Edelman	2439
Earl Thomas	2130
LeGarrette Blount	1948
Doug Baldwin	1541
Darrelle Revis	1342
Kam Chancellor	1010
Vince Wilfork	949
Brandon LaFell	868
Michael Hoomanawanui	737
Jermaine Kearse	646
Bryan Walters	585
Chandler Jones	507
Danny Amendola	491
Jamie Collins	471

	title
player2	
Tony McDaniel	430
Kyle Arrington	405
Shane Vereen	394
Kevin Williams	367
Bruce Irvin	316
Brandon Browner	255
Bobby Wagner	237
Cliff Avril	191
Devin McCourty	163
Patrick Chung	163
K.J. Wright	141
Dont'a Hightower	132
Ricardo Lockette	122
Michael Bennett	118
Byron Maxwell	88
Rob Ninkovich	71
DeShawn Shead	57
Sealver Siliga	45
Robert Turbin	11

- **What is the ratio of number of players found on tweet dataset?**

Show player_count column's normalized value counts.

```
0     0.858839
1     0.116221
2     0.018329
3     0.005231
4     0.001049
5     0.000231
6     0.000069
7     0.000015
8     0.000011
9     0.000004
10    0.000002
Name: player_count, dtype: float64
```

Cleaning Tweets:¶ I did a minimum cleaning on the tweets to experiment on whether text cleaning will further change the emotions distributions for tweets, and help with the model performances.

Cleaned title column: cleans the links and mentions from the tweet. Make all characters lower case.

Cleaned hashtag title column: in addition to links and mentions, also cleans the hashtags from the tweet. Make all characters lower case.

Creating Sentiment Scoring and Label Features¶

nltk.sentiment.vader¶ As my first baseline sentiment analysis score approach, I used nltk.sentiment.vader.

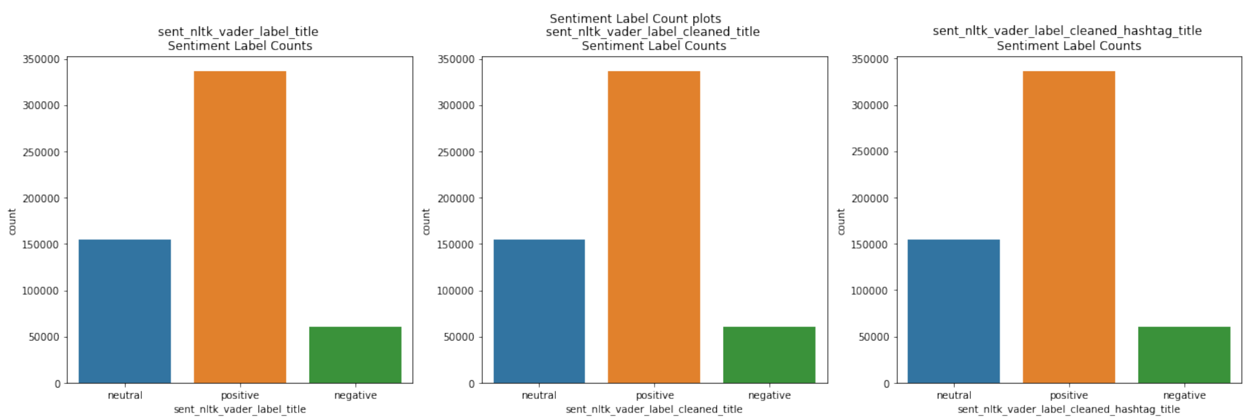
VADER stands for Valence Aware Dictionary for Sentiment Reasoning. Vader is a model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion and one can apply Vader model directly to unlabeled text to get scores for negative, neutral and positive sentiments. Nltk.sentiment.vader also provides a compound score: "is computed by summing the valence scores of each word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). This is the most useful metric if you want a single unidimensional measure of sentiment for a given sentence. Calling it a 'normalized, weighted composite score' is accurate."

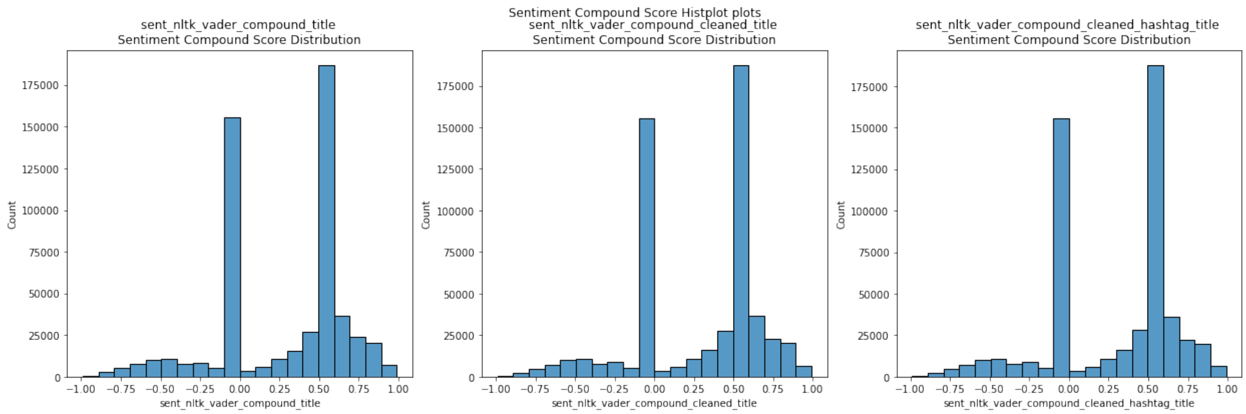
To assign positive, neutral and negative labels, I used the compound score and recommended thresholds in the nltk documentaion: https://github.com/cjhutto/vaderSentiment#about-the-scoring

- positive sentiment: compound score >= 0.05
- neutral sentiment: (compound score > -0.05) and (compound score < 0.05)
- negative sentiment: compound score <= -0.05

I applied Vader model to raw tweets data, cleaned_title and cleaned_hashtag_title columns. When we grouped and check what the ratios for each label, I don't see any huge shift in the overall distribution in emotion for the raw vs cleaned tweets.

```
title

positive    0.610240
neutral     0.280111
negative    0.109649
Name: sent_nltk_vader_label_title, dtype: float64

cleaned_title

positive    0.610313
neutral     0.280203
negative    0.109484
Name: sent_nltk_vader_label_cleaned_title, dtype: float64

cleaned_hashtag_title

positive    0.609850
neutral     0.280681
negative    0.109469
Name: sent_nltk_vader_label_cleaned_hashtag_title, dtype: float64
```

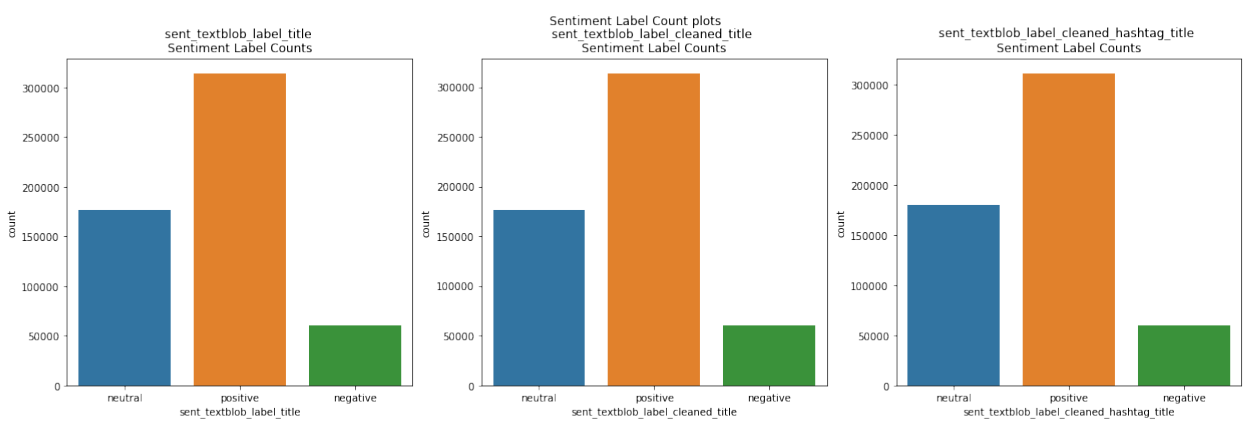


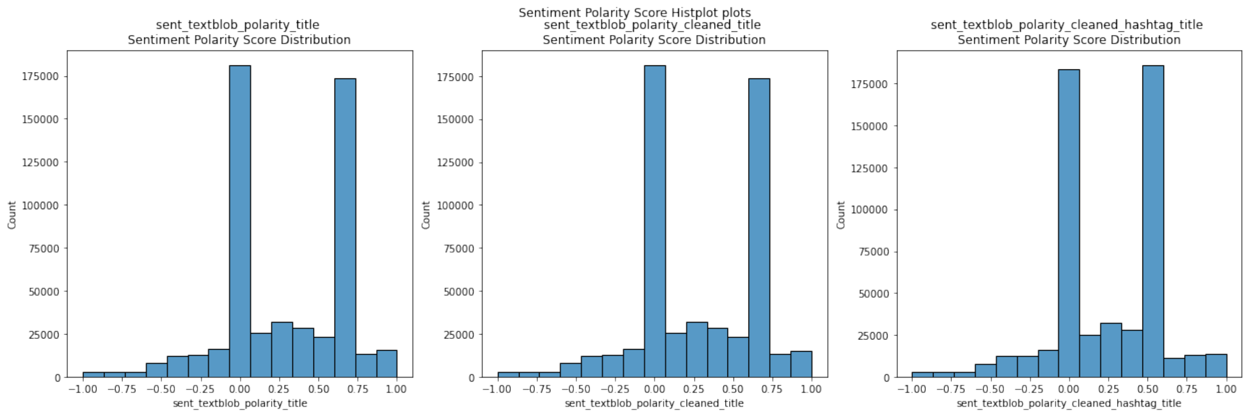


We can see that majority of the labels assigned to positive tag, 60-61%. The compound score has two peaks one at 0 and the other at around 0.5.

TextBlob sentiment¶

```
title

positive    0.569557
neutral     0.320936
negative    0.109507
Name: sent_textblob_label_title, dtype: float64

cleaned_title

positive    0.569103
neutral     0.321288
negative    0.109609
Name: sent_textblob_label_cleaned_title, dtype: float64

cleaned_hashtag_title

positive    0.564556
neutral     0.326089
negative    0.109355
Name: sent_textblob_label_cleaned_hashtag_title, dtype: float64
```





We can see that majority of the labels assigned positive label again, but percentage of positive labels are lower compared to nltk. The polarity distribution is similar to Nltk as it has 2 main peaks, though nltk seems to assign higher scores on positive tweets whereas in this one 0.0-0.75 range has higher bars.

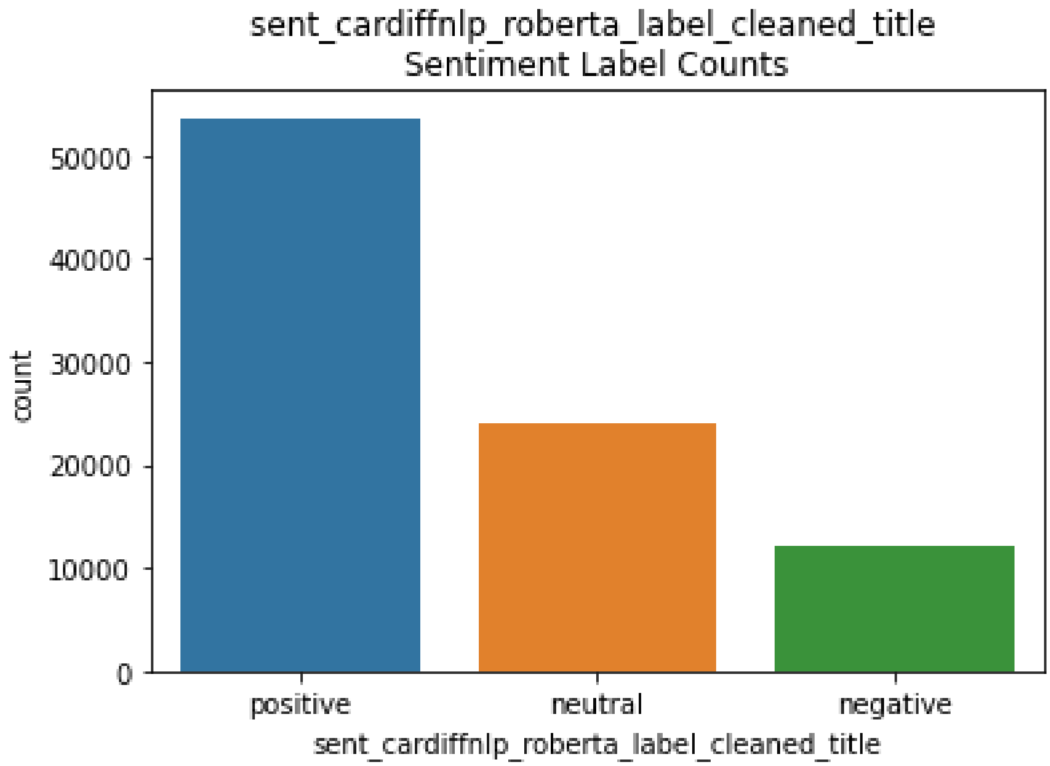
Cardiff NLP Roberta sentiment¶ The code I used to extract features with CardiffNLP Roberta model is in another notebook called "P4_Roberta.ipynb".

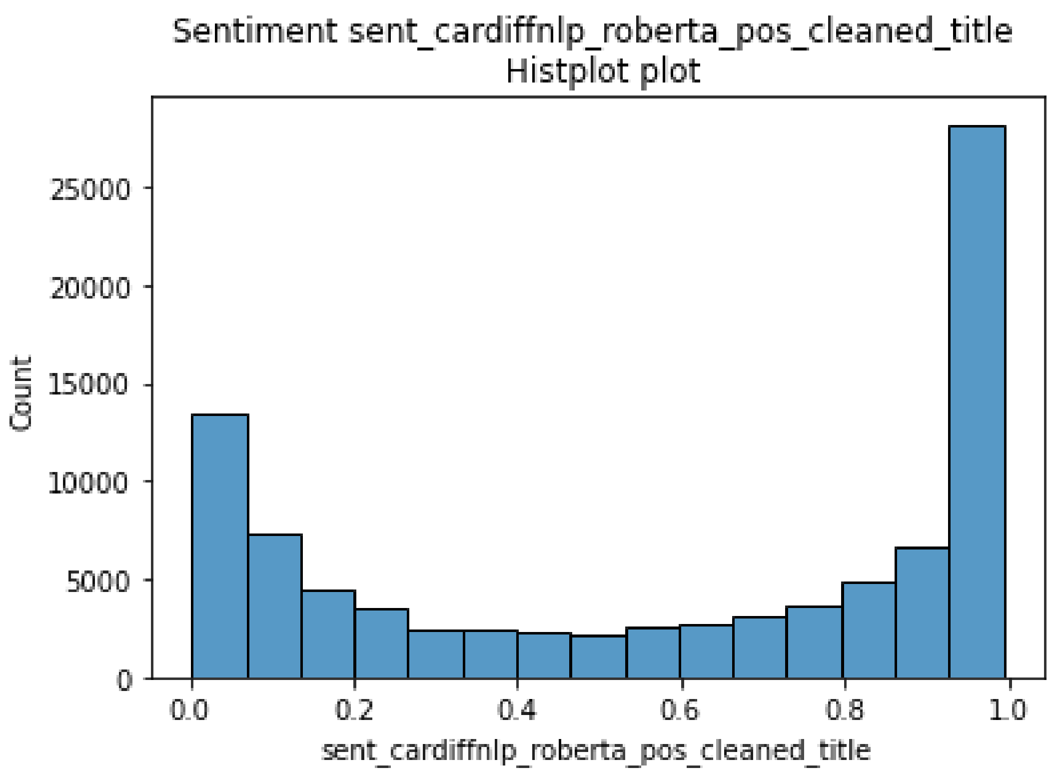
I already extracted the sentiment scores and label from CardiffNLP Roberta using mentioned notebook and saved the results in a csv file. This model is particularly interesting because it is pretrained on Twitter dataset, which I am expecting to see more accurate sentiment score results. 4 columns extracted: positive, neutral, negative scores and label. Since model didn't provide a compounds score I used the positive score in the tracking and prediction task.

I only extracted features for the tweets posted during SB49 game, approx. 90K tweets. Otherwise I have around half million tweets and it was taking too much time to extract them. Also, all my exploration will be mainly focused during game, so this is the most essential information I needed for character-centric time-series tracking and my prediction task. As I didn't see any radical sentiment score changes with the cleaning tweet, I extracted the sentiment results for cleaned_title column only.

```
positive    0.597334
neutral     0.267315
negative    0.135350
Name: sent_cardiffnlp_roberta_label_cleaned_title, dtype: float64
```

Positiive labels again takes the majority of the tweets. We have the lowest neutral label percentage and the highest negative percentage within the 3 first methods. This is a good indication that the extracted features have more polarity.





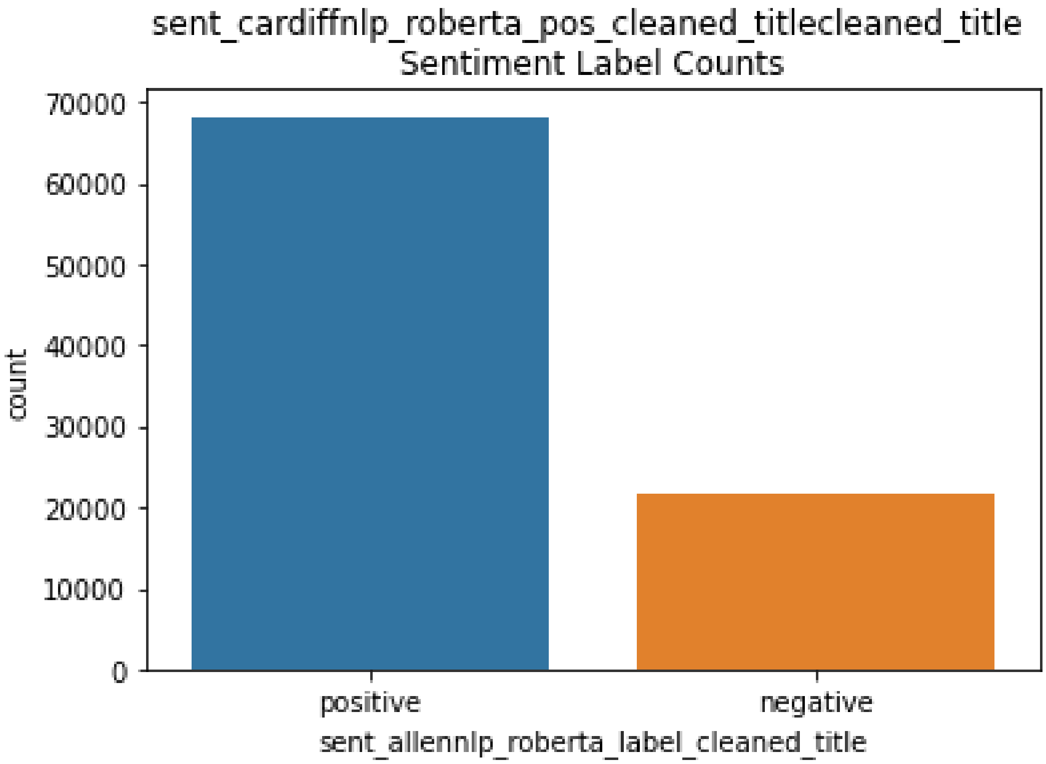
We see there is a lot of tweets that assigned either 1 or very close to 1, and the second peak is towards 0. The polarity between both sides are the most significant so far with the CardiffNLP features.

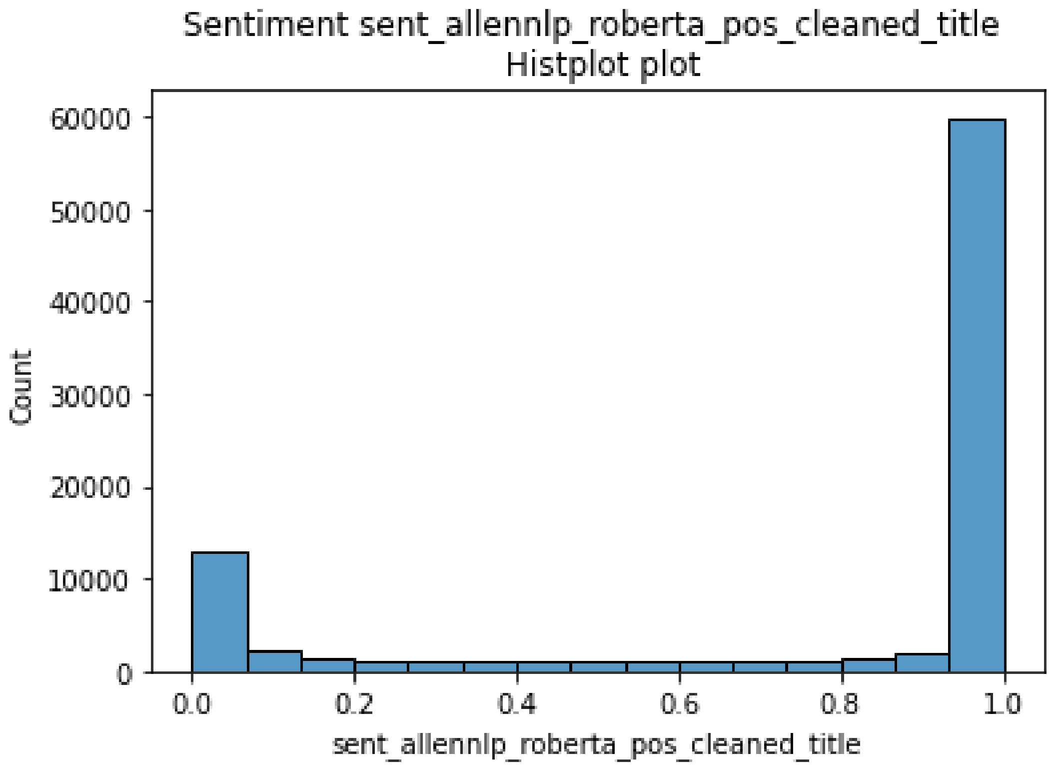
Allen NLP Roberta sentiment¶ The code I used to extract features with AllenNLP Roberta model is in another notebook called "P4_Roberta.ipynb". I already extracted the sentiment scores and label from AllenNLP Roberta using that notebook and saved the results in a csv file. I expect to have an accurate results and good features from this model too. I extracted positive and negative scores and labels.

Similar to CardiffNLP, I only extracted features for the tweets posted during SB49 game and only on cleaned_title. Otherwise I have around half million tweets and it was taking too much time to extract them.

```
positive    0.758964
negative    0.241036
Name: sent_allennlp_roberta_label_cleaned_title, dtype: float64
```

This is a bit harder to compare with other 3 methods as it doesn't have neutral label. Neutral labels in previous features seem to be shared with positive and negative labels. We see the highest negative tweet percentage in all 4 methods.





The polarity is even higher in here. This is a bit concerning, because 75% of the tweets are assigned positive and there the distribution for lower scores seem almost flat and very low. The emotion information for lower scored tweets, might have almost no effect when we aggregate them and get lost within positive scores. Also, it is very interesting that CardiffNLP Roberta and AllenNLP Roberta results are very different even in terms of disitribution. Both provides more polarity and less neutral emotion, but the distribution in CardiffNLP one seems to be more varied whereas AllenNLP one is mostly seem flat except 0 and 1.

Explore Sentiments Examples:¶

```
sent_nltk_vader_compound Top 3 Positive Titles
Ex 1: Score: 0.9906 - Tweet: #patriots win. wow wow wow wow wow wow wow omg wow
wow. #superbowl
Ex 2: Score: 0.987 - Tweet: yesssss, #patriots #win #superbowi !!! great game,
great #halftimeshowkatyperry great sunday!! :-)
happy, happy, happy!! :-) :-) :-)
Ex 3: Score: 0.9843 - Tweet: congratulations  you deserve/earn it! you won!
brady the best! god bless! peace &amp; love! xoxo!
#patriots #brady #superbowl2015!!!


sent_textblob_polarity Top 3 Positive Titles
Ex 1: Score: 1.0 - Tweet: happy friday!! #seahawks #superbowlxlix #gohawks
#nfl #seagals #deflategate
Ex 2: Score: 1.0 - Tweet: best dressed #seahawksfan on #superbowlsunday?
done and done.  #gohawks #12fan #beastmode
Ex 3: Score: 1.0 - Tweet:   a perfect topper to my day. glorious. #gohawks


sent_cardiffnlp_roberta_pos Top 3 Positive Titles
```

```
Ex 1: Score: 0.99376816 - Tweet: i love you so much! thank you for this amazing
experience!  #blessed #gohawks #seahawks
#12thcouple…
Ex 2: Score: 0.9936446 - Tweet: that's my boo  you are beautiful! ahhhh love you!
#gohawks #sb49
Ex 3: Score: 0.9935559000000002 - Tweet:  it was awesome! beautiful! &lt;3 you!! #sb49
#gohawks #12s


sent_allennlp_roberta_pos Top 3 Positive Titles
Ex 1: Score: 0.9999048709869384 - Tweet:  give the strength to the #patriots
Ex 2: Score: 0.9999042749404908 - Tweet: we won ! #patriots
Ex 3: Score: 0.9999040365219116 - Tweet: beautiful #patriots


sent_nltk_vader_compound Top 3 Negative Titles
Ex 1: Score: -0.9897 - Tweet: hell of a game. hell of a finish. hell of an
adjustment. hell of a fight. hell of a comeback.
hell of a situation. hell of a team. #gohawks
Ex 2: Score: -0.9855 - Tweet: no no no no no no no no no no no no no no no no no
you can do this  give them something to cry about
#superbowlxlix #gohawks
Ex 3: Score: -0.9812 - Tweet:  ooohhhh, kill 'em, doug! haters gonna hate, hate,
hate, hate, hate #gohawks


sent_textblob_polarity Top 3 Negative Titles
Ex 1: Score: -1.0 - Tweet: they hate us cuz they ain't us! #patriots
Ex 2: Score: -1.0 - Tweet: game time!! let's go #patriots!!!!
Ex 3: Score: -1.0 - Tweet: the  are boring.  bring out the   #sbmediaday
#gohawks


sent_cardiffnlp_roberta_neg Top 3 Negative Titles
Ex 1: Score: 0.9836120999999999 - Tweet: i hate everything about the #patriots*, it
actually makes me angry just seeing them in this
game
Ex 2: Score: 0.9834545 - Tweet: im gonna puke this is fuckng stupid #patriots
getting jobbed again
Ex 3: Score: 0.98309845 - Tweet: so irritated that i don't get to enjoy the game.
instead i get to go to this stupid class for cps.
#ijustwannawatchfootball #gohawks


sent_allennlp_roberta_neg Top 3 Negative Titles
Ex 1: Score: 0.9997636675834656 - Tweet: second three-and-out for #patriots, who are
looking a bit uninspired.
Ex 2: Score: 0.9997625946998596 - Tweet: argh, stupid stupid foul. bad choice, guy.
#patriots
Ex 3: Score: 0.9997610449790956 - Tweet: this is a very bad place for the #patriots to be.
```

Examples mostly seem to make sense. Also some of the examples are extremely small or complicated to predict. For example: Nltk.vader's first negative example: 'hell of a game. hell of a finish. hell of an adjustment. hell of a fight. hell of a comeback. hell of a situation. hell of a team. #gohawks ' might be wrongly predicted. Depending on what time of the game it is posted this could be interpreted as either positive or negative.

Mapping Game Time to Real Time:¶ There are 3 main time information we need to find:

1. Categorize tweets as pre, during and after game.
2. Find quarter and halftime show beginning and end times.
3. Find score times.
4. Find significant event times (interceptions, fumbles).

One of the biggest challenge I had is to map the real time with the game time. Game scores and significant events data available on the internet is relative to the game time such as first touch down is happend in second quarter 9:47 (example: https://www.espn.com/nfl/playbyplay/_/gameId/400749027 check all plays and scoring plays table). However, tweets are based on real time.

Unfortunately, game clock isn't kept counting down continuously during game. Even though game is considered 1 hour with each quarter 15 minutes, I realized that the actual game took generally 2-3 hours. In the superbowl 49 case, it took 3h 36 minutes as written on the internet. Hence, I need to find a way to map the game time and real time where tweets are written.

I tried several things on how to assign time frames. My first approach was to find timestamps of the game (either significant event or scores), but after hours of search it lead me nowhere:

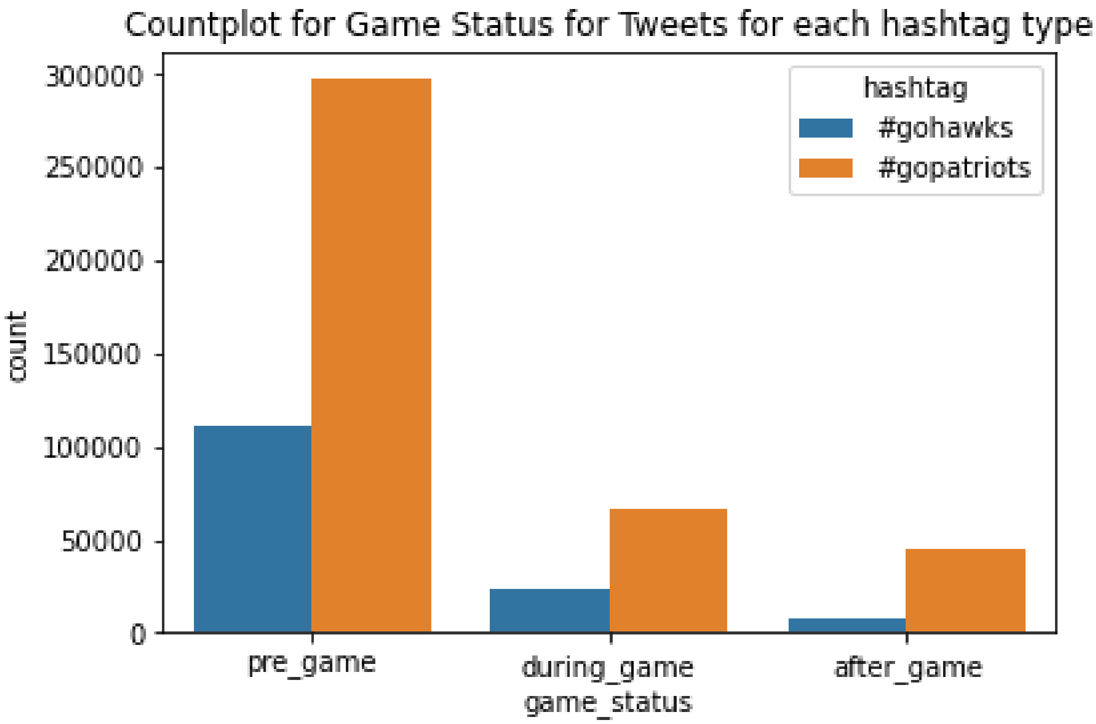
- I couldn't find any data online that is relative to real timestamps like PST, UTC on neither on quarter start and end, nor for significant or score times. Every play by play and score play data I found were on relative to the game time shown minutes within quarters.
- Checked youtube videos of the full game, to extract times manually, either by checking time on the screen or find it relative from video time. In full game video, there is no real time shown on the screen and I realized that the video was shortened where halftime shows removed, so that didn't helped either on mapping the game time to real time.
- I tried to find timestamps from the tweets of official twitter accounts of Seattle Seahawks and Patriots. I check both dataset and then twitter but the tweets from there limited and you can't infer any of the information completely.
- Final resort: refer score and quarter times from tweet dataset. Assumption: if a tweet with score is tweeted than either one of the team is already scored, use the earliest one within the game timeframe.
- As a last resort, I decided to extract the score times and quarter times from the tweets.

PS: I skipped the (4) Find significant event times (interceptions, fumbles), since there is no fumbles in the game and finding interceptions from tweets both very difficult and not very reliable.

1. Categorize tweets as pre, during and after game:¶ We already know the game start time 3:30 PST and how long game lasted 3 hours 36 min. By using start and game end time we can assign pre-game, during game and after game tags to each tweet. The distribution of game status can be seen below:

```
pre_game       408154
during_game     89804
after_game      52974
Name: game_status, dtype: int64
```

```
pre_game       0.740843
during_game    0.163004
after_game     0.096153
Name: game_status, dtype: float64
```



Most of the tweets were posted before game, so American Football fans, especially Patriots and SeaHawks are really fans and they watch the game during game, instead of posting tweet about it.

Distribution of tweets by each fan and game status:

```
hashtag      game_status
#gohawks     after_game       7761
             during_game     23706
             pre_game       110900
#gopatriots  after_game      45213
             during_game     66098
             pre_game       297254
Name: title, dtype: int64
```

2.1. Finding Quarter Times:¶ To go one step further, I used the tweet information to find approximate quarter times.

For each quarter, I checked the tweets that has 'end of' and quarter name in the same tweet. For example for first quarter, I filtered out only tweets that has 'end of' and either '1st quarter' or 'first quarter' substrings.

The results are not too many, so I manually checked the meanings and assigned the earliest tweet that refers to the end of quarter. In this case, the results are driven from users and approximate, but the best mapping I could find.

```
Looking matches for first quarter|1st quarter ...
```

	title	citation_date_fixed
425143	0-0 at the end of the first quarter. #Seahawks...	2015-02-01 15:58:08-08:00
425148	#SB49 end of first quarter. #Seattle 0 #Patrio...	2015-02-01 15:58:09-08:00
425264	At the end of the first quarter the game is ti...	2015-02-01 15:58:26-08:00
425289	It's scoreless between the #Patriots & #Seahaw...	2015-02-01 15:58:29-08:00
425291	At the end of the 1st quarter, score is 0-0! #...	2015-02-01 15:58:29-08:00

	title	citation_date_fixed
425560	At the end of a speedy first quarter, #SuperBo...	2015-02-01 15:59:02-08:00
425561	At the end of a speedy first quarter, #SuperBo...	2015-02-01 15:59:02-08:00
425589	And thatâ€™s the end of the 1st quarter. 0-0. ...	2015-02-01 15:59:05-08:00
425611	Alright, whoever had 0-0 at the end of the 1st...	2015-02-01 15:59:08-08:00
425722	At the end off the first quarter yhe score is ...	2015-02-01 15:59:25-08:00
425867	At the end of the 1st quarter Seattle #Seahawk...	2015-02-01 15:59:52-08:00
426071	And that's the end of the 1st quarter. 0-0 #Se...	2015-02-01 16:00:28-08:00
426073	And that's the end of the 1st quarter. 0-0 #Se...	2015-02-01 16:00:28-08:00
426609	0 0 end of 1st quarter #SuperBowlXLIX who gonn...	2015-02-01 16:01:52-08:00
433033	I lasted until the end of the 1st quarter unti...	2015-02-01 16:13:03-08:00

```
Looking matches for second quarter|2nd quarter ...
```

	title	citation_date_fixed
444742	agreement btwn #Seahawks & #Patriots: make...	2015-02-01 16:35:29-08:00
461206	@DJAelita @TheFaithBreaker 14 - 14 at the end ...	2015-02-01 17:09:09-08:00
468326	I swear the #SB49 is paid to win.. We were pla...	2015-02-01 17:46:58-08:00
476059	#patriots have to show the same guts the #Seat...	2015-02-01 18:11:09-08:00
538569	Last play last nite was a bad call cuz it didn...	2015-02-02 07:33:55-08:00

```
Looking matches for third quarter|3rd quarter ...
```

	title	citation_date_fixed
476087	The #Seahawks lead the #Patriots, 24-14, at th...	2015-02-01 18:11:15-08:00
477642	At the end of the 3rd quarter, the #Seahawks h...	2015-02-01 18:18:36-08:00
477644	24-14 at the end of 3rd quarter. Seahawks lead...	2015-02-01 18:18:37-08:00

```
Looking matches for fourth quarter|4th quarter ...
```

	title	citation_date_fixed
475939	At the end of the 4th quarter of #SuperBowlXLI...	2015-02-01 18:10:43-08:00
483169	You HAVE to have a big lead on the #Patriots a...	2015-02-01 18:39:49-08:00
486112	Made it just for the end of the fourth quarter...	2015-02-01 18:48:30-08:00
489895	Coming from behind in end of 4th quarter is wh...	2015-02-01 18:53:28-08:00
532798	Seahawks last great play near the end of the 4...	2015-02-02 00:15:37-08:00
536105	So... Yes, stupid call at the end of the game....	2015-02-02 05:23:54-08:00

2.2. Finding Halftime Show Times:

I used a similar approach above and used halftime and various strings for Katty Perry who performed at the half time: 'Katty Perry|@kattyperry|katty'

```
Check Head 10 of the found tweeets:
```

	title	citation_date_fixed
459695	Game tied at the half! Who will win? #SuperBow...	2015-02-01 17:05:02-08:00
459764	Well it's halftime and the game is tied at 14 ...	2015-02-01 17:05:09-08:00
459771	so i catch the game shortly before halftime. ...	2015-02-01 17:05:10-08:00
459802	Shit it's halftime, what channel is the blowjo...	2015-02-01 17:05:16-08:00

	title	citation_date_fixed
459810	Super Bowl 49 #patriots #brady #halftime #icec...	2015-02-01 17:05:17-08:00
459811	#katyperry #halftime \n#patriots — watching Su...	2015-02-01 17:05:17-08:00
459869	Wow exciting superbowl 2 great teams 2 big QB,...	2015-02-01 17:05:26-08:00
459887	That's how you go to halftime. #GoHawks	2015-02-01 17:05:28-08:00
459893	What a game halftime great #SuperBowlXLIX #Pat...	2015-02-01 17:05:30-08:00
459898	Super Bowl halftime shows are a turd oasis wit...	2015-02-01 17:05:31-08:00

```
Check Tail 10 of the found tweeets:
```

	title	citation_date_fixed
462907	Bad ass halftime show! @katyperry @pepsi #SBXL...	2015-02-01 17:18:39-08:00
462912	@katyperry is just incredible, I'm in love #ha...	2015-02-01 17:18:40-08:00
462913	So far my favorite halftime show ever. #katype...	2015-02-01 17:18:40-08:00
462939	Loving the halftime #GoHawks	2015-02-01 17:18:50-08:00
462952	I'm quite enjoying KP halftime show\n\n#GoHawks	2015-02-01 17:18:55-08:00
462959	This halftime show is so damn good! Katy Perry...	2015-02-01 17:18:59-08:00
462995	You're gonna hear me ROAR katyperry #katyperry...	2015-02-01 17:19:20-08:00
463020	I'm so glad Katy Perry is doing halftime... Be...	2015-02-01 17:19:35-08:00
463034	@katyperry @LennyKravitz Awesome halftime show...	2015-02-01 17:19:46-08:00
463058	Missy made a comeback! #superbowl #halftime #p...	2015-02-01 17:19:55-08:00

I also explored other combinations for 'halftime' and 'started|start' and 'halftime and end|ended' tweet matches to further investigate. Also, from Google and youtube videos, I found that the halftime show lasted 12 minutes 35 seconds. I incorporated this information with some buffer time as well.

Again the times are rather approximate, but this will at least help us in approximately when in the game, these tweets are being posted and put the information conceptually in place. Final time intervals I decided are below:

```
first_qt_start : 2015-02-01 15:30:00-08:00
first_qt_end : 2015-02-01 15:58:00-08:00
second_qt_start : 2015-02-01 15:58:00-08:00
second_qt_end : 2015-02-01 17:09:00-08:00
half_time_start : 2015-02-01 17:09:00-08:00
half_time_end : 2015-02-01 17:27:00-08:00
third_qt_start : 2015-02-01 17:27:00-08:00
third_qt_end : 2015-02-01 18:18:00-08:00
fourth_qt_start : 2015-02-01 18:18:00-08:00
fourth_qt_end : 2015-02-01 19:06:00-08:00
```

Distribution of tweets by each fan and game detailed status:

```
hashtag       game_status_detailed
#gohawks      after_game                7755
              first-quarter             3726
              fourth-quarter            4642
              half-time show             900
              pre_game                110900
              second-quarter            8208
              third-quarter             6236
#gopatriots   after_game               45203
              first-quarter            13210
              fourth-quarter           15808
```

```
            half-time show          1921
            pre_game              297254
            second-quarter         27867
            third-quarter           7302
Name: title, dtype: int64
```

The final time mapping missing is to identifying the times where scores changed. I address this part on feature engineering section.

3. Finding Score Times:¶ I know the scores of the game, so with regex I searched the tweets where scores shared, and from those tweets I filtered and kept only the correct scores realized within the duration of the game (see scores_list).

scores_list = ['7-0','7-7','14-7','14-14','14-17','14-24','21-24','28-24']

This is especially important because there are some tweets with prediction scores of the game. Also, according to american footbal rules, once touchdown happens a team gets 6 points and then with extras that score can increase up to 1 or 2 more points. Some tweets were posted without the extra points (ie 14-23) so scores*list helped me filter out those as well since I want to be consistent with the Scoring Plays results I dounf in espn website (https://www.espn.com/nfl/playbyplay/*/gameId/400749027). I also limited the post date of the tweets after game start time to make sure to get real scores.

Then, my assumption is that if a user tweets the score, then it means there is a change in score and one of the team has scored. And I could use the relative real time of the score by taking the time of the earliest tweet posted with that score. In this way for each score, I can have a relative real tweet times in PST time where each score happened. Of course, the earliest tweet might be posted couple minutes later and might not be exact with the score time. But this is the best approximation and map I could get with the available data I have.

As a sanity check I also looked whether the found time frames fall in the relative place in the quarter the score happened. You can see the final dataframe with score and time mapping below:

	score	citation_date_fixed	leading	quarter
0	0-0	2015-02-01 15:30:00-08:00	Tie	1st Quarter
0	7-0	2015-02-01 16:12:00-08:00	Patriots	2nd Quarter
1	7-7	2015-02-01 16:33:59-08:00	Tie	2nd Quarter
2	14-7	2015-02-01 16:48:10-08:00	Patriots	2nd Quarter
3	14-14	2015-02-01 16:58:56-08:00	Tie	2nd Quarter
4	14-17	2015-02-01 17:39:21-08:00	Seahawks	3rd Quarter
5	14-24	2015-02-01 17:54:34-08:00	Seahawks	3rd Quarter
6	21-24	2015-02-01 18:28:17-08:00	Seahawks	4th Quarter
7	28-24	2015-02-01 18:30:34-08:00	Patriots	4th Quarter

This table shows approximately at what time the score happened according to the tweets, in which quarter and who is leading the game after the score. I will use this data frame to table my data for the prediction task. Real scoring plays relative to game time can be found here: https://www.espn.com/nfl/playbyplay/_/gameId/400749027

Character-centric time-series tracking¶

Can we track the average perceived emotion across tweets about each player in the game across time in each fan base? Can we correlate these emotions with the game score in the given time interval?

In this part, I used the cleaned title column, with 4 different emotion features. The analysis is only done the tweeets that are tweeted during the game.

To find time based perceived emotion for each player on each fan base, I aggregated data over hashtag, player and time interval and take the mean of the selected emotion feature score. I used hashtag for each fan base, by making the assumption that if user post a tweet using #gopatriots, #patriots and #gohawks hashtag they are supporting corresponding team. I also included #patriots mainly because the tweets for #gopatriots were very low. The ratio of #gohawks (170k) to #gopatriots (24K) was 92-8%. Adding #patriots tag increased the total tweets to 550K and changed the balance to 75% patriots to 25% hawks. Using #patriots hashtag and making the assumption that only supporters/fan uses #patriots hashtag is less likely to be true, however, this change helped me find more players in more timeframes and reduced the data sparsity substancially.

For time intervals, I used different ranges 1,2,5,10 minute intervals. This is more related on the accuracy and granularity we are interested in. According the play by play and scoring play tables, put/interception and touchdowns happin 2-5 minute intervals. Aggregating the data too much might dillute the peaks in the important moments. For other plots, we are more interested in general pattern to compare it for different features/playeers etc, to make plots more consumable, I increased the time interval used on those plots. Time interval used either mentioned on x axis or title of the plots/subplots.

¶

Sample Table showing how rounded 1-minute time interval looks like for 5 samples:

	citation_date_fixed	post_time_1_intervals
0	2015-01-14 00:04:41-08:00	2015-01-14 00:04:00-08:00
1	2015-01-14 00:05:50-08:00	2015-01-14 00:05:00-08:00
2	2015-01-14 00:07:18-08:00	2015-01-14 00:07:00-08:00
3	2015-01-14 00:09:58-08:00	2015-01-14 00:09:00-08:00
4	2015-01-14 00:12:20-08:00	2015-01-14 00:12:00-08:00

Image shows the scoring summary of the game shown in ESPN website:

Scoring Summary

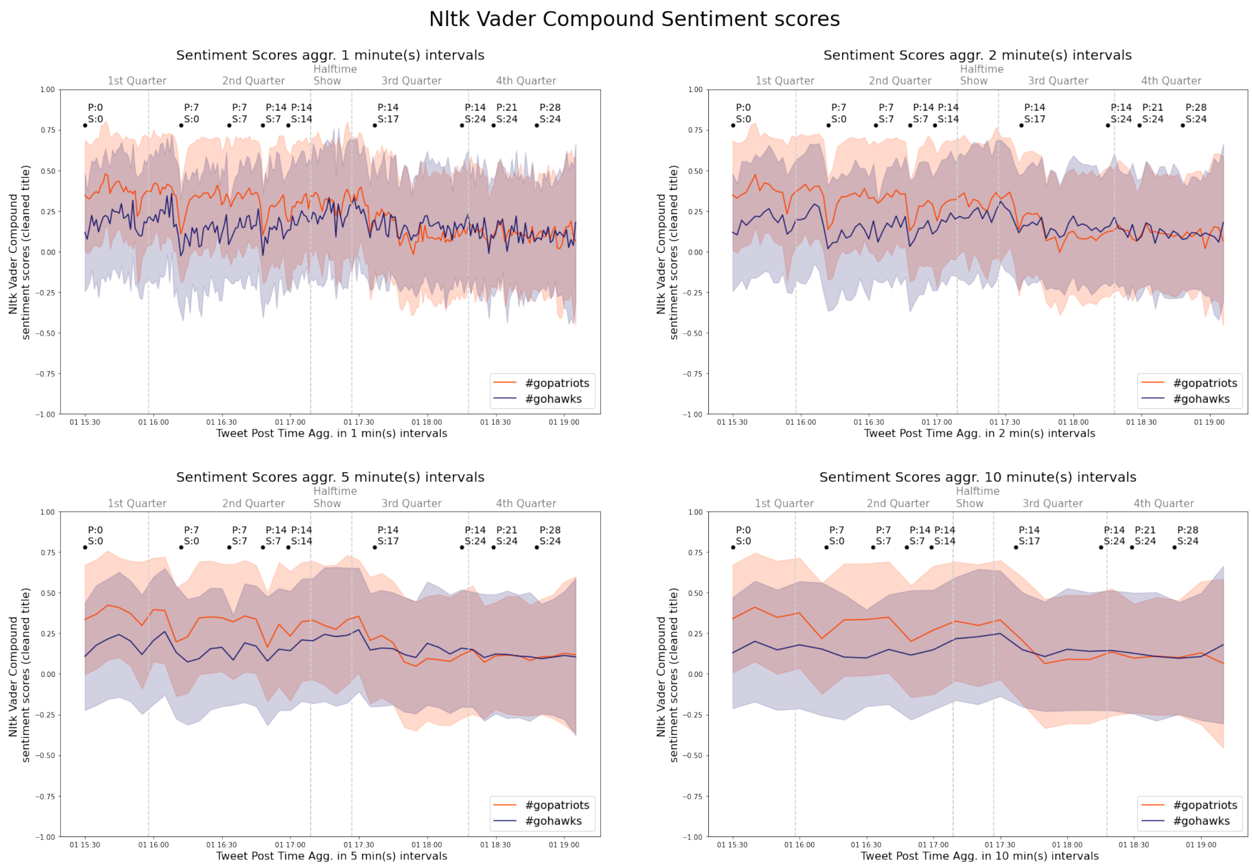
SECOND QUARTER			NE	SEA
TD	9:47	Brandon LaFell 11 Yd pass from Tom Brady (Stephen Gostkowski Kick) *9 plays, 65 yards, 4:10*	7	0
TD	2:16	Marshawn Lynch 3 Yd Run (Steven Hauschka Kick) *8 plays, 70 yards, 4:51*	7	7
TD	0:31	Rob Gronkowski 22 Yd pass from Tom Brady (Stephen Gostkowski Kick) *8 plays, 80 yards, 1:45*	14	7
TD	0:02	Chris Matthews 11 Yd pass from Russell Wilson (Steven Hauschka Kick) *5 plays, 80 yards, 0:29*	14	14
THIRD QUARTER			**NE**	**SEA**
FG	11:09	Steven Hauschka 27 Yd Field Goal *7 plays, 72 yards, 3:51*	14	17
TD	4:54	Doug Baldwin 3 Yd pass from Russell Wilson (Steven Hauschka Kick) *6 plays, 50 yards, 3:13*	14	24
FOURTH QUARTER			**NE**	**SEA**
TD	7:55	Danny Amendola 4 Yd pass from Tom Brady (Stephen Gostkowski Kick) *9 plays, 68 yards, 4:15*	21	24
TD	2:02	Julian Edelman 3 Yd pass from Tom Brady (Stephen Gostkowski Kick) *10 plays, 64 yards, 4:50*	28	24

ref: https://www.espn.com/nfl/game/_/gameId/400749027

Fans' emotion overtime aggregated on different time intervals¶

I start with comparing different time intervals. Below plots show emotion of each fan group over 1,2,5 and 10 minute(s) intervals. X-axis shows the time during game (Feb 1st 15:30 to 19:00), y-axis shows the nltk.vader compound score applied on cleaned title. The aggregation is made taking mean of the compound score for each hashtag within given time interval. Dashed gray lines shows the approximated quarter and halftime show end/start times. The dots with score text shows the game score during that point in time. First score belongs to Patriots and second belongs to Seahawks. 7-0 means Patriots 7, Seahawks 0. Transparent band colors shows the standard deviation of the aggregated data. I used nltk vader sentiment features:

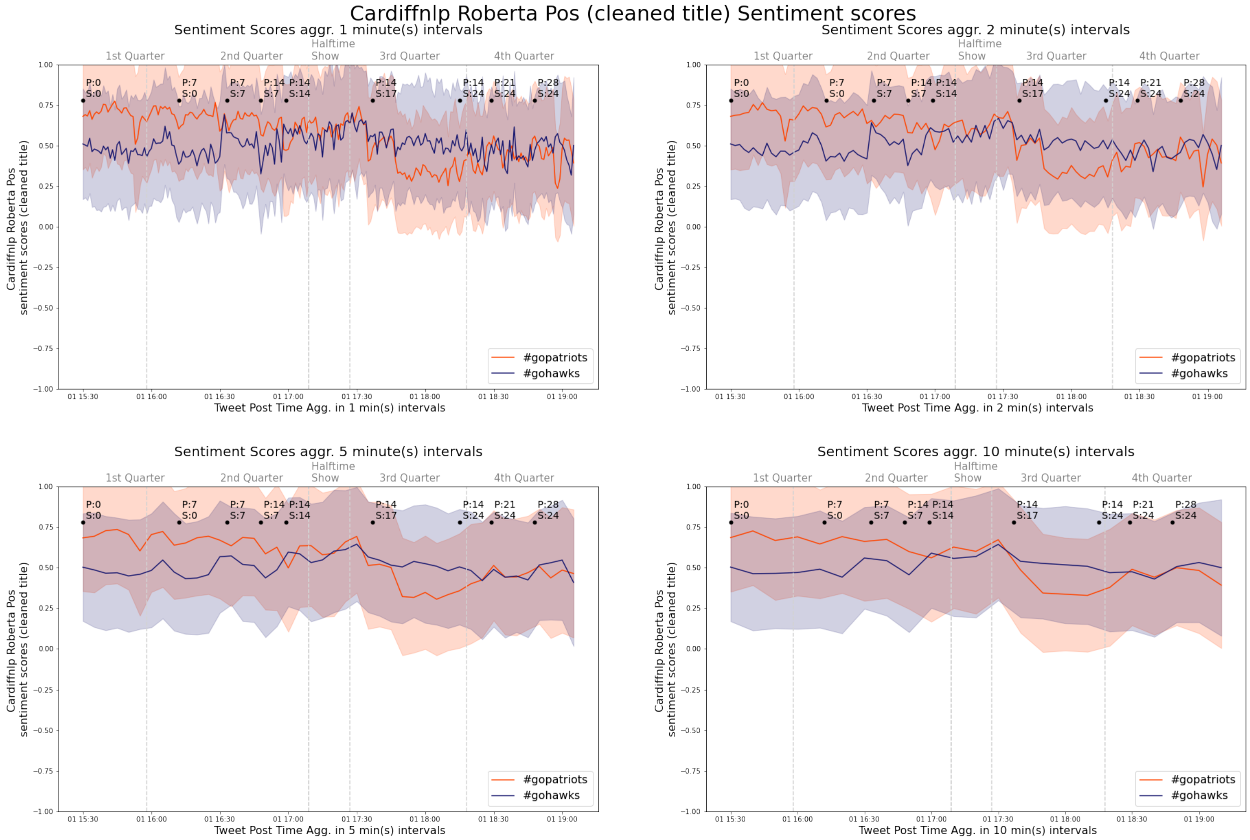
NLTK Vader Fans' emotion during game¶ Plot:



Observations:

- We can see the spikes and downs in the teams when scores happened, for example when Patriots scored a touch down and game score is 7-0, the orange line peaks, showing increase in positive emotion, we can see the huge decrease on Patriots emotions when hawks makes score 14-17, hawks emotions get more positive than patriots when the score is 14-24.
- With nltk vader features, in 2 and 5 minute intervals plot we see same emotion patteerns for patriots and hawks from beginning until midlle of the 3rd quarter in the game. PAtriots have more positive emotions compared to seahawks, though, after the middle of 3rd quarter, at first hawks emotion gets higher and then gets almost similar with Patriots one.

CardiffNLP Fans' emotion during game¶ Same plot drawn for CardiffNLP sentiment features:

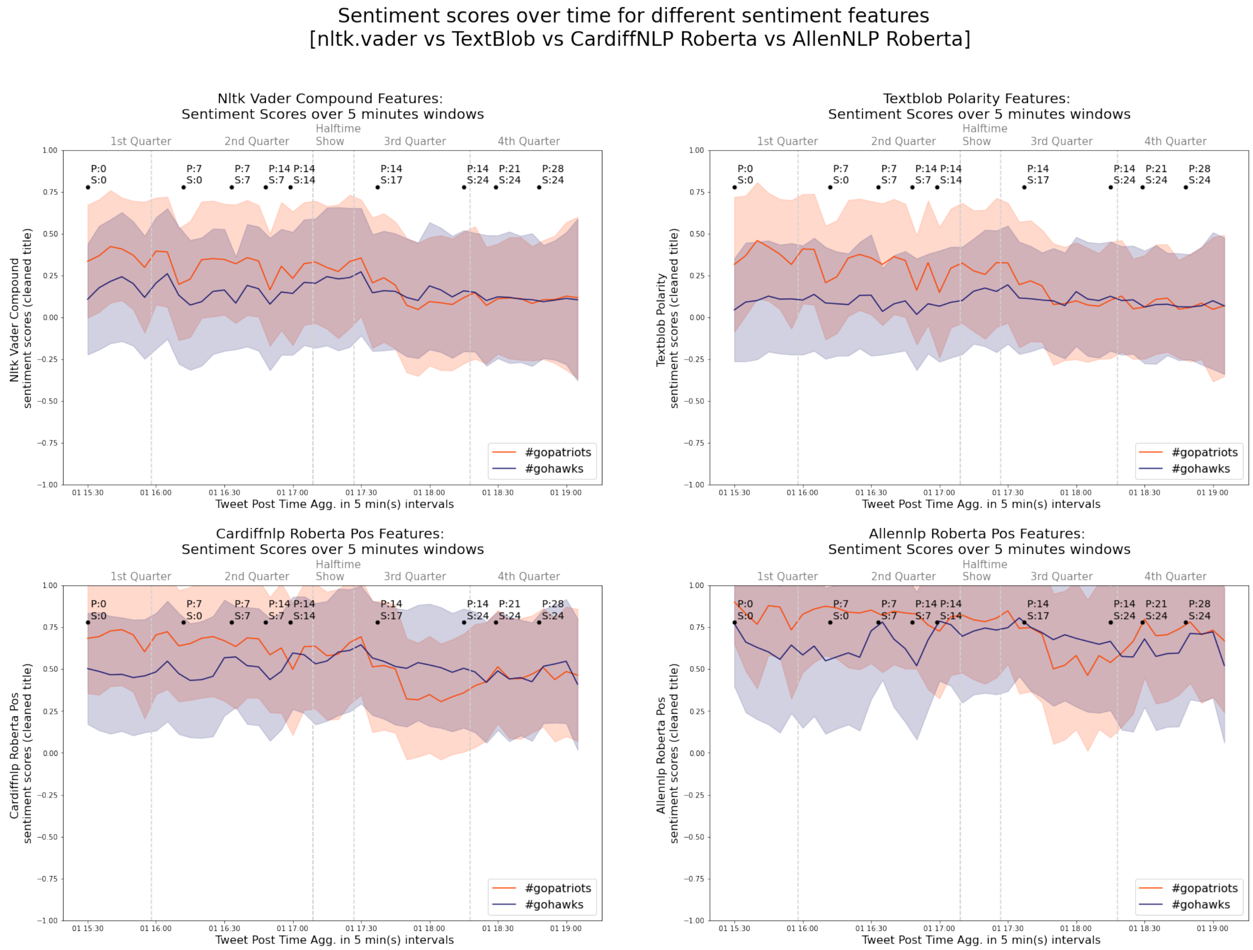


Observations:

- Positive scores with CardiffNLP seems to be much higher compared to NLTK in overall.
- The trends are more clear as well. By looking at the lines we can mostly tell which team is winning. We can see that orange line, patriots, overall emotions is more positive until the end of 2nd quarter, the periods where Patriots are leading the game. Whenever the score difference decrease or a tie happens between the teams the curve of each hashtag becomes closer to each other and the gap decreases as well. When Seahawks, starts to lead in the 3rd quarter, we can also see that the line for seahawks surpasses the patriots as well.
- We can also see big spikes and down emotion trends also correlated with scores.
- As a first intuition, Cardiff NLP emotion distribution seems to show the correlation between scores and perceived emotions on teams more clearly than NLTK. The gaps between lines are more telling especially in the 3rd quarter in CardiffNLP.

Fans' Emotion towards each team: Comparison of different sentiment feature types¶

In this section, I compare the 4 extracted sentiment features in a fixed interval time of 5 minute windows during thee game.



Observations:

- For NLTK in the first half of game the trends for each team is very close, on the other hand, for Textblob the closeness is less and for the 2 Roberta features set the overall line shapes are different.
- For NLTK and Textblob, rule-based approaches, we can see that Patriots have more positive emotions, in the first half of the game, then in the second half the emotions are more flat and lower for both teams, the gap is also very small. For CardiffNLP and AllenNLP Roberta models, we can see better trends.
- If we assume higher positive emotions corresponds to that team is leading the game, then we can see the game status clearly with line trends through out the game with AllenNLP:
  - P:7-S:0, orange line higher -> patriots winning
  - P:7-S:7, orange and purple line gap decrease -> tie
  - P:14-S:7, blue line curve decrease -> patriots score, tie break, patriots leading again
  - P:14-S:14, seahawks scores, gap decrease -> tie
  - P:14-S:24, purple line surpasses orange -> seahawks winning
  - P:21-S:24, orange line surpasses purple -> patriots winning
  - P:28-S:24, orange line surpasses purple -> patriots winning
- The gap between the lines, can even tell the score difference. Notice that when the score difference is higher the gap is bigger, whereas when there is a tie the two lines either merged or get very close.

Fans’ Emotion for Top 8 tweeted players¶

Who are the most tweeted players?

	#gopatriots	#gohawks	Tom Brady	Russell Wilson	Rob Gronkowski
tweet_counts	357785.000000	115377.000000	32383.000000	8710.00000	5395.000000
tweet_percentage	0.649418	0.209421	0.058779	0.01581	0.009792

	Marshawn Lynch	Richard Sherman	Chris Matthews	Julian Edelman	Earl Thomas
tweet_counts	4845.000000	4314.00000	2753.000000	2439.000000	2130.000000
tweet_percentage	0.008794	0.00783	0.004997	0.004427	0.003866

	LeGarrette Blount	Doug Baldwin	Darrelle Revis	Kam Chancellor	Vince Wilfork
tweet_counts	1948.000000	1541.000000	1342.000000	1010.000000	949.000000
tweet_percentage	0.003536	0.002797	0.002436	0.001833	0.001723

Who are the top players for each hashtag? Are the players play in defense or offense? Are the 10 players of each hashtag is for that team or the rival team?

Below table shows top 10 players for each hashtag:

#gopatriots

	0	1	2	3	4
hashtag	#gopatriots	#gopatriots	#gopatriots	#gopatriots	#gopatriots
player2	Tom Brady	Rob Gronkowski	Julian Edelman	LeGarrette Blount	Richard Sherman
Team	Patriots	Patriots	Patriots	Patriots	Seahawks
Category	Offense	Offense	Offense	Offense	Defense
tweeet_count	28360	4327	2401	1907	1457

	5	6	7	8	9
hashtag	#gopatriots	#gopatriots	#gopatriots	#gopatriots	#gopatriots
player2	Marshawn Lynch	Russell Wilson	Darrelle Revis	Chris Matthews	Earl Thomas
Team	Seahawks	Seahawks	Patriots	Seahawks	Seahawks
Category	Offense	Offense	Defense	Offense	Defense
tweeet_count	1306	1246	1245	1156	1037

#gohawks

	39	40	41	42	43
hashtag	#gohawks	#gohawks	#gohawks	#gohawks	#gohawks
player2	Russell Wilson	Tom Brady	Marshawn Lynch	Richard Sherman	Chris Matthews
Team	Seahawks	Patriots	Seahawks	Seahawks	Seahawks
Category	Offense	Offense	Offense	Defense	Offense
tweeet_count	7464	4023	3539	2857	1597

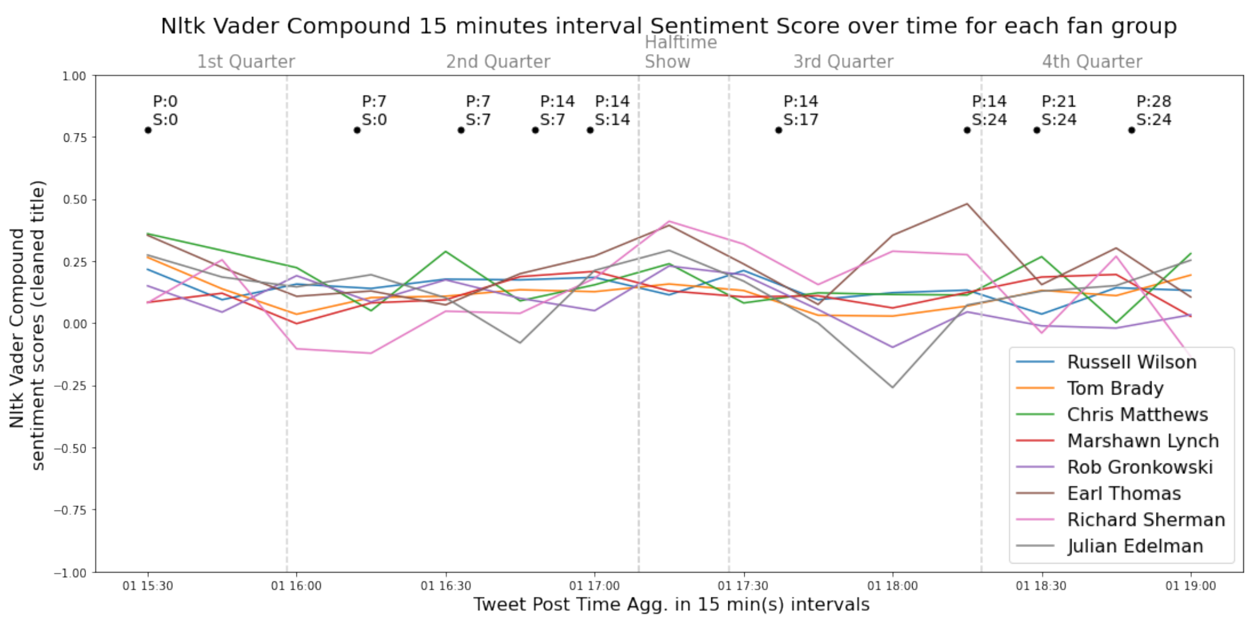
	44	45	46	47	48
hashtag	#gohawks	#gohawks	#gohawks	#gohawks	#gohawks
player2	Doug Baldwin	Earl Thomas	Rob Gronkowski	Kam Chancellor	Jermaine Kearse
Team	Seahawks	Seahawks	Patriots	Seahawks	Seahawks
Category	Offense	Defense	Offense	Defense	Offense
tweeet_count	1307	1093	1068	843	518

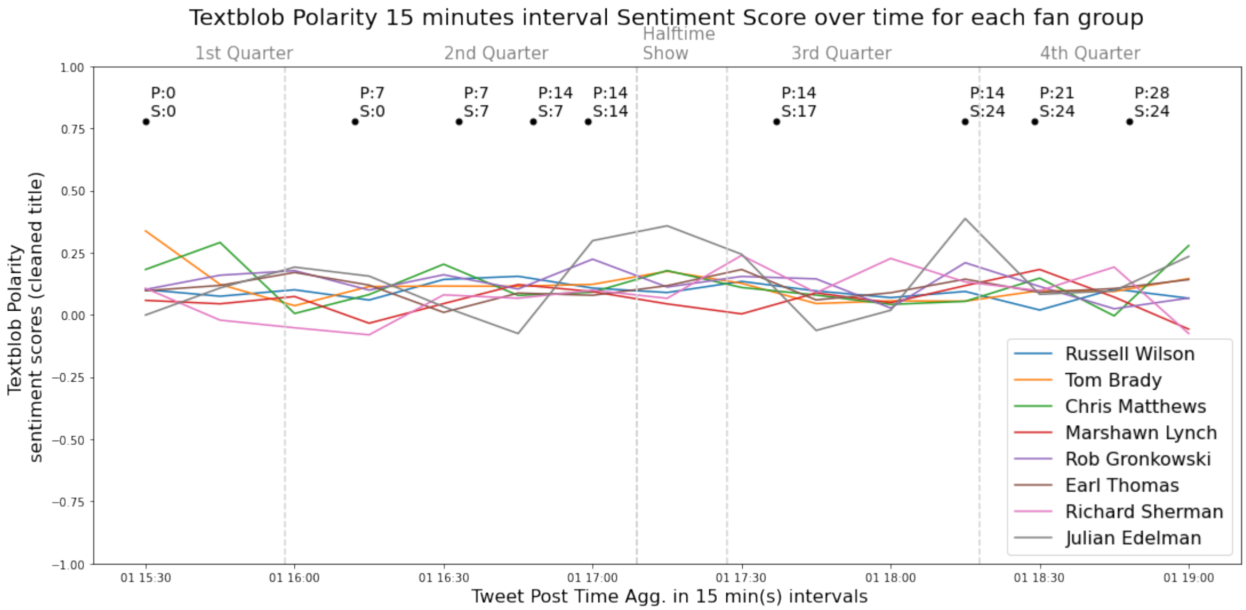
Tom Brady gets the bigger number of tweets. We can see that hawks supporters tends to talk more about Seahawks players whilst patriots fan talks for players of both teams 5/10 top players mentioned by patriots are SeaHawks players. One might suggest that this might be because for patriots we merged #gopatriots and #patriots tags, however, this was the case with 5/10 ratio before merging #gopatriots and #patriots tag as well.

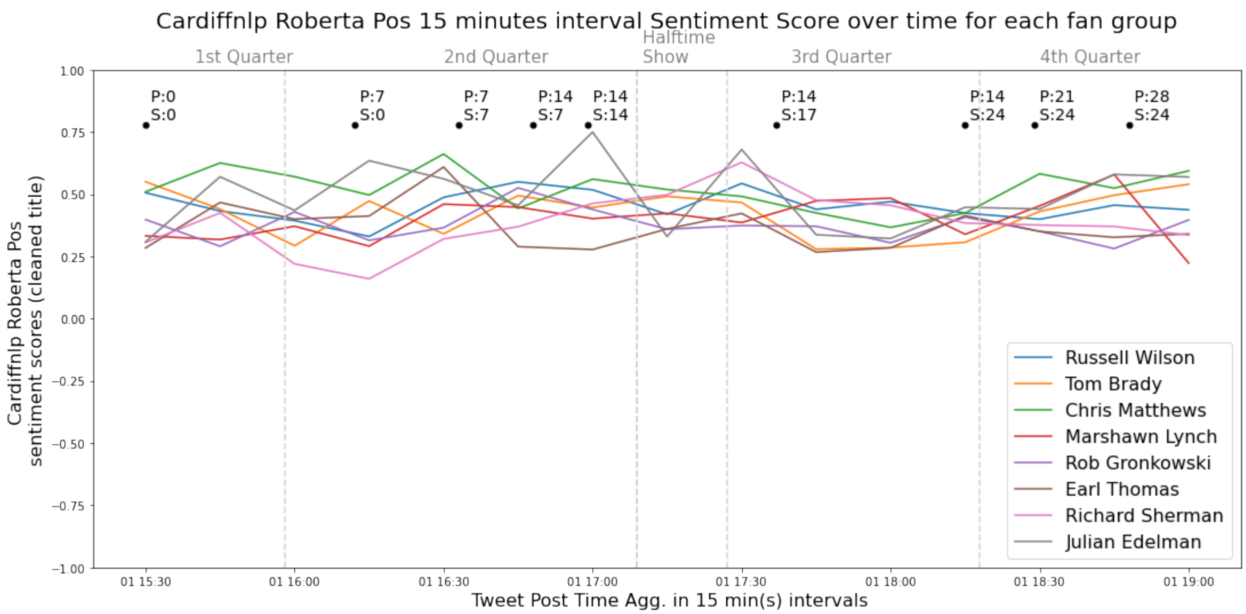
In both of the hashtags majority of the comments are for offense players than defense players. #gopatriots mainly talks about opposite team's defense players when the player's category is defense. This potentially means that even talking about defense players, patriots fan are more focusing on the offense moments of their team. We may conclude that fans are more interested talking about attack moments of their team (deffense of opposing team).

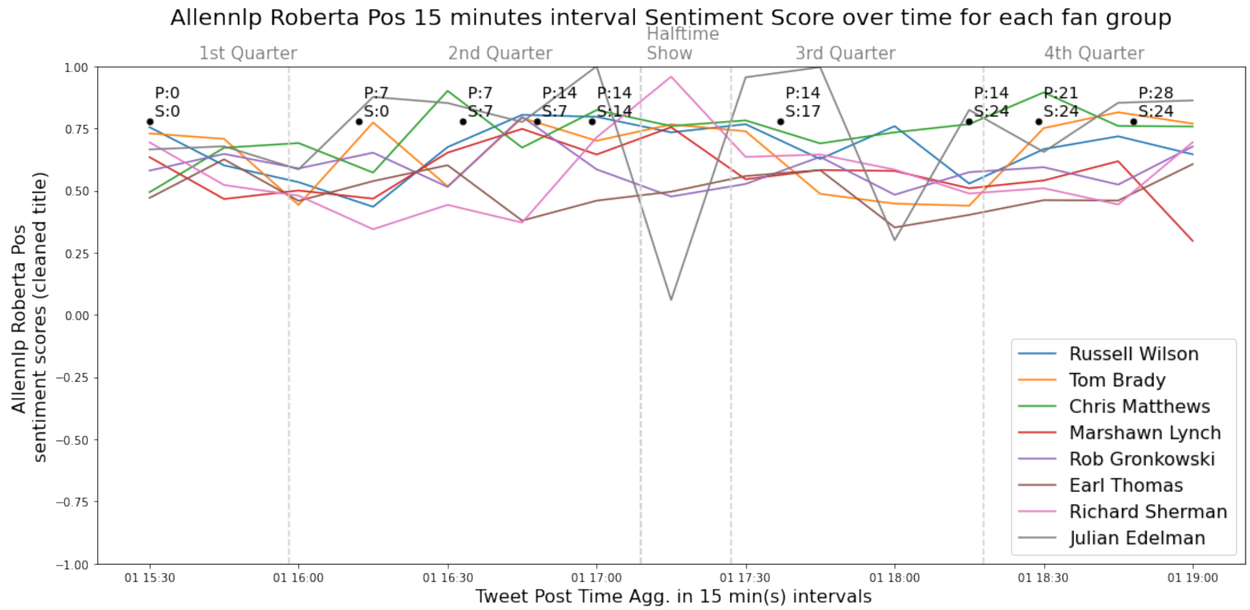
I plotted the perceived emotion about players over 15 minutes time inteerval for each feature set. I chose the most tweeted 8 players and increaseed time interval to 15 minutes for convenience of reading plots more easily. The perceived emotions in these plots are combined for each fan group. Given that the data is unbalanced against each hashtag, the results might be towards more Patriots' fan feeling.

```
Tom Brady, Russell Wilson, Rob Gronkowski,
Marshawn Lynch, Richard Sherman, Chris Matthews,
Julian Edelman, Earl Thomas
```




Textblob Polarity 15 minutes interval Sentiment Score over time for each fan group
1st Quarter
2nd Quarter
Halftime Show
3rd Quarter
4th Quarter
P:0 S:0
P:7 S:0
P:7 S:7
P:14 S:7
P:14 S:14
P:14 S:17
P:14 S:24
P:21 S:24
P:28 S:24
Textblob Polarity sentiment scores (cleaned title)
Tweet Post Time Agg. in 15 min(s) intervals
Russell Wilson
Tom Brady
Chris Matthews
Marshawn Lynch
Rob Gronkowski
Earl Thomas
Richard Sherman
Julian Edelman


Cardiffnlp Roberta Pos 15 minutes interval Sentiment Score over time for each fan group
1st Quarter
2nd Quarter
Halftime Show
3rd Quarter
4th Quarter
P:0 S:0
P:7 S:0
P:7 S:7
P:14 S:7
P:14 S:14
P:14 S:17
P:14 S:24
P:21 S:24
P:28 S:24
Cardiffnlp Roberta Pos sentiment scores (cleaned title)
Tweet Post Time Agg. in 15 min(s) intervals
Russell Wilson
Tom Brady
Chris Matthews
Marshawn Lynch
Rob Gronkowski
Earl Thomas
Richard Sherman
Julian Edelman



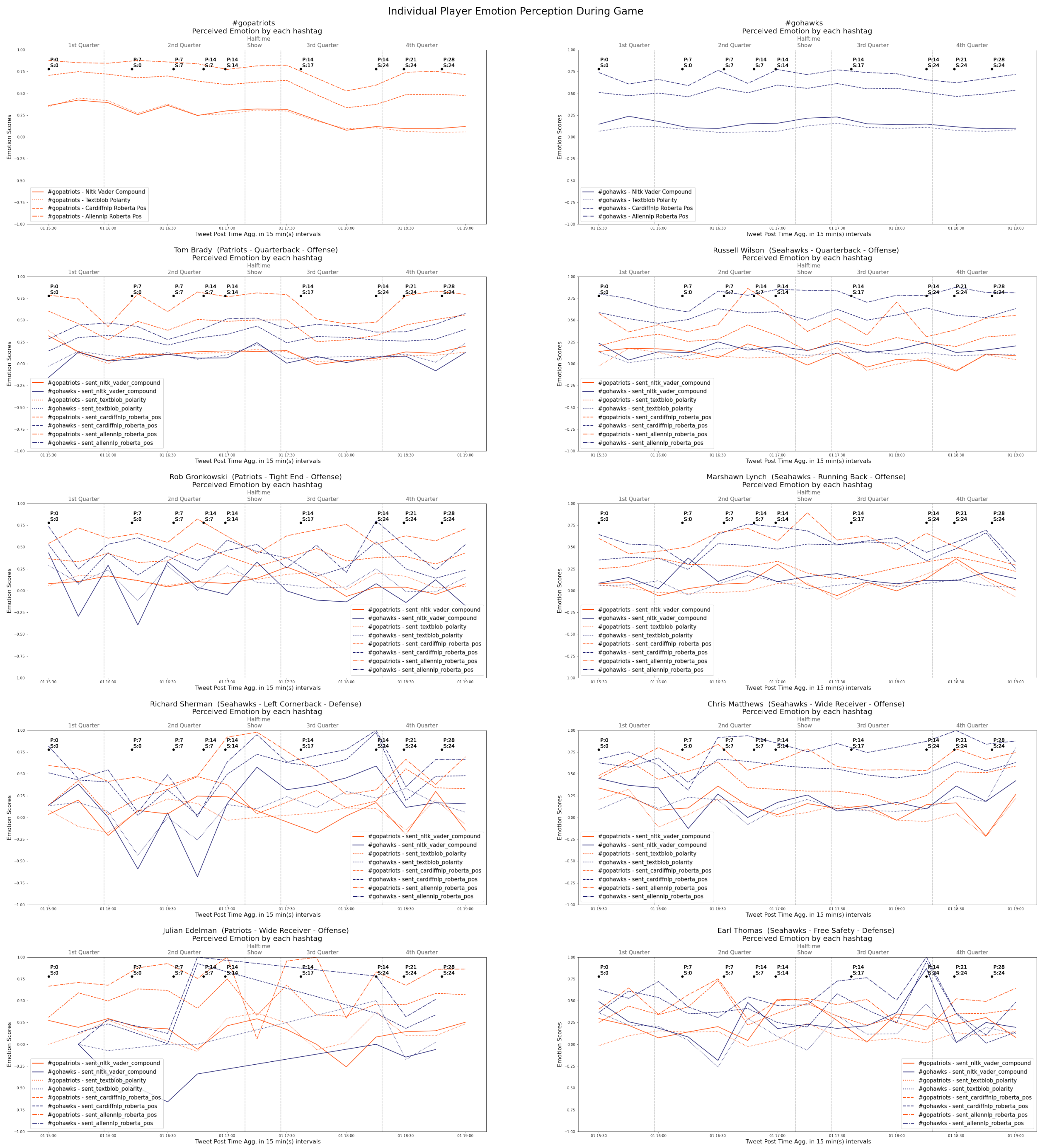
Observations:

- Notice that 5 of the 8 top tweeted players are from Seahawks. (Russell Wilson, Chris Matthews, Marshawn Lynch, Earl Thomas, Richard Sherman)
- First two players are passing leaders of each team, the next 2 are receiver for Seahawks, Rob is receiver for Patriots, Earl and Richard are defense players for Seahawks, and Julian is receiver for Patriots.
- We can see that emotions for players, varies a lot during the game, the most sharp and bigger changes seeem to happen in AllenNLP and NLTK. Julian Edelman seems to have the biggest change of heart from fans during the game, where it has the biggest peaks and downs.The emotions perceived for Tom Brady seems rather stale in the first 3 approaches, but changes a lot in AllenNLP.
- For CardiffNLP, we can see clear peaks on the players who involved plays during Touch Down and Field Goal times. This also makes sense since it is very likely that team's praise the player who take role in scoring or shades the players who fail to defense.

Individual Players' Emotion Plots over time during the game¶

In this part, I plotted individual players' perceived emotion during game. Each plot has emotion trends for 4 sentiment feature types and two of the fan base. Again plotted for 15 minute intervals for convenience.

```
Players selected: #gopatriots, #gohawks, Tom Brady, Russell Wilson,
Rob Gronkowski, Marshawn Lynch, Richard Sherman, Chris Matthews,
Julian Edelman, Earl Thomas
```



Observations:

- #gohawks, #gopatriots: these emotion groups belong to the tweets where I can't identiy a player. It also corresponds to the majority of the tweets. #gopatriots is mostly on highly positve stable curve up to 3rd quarter, and has a huge decrease in 3rd quarter. These makes perfect sense since Patriots lead thee game in first half and Seahawks, lead the 3rd quarter. I cannot clearly see trends in Touch Downs though. For seahawks, we can see smooth little peaks when Seahawks ties the game twice in thee second quarteer. Textblob and NLTK curves are almost same. The trends of Roberta models are very similar. AllenNLP assigns higher emotion scores than CardiffNLP.
- Tom Brady: The emotions are keep changing for Tom Brady during the game for both fans. Patriots

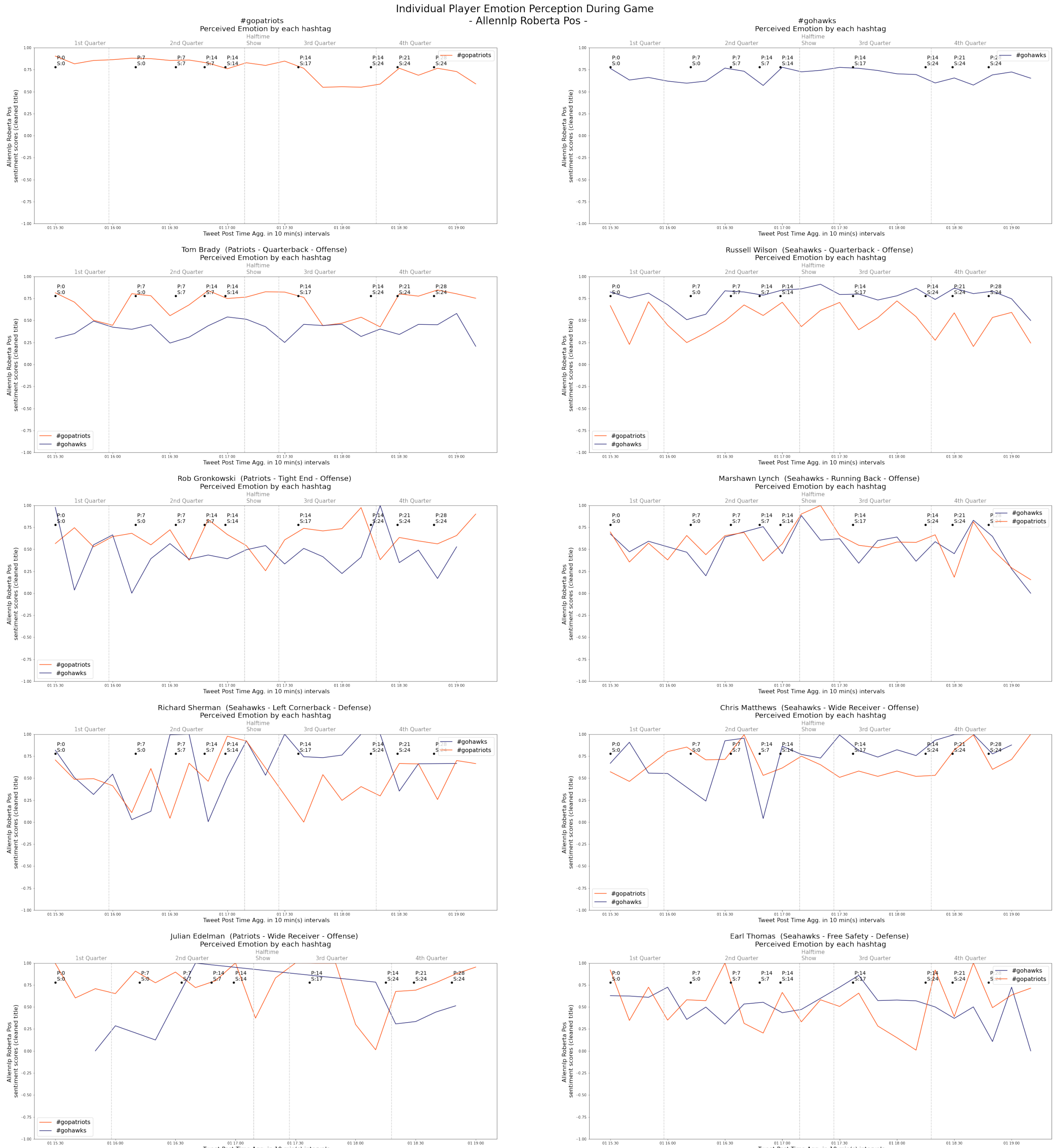
fan's positive emotion increases towards Tom Brady when Patriots score, and decrease considerably when Seahawks scores. Again we see that AllenNLP assigns higher values, then CardiffNLP and then the rule based features.

- Russell Wilson: The zigzag trend is more towards Russell Wilson from both fans, compared to Tom Brady.
- The trends of different feature types throughout the game per fan base seems to be similar but differs in the score amount.
- The more lower towards the subplots, players with lower tweet number, the data becomes more sparse throughout time or fan base, and the more chaotic lines seem to get. Patterns are harder to interpret, and for different feature set, the trends are no longer simiilar either.

Individual Players' Emotion Plots over time during the game for AllenNLP¶

Since it is harder to read the previous plots, I also plotted the same thing on only AllenNLP Roberta Feeatures. I chose AllenNLP since the polarity seem to be highhtened in this feature set and previous results were more interesting for this one than the other feature sets. I decreased the intervals to 10 minutes, since we have less lines, and want to see more granular, accurate patterns.

```
Players selected: #gopatriots, #gohawks, Tom Brady, Russell Wilson,
Rob Gronkowski, Marshawn Lynch, Richard Sherman, Chris Matthews,
Julian Edelman, Earl Thomas
```



Observations:

- Tom Brady: Patriots have a clear pattern for Tom, when Patriots score huge jump in emotion towards positive, when Seahawks scores huge decrease. This makes sense since Tom Brady is the main player. Patriots overall emotion is more positive for Tom.
- Russell Wilson: Again, as stated in the previous plot, the emotion is way zigzagy and fragile for Russell Wilson than Tom Brady. Seahawks fans emotion is more stable, and has an upward trend when SeaHawks scores. Seahawks emotions are more positive overall for Russell.
- Rob Gronkowski: He is an offense player for Patriots and the most interesting thing is that sometimes hawks fans' emotions are higher to a Patriot player than from patriot fans (halftime show, beegining

of 4th quarter). Could positive emotions be more sarcasm? Especially in 4th quarter the positive emotions come after hawks score. It is also possible that more than 1 player is listed in the tweet and we see these results, ie. hawks fan may be talking about a specific moment between the players of the teams such as Rob and his counterpart in the game.

- Marshawn Lynch: Perceived emotion by each fan group is again more complicated, we can't tell which fan group as an overall higher/lower emotion for him during the game. Two clear peaks are seahawks scored at the end of second quarter and when Patriots scored 21-24 in 4th quarter. He seems to make a comeback in the second half of the game, since hawks fan emotion is way higher in the second half of the game for him.
- Richard Sherman & Chris Matthews: I see huge peaks in emotion when Seahawks scores, which makes sense. Looks like more seahawks emotion towards the whol game rather than player itself.
- Julian Edelman: Very few hawks fan tweets. There is a huge PAtriots' fan emotion decrease twice during game, both which seahawks scored. He might have been interrupted or played a role on loosing the ball? (excuse my lousy american football terms).

For most of the players, I believe there might be more than one player stated in the tweet, and the tweet emotion might not directly correlate to the player itself but rather a significant position/event happening in the game where name of the player is stated. Such as "Wow, such a comeback Tom Brady and Russell Wilson are playing big. Way to go Seahawks I hope you win.": if this would have been a real tweet we would assign positive scores both Tom and Russell no matter which team's fan I am. The problem is the emotion in the tweet is not directly towards either player but rather team itself. There might be more interesting inferences from these plot about players but my american football knowledge is limited to the things I learned in the last two weeks.

Pairwise Players' Perceived Emotion Change During Game¶

In this part, I coupled the leaders of the game by their counterparts in the match, such as Tom Brady and Russell Wilson and check the overall perceived emotion for both players oveer time during game. Note that at this part perceived emotion is calculated for both fans together. Time interval over 5 minutes. I used AllenNLP Roberta feature for this part.

Patriots

Top 10 players tweet counts in all dataset:

	20	21	22	23	24
Team	Patriots	Patriots	Patriots	Patriots	Patriots
player2	Tom Brady	Rob Gronkowski	Julian Edelman	LeGarrette Blount	Darrelle Revis
Category	Offense	Offense	Offense	Offense	Defense
tweeet_count	32383	5395	2439	1948	1342

	25	26	27	28	29
Team	Patriots	Patriots	Patriots	Patriots	Patriots
player2	Vince Wilfork	Brandon LaFell	Michael Hoomanawanui	Chandler Jones	Danny Amendola
Category	Defense	Offense	Offense	Defense	Offense
tweeet_count	949	868	737	507	491

Saahawks

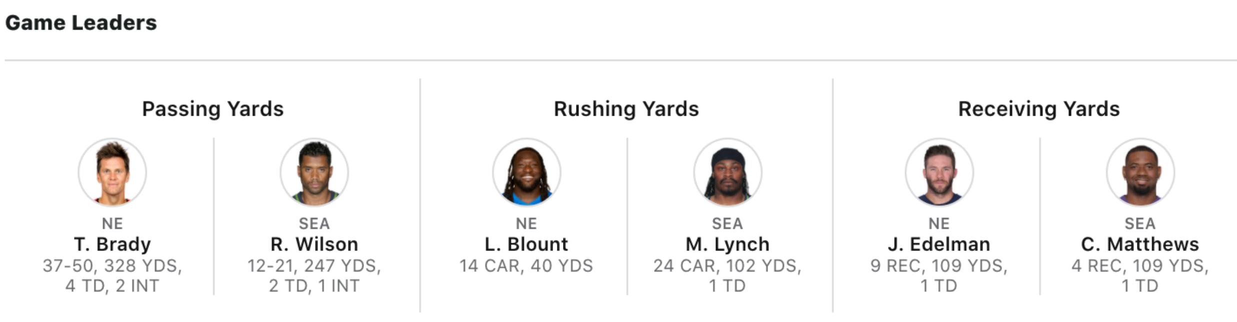
Top 10 players tweet counts in all dataset:

	0	1	2	3	4
Team	Seahawks	Seahawks	Seahawks	Seahawks	Seahawks
player2	Russell Wilson	Marshawn Lynch	Richard Sherman	Chris Matthews	Earl Thomas
Category	Offense	Offense	Defense	Offense	Defense
tweeet_count	8710	4845	4314	2753	2130

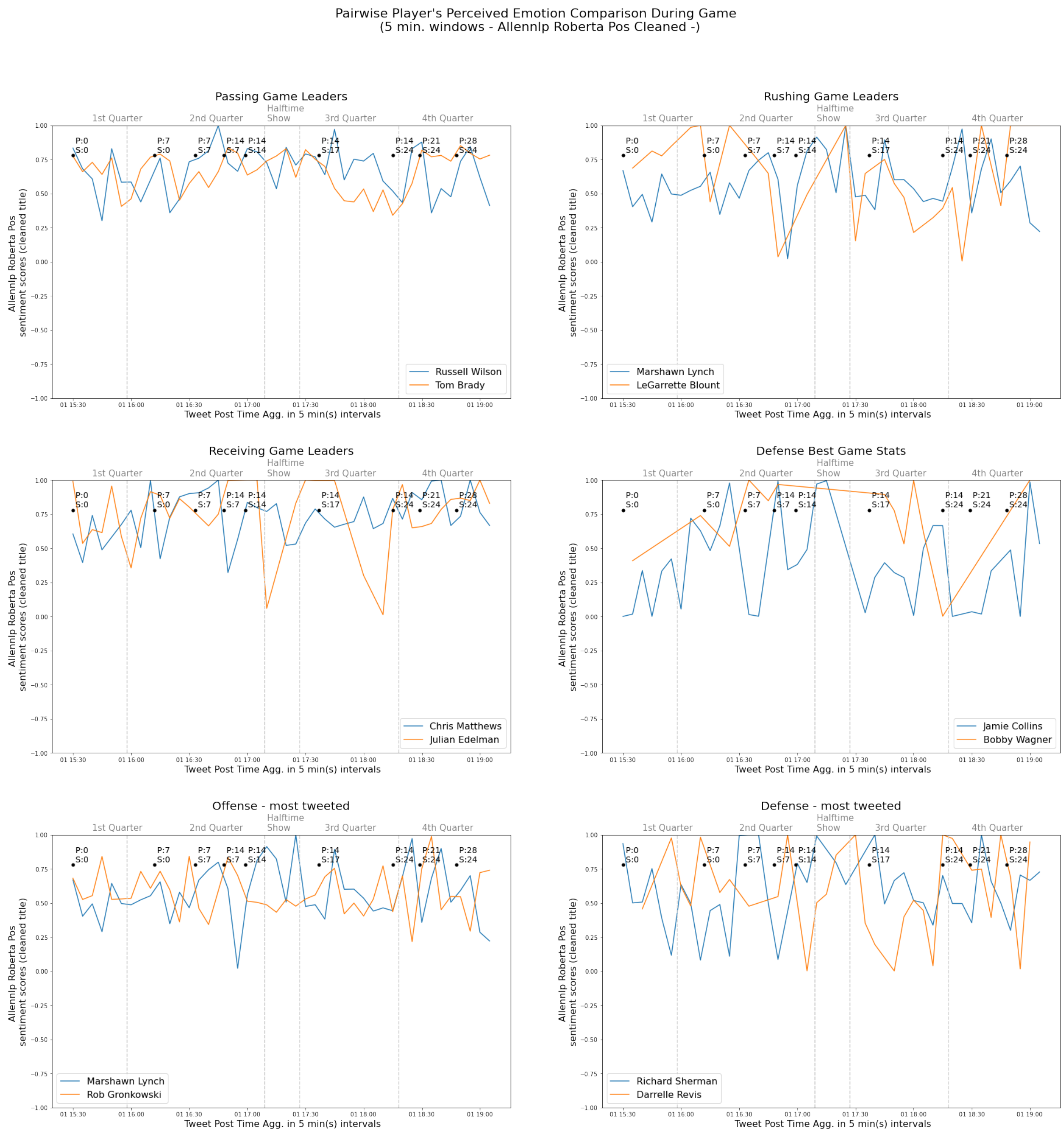
	5	6	7	8	9
Team	Seahawks	Seahawks	Seahawks	Seahawks	Seahawks
player2	Doug Baldwin	Kam Chancellor	Jermaine Kearse	Bryan Walters	Tony McDaniel
Category	Offense	Defense	Offense	Offense	Defense
tweeet_count	1541	1010	646	585	430

For offense and defense use players other than Tom Brady and Russell Wilson since we already compared them in Passing Leaders.

Image shows the game leaders listed in ESPN website for the game:



ref: https://www.espn.com/nfl/game/_/gameId/400749027



Passing Game Leaders: Tom Brady vs Russell Wilson

- There is a pick in both players emotion positively whenever a team is scored. Sometimes the emotion with each player is related based on their team's status only, ie. peak in Russell 3rd quarter when Seahawks scores, peak in Tom's curve when Patriots score in 2nd quarter P14-S7. Though sometimes emotions are rather merged, see peaks in P7-S0, P21-S24, P28-S24, both player's get peak, irrespective of which team is scored. I believe the reason behind this is that both players are mentioned in the time of these touch down tweets.

Rushing Game Leaders: Marshawn Lynch vs LeGarette Blount

- The curve patterns mostly have same high peaks and low drop moments for both players.

Receiving Game Leaders: Chris Matthews vs Julian Edelman

- The emotions are motly opposite for each player when touch downs happened. When Patriots score, Julian's emotion raise towards more positive and Chris' drops, and when Seahawks scores reverse happens. This is interesting, we can see the expected opposite team emotions for these player throughout the game.

Defense Best Game Stats: Jamie Collins vs Bobby Wagner

- Bobby Wagner's emotion features are more sparse than Jamie Collins. Again wee see the opposite feeling pattern we observed above with these two players.

Offense - most tweeted: Marshawn Lynch vs Rob Gronkowski & Defense - most tweeted: Richard Sherman - Darnell Revis

Again we see the opposite patterns, when one player peaks other's emotion drops in the touch down times and most likely in other significant events during game.

Prediction Task¶

From tweet emotions expressed to players in 1-minute time windows can we predict who is winning?

I aggregated the emotions about each player in 1-minute time windows during the game, for aggregation I used mean of the given sentiment scores. For mapping tweets to players I used the same starategy described in previous parts. I also added 2 more columns: #gohawks and #gopatriots where I get the overall sentiment for each team in 1-minute intervals. There is two main reason I added these two columns: 1- around 75% of the data I couldn't idendity player name, so if I don't I would be loosing too much information that might help my prediction task. 2- The final matrix created is a sparse matrix, not many players have tweets posted with their name each minute of the game. These two columns are dense one, that might increase the prediction power of the task.

The classes of this task is Tie, Patriots or Seahawks. This was also unable in the raw tweet data and for the labels I used the 'leading' column of the score_df I created in the previous parts. To map the game time to the tweets real time, I used the approximated score times (described in detail Data Exploration & Feature Engineering: Mapping Game Time to Real Time section), with the use of these time mappings I assigned the labels of who is leading the game to every 1-minute interval.

For each 4 feature set type, I created different datasets for the prediction task and applied 3 ML models to all 4 of them.

Feature sets:

- Nltk.sentiment.vader,
- TextBlob,
- CardiffNLP Roberta and
- AllenNLP Roberta

ML Models:

- Logistic Regression
- Random Forest
- Neural Network

I choose these 3 models, because I want to see the performance of the models in a simple baseline model (LR), a model that is performed well in unbalanced dataseet and a more complex model (NN).

The steps followed:

- Create data for the modelling, get labels and prepare data for modelling
- Split data to train-test set, do 10-fold cross validation for model parameter tuning using train dataset to select best estimators for each model type
- Apply test set on the refitted ML models with best parameters on the all training dataset

- Check Feature Importance for Random Forest and coeff weights for Logistic Regression models.

I only used the data tweeted during the superbowl 49 game and English tweets.

In addition to the prediction task, I want to check the importance of the models I created to see whether I can say anything additional about the players. Who is the most important player for the game. Can we tell who might be the MVP of the game? Along with player's position can we tell who could be the best Receiver, Defense player?

Creating the data to be used for prediction task:¶ How many tweeets each player received during game?

```
#gopatriots            57138
#gohawks               19203
Tom Brady               5977
Russell Wilson          1302
Chris Matthews          1100
Marshawn Lynch           864
Julian Edelman           646
Rob Gronkowski           571
Doug Baldwin             354
Brandon LaFell           272
Richard Sherman          251
Earl Thomas              240
Jermaine Kearse          221
Danny Amendola           220
Darrelle Revis           154
LeGarrette Blount        137
Kyle Arrington           129
Shane Vereen             113
Jamie Collins            106
Bobby Wagner             103
Vince Wilfork             77
Chandler Jones            71
Michael Hoomanawanui      69
Tony McDaniel             64
Kevin Williams            55
Kam Chancellor            54
Bruce Irvin               46
Cliff Avril               39
K.J. Wright               38
Bryan Walters             33
Rob Ninkovich             30
Ricardo Lockette          30
Dont'a Hightower          26
Brandon Browner           26
Patrick Chung             15
Byron Maxwell             12
Michael Bennett           10
Devin McCourty             5
Robert Turbin              3
Name: player2, dtype: int64
```

How Data Looks Like?¶ Game took 216 minutes, that is why we have 216 rows, samples in our dataset and I found tweets posted on 40 players during game + 2 team columns, 42 columns. Below, we can see the data created for AllenNLP Roberta Positive Score Features for the first 5 minute of the game. Notice that the data seems very sparse for some of the players.

Data Preview for the first 6 columns of first 5 minutes in the game:

```
(216, 42)
```

player2	post_time_1_intervals	#gohawks	#gopatriots	Bobby Wagner	Brandon Browner	Brandon LaFell
0	2015-02-01 15:30:00-08:00	0.765169	0.915447	NaN	NaN	NaN
1	2015-02-01 15:31:00-08:00	0.815970	0.918031	NaN	NaN	NaN
2	2015-02-01 15:32:00-08:00	0.781914	0.905448	NaN	NaN	0.951774
3	2015-02-01 15:33:00-08:00	0.733644	0.895856	NaN	NaN	NaN
4	2015-02-01 15:34:00-08:00	0.657131	0.878550	NaN	NaN	NaN

As expected, data becomes less sparse for the most tweeted players. Below you can see only the most tweeted players and randomly selected 10 minutes of the game:

(post_time_1_intervals won't be used in the modelling, I put in here just for showing what time in the game the selected 1-minute interval corresponds to)

player2	post_time_1_intervals	#gohawks	#gopatriots	Tom Brady	Russell Wilson	Chris Matthews
201	2015-02-01 18:51:00-08:00	0.768218	0.672990	0.713158	0.709041	0.781835
213	2015-02-01 19:03:00-08:00	0.548149	0.745636	0.819526	0.460369	0.000856
138	2015-02-01 17:48:00-08:00	0.700695	0.559546	0.433796	0.999411	0.809007
177	2015-02-01 18:27:00-08:00	0.532102	0.788759	0.660526	NaN	0.999224
15	2015-02-01 15:45:00-08:00	0.630571	0.881566	0.666181	NaN	0.285398
111	2015-02-01 17:21:00-08:00	0.765330	0.791283	0.999210	0.943161	NaN
182	2015-02-01 18:32:00-08:00	0.700104	0.748749	0.829745	0.947678	0.996738
73	2015-02-01 16:43:00-08:00	0.684693	0.853567	0.571063	0.998351	NaN
205	2015-02-01 18:55:00-08:00	0.640510	0.675349	0.702878	0.647346	0.999822
139	2015-02-01 17:49:00-08:00	0.737002	0.507526	0.465867	0.602391	0.757743

player2	Marshawn Lynch	Julian Edelman	Rob Gronkowski	leading
201	0.678363	0.892279	0.990402	Patriots
213	0.163377	0.828680	0.000495	Patriots
138	NaN	NaN	0.382201	Seahawks
177	NaN	0.606261	NaN	Seahawks
15	0.201507	0.999847	0.786160	Tie
111	0.998131	NaN	0.572241	Tie
182	0.999836	0.462827	0.995795	Patriots
73	0.997983	NaN	0.320715	Tie
205	0.757766	0.990173	NaN	Patriots
139	NaN	NaN	NaN	Seahawks

In the prediction task I will only use a subset of the players, since for some players the data is very very sparse, it might hurt prediction rather than help. As I experimented with the full dataset, I realized that those sparse columns are creating noise to confuse model than to increase prediction power. For example, for Rober Turbin I could only match 3 tweets in the entire game, in the best case if every tweet is posted in separate minutes, we only have 3 rows filled out of 216.

That is why, I will only use a subset of these 42 columns and select the best ones based on their density level, by dropping the most sparse columns.

Parameter Tuning Experiments:¶ For parameter tuning I applied 10-fold cross validation on training dataset (contains 80% of samples). For each model I tuned the following parameters:

Logistic Regression:

- penalty: ['none','l1','l2'],
- class_weight: [None, 'balanced'],
- C: [10**(x) for x in range(-2,2)],

Random Forest:

- max_features: ['auto', None, 1, 2, 3, 4, 6] ,
- n_estimators: [5, 10, 20, 40, 60],
- max_depth: [2, 4, 6, None],
- class_weight: [None, 'balanced', 'balanced_subsample'],

Neural Network:

- hidden_layer_sizes: [(5,), (8,), (10,), (30,),(50,),(100,), (120,), (30,30), (50, 50)],
- alpha: [10**(x) for x in range(-5,3)] ,
- learning_rate_init:[0.001, 0.01, 0.1],
- activation:['relu','logistic','tanh']

For Logistic Regression and Neural Network, I applied 10-fold cross validation on gridsearch, for Random forest I used 10-fold cross validation on randomized search with 100 samples.

In this experiment I only kept selected denser columns that have at least 50% of the rows that are non-nulls. This corresponds to 8 out of 42 columns. The columns kept are: ['#gohawks', '#gopatriots', 'Chris Matthews', 'Earl Thomas', 'Marshawn Lynch', 'Rob Gronkowski', 'Russell Wilson', 'Tom Brady']

I also filled the remaining null values with 0s as 0 is the neutral score in this case:

```
-- Info about data:

X shape: (216, 8)

'Class Distribution:'

Tie         98
Patriots    67
Seahawks    51
Name: leading, dtype: int64

'Class Distribution (normalized):'

Tie         0.453704
Patriots    0.310185
Seahawks    0.236111
Name: leading, dtype: float64

   Dropped columns where less than 50.0% is empty
    X shape after dropping columns: (216, 8)
        Remaining columns:                  ['#gohawks',
'#gopatriots', 'Chris Matthews', 'Earl Thomas', 'Marshawn Lynch', 'Rob
Gronkowski', 'Russell Wilson', 'Tom Brady']
    Filling NaN emotions scores with 0..
Model Experiments for : sent_nltk_vader_compound_1_mins data:
Model Experiments for : sent_textblob_polarity_1_mins data:
Model Experiments for : sent_cardiffnlp_roberta_pos_1_mins data:
```

`Model Experiments for : sent_allennlp_roberta_pos_1_mins data:`

Top 3 Best Features Found for each feature set and model pairs:¶ Below, I shared the top 3 best tuned parameter combination for each feature set, model pairs along with 10-fold training and validation scores.

`sent_nltk_vader_compound_1_mins Features - LR Model CV Results`

	mean_train_score	mean_validation_score	param_model___C	param_model___class_weight
9	0.709271	0.655229	0.10	balanced
22	0.712501	0.655229	10.00	balanced
3	0.709271	0.655229	0.01	balanced

	param_model___penalty	rank_test_score
9	none	1
22	l1	1
3	none	1

`sent_nltk_vader_compound_1_mins Features - RF Model CV Results`

	mean_train_score	mean_validation_score	param_model___n_estimators	param_model___max_features
77	0.950930	0.743137	40	3
74	0.788726	0.738562	20	3
80	0.880494	0.738235	20	3

	param_model___max_depth	param_model___class_weight	rank_test_score
77	6.0	NaN	1
74	2.0	NaN	2
80	4.0	NaN	3

`sent_nltk_vader_compound_1_mins Features - NN Model CV Results`

	mean_train_score	mean_validation_score	param_model___hidden_layer_sizes	param_model___alpha
101	0.754512	0.609150	(120,)	0.0100
575	0.642145	0.604575	(10,)	1.0000
38	0.708638	0.604248	(30,)	0.0001

`sent_textblob_polarity_1_mins Features - LR Model CV Results`

	mean_train_score	mean_validation_score	param_model___C	param_model___class_weight
9	0.647294	0.633333	0.10	balanced
3	0.647294	0.633333	0.01	balanced
21	0.647294	0.633333	10.00	balanced

	param__model___penalty	rank__test__score
9	none	1
3	none	1
21	none	1

`sent_textblob_polarity_1_mins Features - RF Model CV Results`

	mean__train__score	mean__validation__score	param__model___n__estimators	param__model___max__features
87	0.891487	0.726797	40	2
58	0.898580	0.715033	20	6
15	0.974805	0.709477	20	6

	param__model___max__depth	param__model___class__weight	rank__test__score
87	4.0	balanced__subsample	1
58	4.0	balanced__subsample	2
15	6.0	balanced__subsample	3

`sent_textblob_polarity_1_mins Features - NN Model CV Results`

	mean__train__score	mean__validation__score	param__model___hidden__layer__sizes	param__model___alpha
119	0.709305	0.628758	(30,)	0.100
131	0.679535	0.623203	(30, 30)	0.100
65	0.714491	0.622876	(30,)	0.001

`sent_cardiffnlp_roberta_pos_1_mins Features - LR Model CV Results`

	mean__train__score	mean__validation__score	param__model___C	param__model___class__weight
20	0.748710	0.697386	10.00	NaN
19	0.746125	0.697386	10.00	NaN
0	0.741605	0.685621	0.01	NaN

	param__model___penalty	rank__test__score
20	l2	1
19	l1	1
0	none	3

`sent_cardiffnlp_roberta_pos_1_mins Features - RF Model CV Results`

	mean__train__score	mean__validation__score	param__model___n__estimators	param__model___max__features
34	0.892765	0.750980	10	NaN
60	0.770670	0.745752	40	NaN

	mean_train_score	mean_validation_score	param_model__n_estimators	param_model__max_features
64	0.983192	0.745098	40	4

	param_model__max_depth	param_model__class_weight	rank_test_score
34	4.0	balanced	1
60	2.0	balanced	2
64	6.0	NaN	3

`sent_cardiffnlp_roberta_pos_1_mins Features - NN Model CV Results`

	mean_train_score	mean_validation_score	param_model__hidden_layer_sizes	param_model__alpha
11	0.658140	0.646732	(30,)	0.00001
149	0.676984	0.634314	(50,)	1.00000
92	0.649074	0.629412	(30,)	0.01000

`sent_allennlp_roberta_pos_1_mins Features - LR Model CV Results`

	mean_train_score	mean_validation_score	param_model__C	param_model__class_weight
20	0.724801	0.668954	10.00	NaN
0	0.724835	0.657516	0.01	NaN
6	0.724835	0.657516	0.10	NaN

	param_model__penalty	rank_test_score
20	l2	1
0	none	2
6	none	2

`sent_allennlp_roberta_pos_1_mins Features - RF Model CV Results`

	mean_train_score	mean_validation_score	param_model__n_estimators	param_model__max_features
25	0.998056	0.703595	20	6
17	0.991609	0.698693	40	3
72	0.970268	0.693137	10	3

	param_model__max_depth	param_model__class_weight	rank_test_score
25	NaN	NaN	1
17	6.0	balanced	2
72	6.0	balanced_subsample	3

`sent_allennlp_roberta_pos_1_mins Features - NN Model CV Results`

	mean_train_score	mean_validation_score	param_model___hidden_layer_sizes	param_model___alpha
68	0.594378	0.557516	(50,)	0.001
97	0.625987	0.552288	(100,)	0.010
583	0.617495	0.547059	(100,)	1.000

Evaluation Results¶

For evaluating I used accuracy as my metric.

Below table shows 10-fold average train and validation scores and final test score for each model and feature types:

	Mean Train Score			Mean Validation Score			Mean Test Score		
Model Name	LR	NN	RF	LR	NN	RF	LR	NN	RF
Sentiment Feature									
Nltk Vader Compound	0.709	0.755	0.951	0.655	0.609	0.743	0.773	0.432	0.727
Textblob Polarity	0.647	0.709	0.891	0.633	0.629	0.727	0.750	0.705	0.705
Cardiffnlp Roberta Pos	0.746	0.658	0.893	0.697	0.647	0.751	0.750	0.477	0.795
Allennlp Roberta Pos	0.725	0.594	0.998	0.669	0.558	0.704	0.705	0.568	0.659

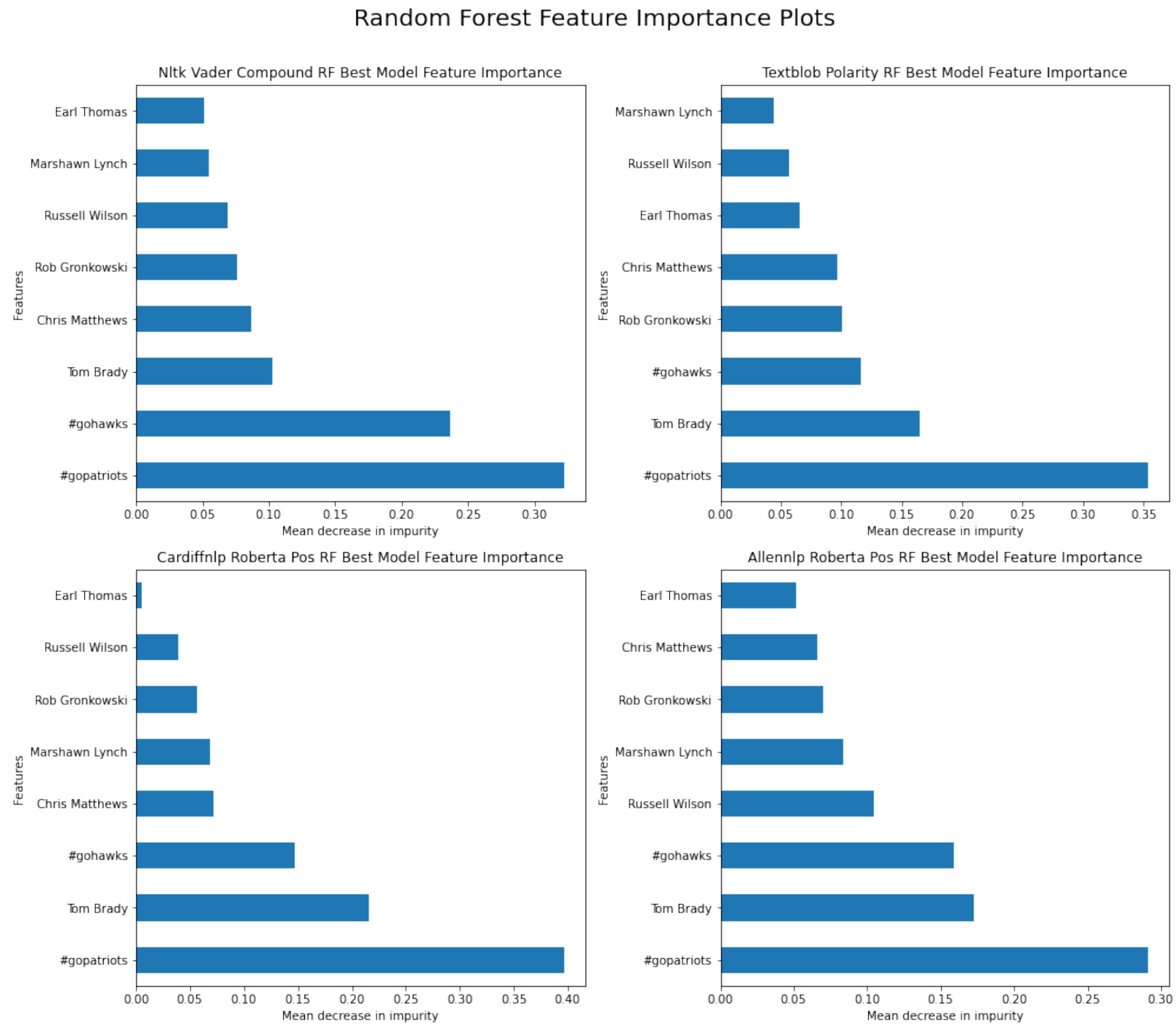
We need to be mindful while checking these results given that we have 1-minute intervals, the samples in the data is 216, very small dataset. Test sample corresponds only 20% of the data, which is around 43 samples. Hence, test samples are small and might not reflect the overall game status distribution, and should be taken into account with caution.

I will mainly focus on validation scores: the best feature set for this prediction task seems to be Cardiffnlp Roberta Pos and best model is Random Forest. With Random Forest on CardiffNLP features, we can reach up to 75% mean validation accuracy and 79.5% test accuracy.

I also did the same experimentation on different subset of columns (Players) based on different density levels with fewer and more columns than I had in this data. 30% and 80% density selected feature experiment results are shared in the appendix.

Interpreting Models¶

RF feature importances and LR coeff plots for each feat type¶ Below you can seee the feature importances for each feature set on the best estimator found from parameter tuning.




```
Index(['Player', 'Pos', 'Team', 'espn_team_list', 'Abbreviation', 'Position',
        'Category', 'player_first_name', 'player_last_name'],
      dtype='object')
```

	Player	Pos	Team	espn_team_list	Abbreviation	Position	Category
0	Tom Brady	QB	Patriots	True	QB	Quarterback	Offense
4	Rob Gronkowski	TE	Patriots	True	TE	Tight End	Offense
21	Russell Wilson	QB	Seahawks	True	QB	Quarterback	Offense
22	Marshawn Lynch	RB	Seahawks	True	RB	Running Back	Offense
42	Earl Thomas	FS	Seahawks	True	FS	Free Safety	Defense
46	Chris Matthews	WR	Seahawks	True	WR	Wide Receiver	Offense

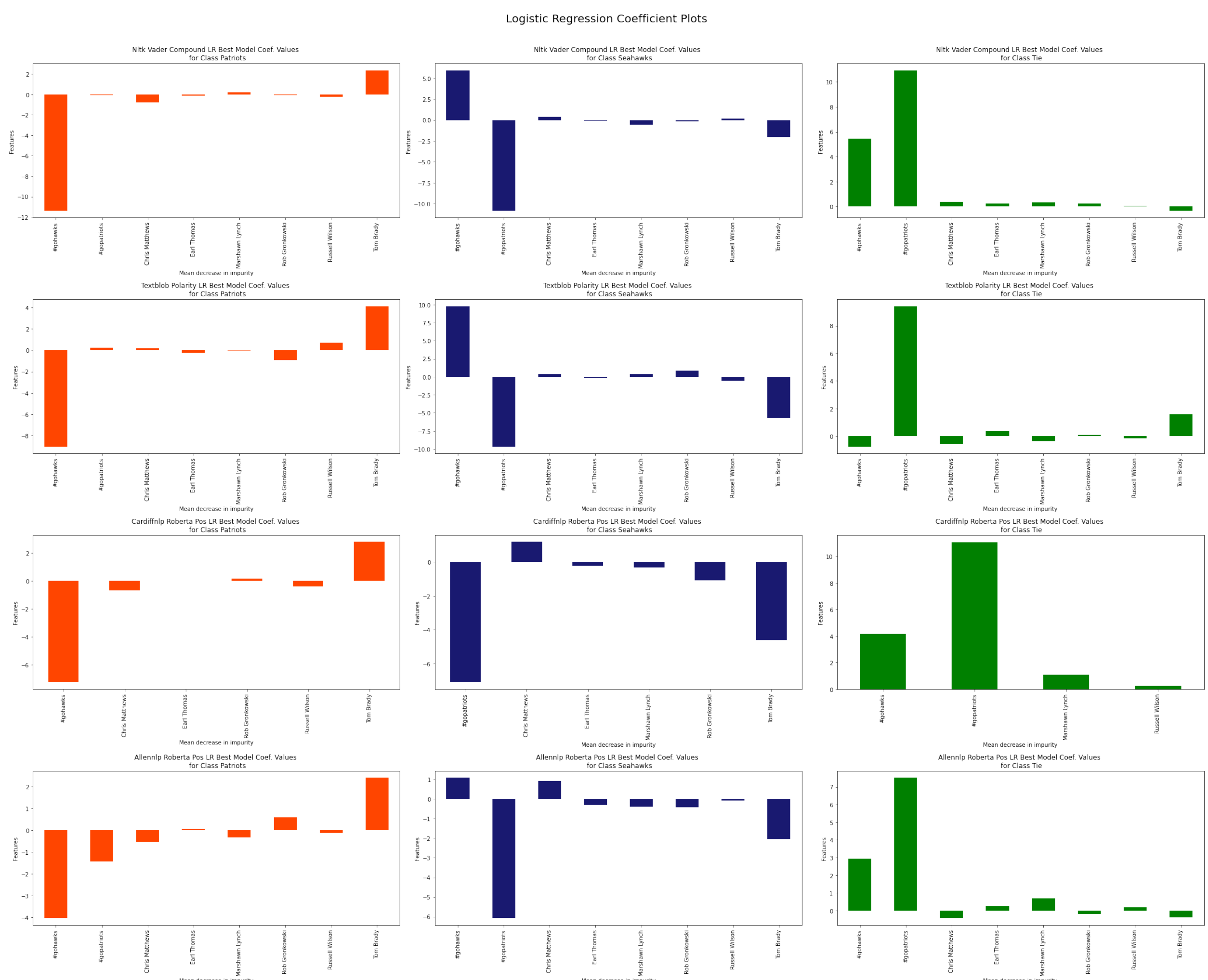
Overall, we see that #gohawks and #gopatriots are the two columns that have high influence on the model results. We can see that Tom Brady is the player who has the most importance in all 4 models independent of which feature set we use.

When we check AllenNLP RF feature importances, what I see is that the paired most important players shown in order. The first two most important players are Tom Brady and his counterpart Russell Wilson, following them Marshawn Lynch and his Patriots counterpart Rob Gronkowski. TexBlob, seems to put more importance on Patriots' players.

Best Model we found was CardiffNLP's Random Forest: We can see that for this model the second most important feature and most important player is Tom Brady. Followed by, Chris Matthews, Marshawn Lynch, Rob Gronkowski and Russell Wilson.

Can we guess who might be the MVP of the game by looking at the Random Forests' importance for each model? All seems to point Tom Brady and indeed he was the MVP. If we look at the department of each player can we also guess who might be best Receiver and Defense player? For the CardiffNLP feature set, the second most important player is Chris Matthews, he is a receiver from Seahawks, when I check the Team Stats from ESPN site, he is indeed the Game Leader for Receiving Yards for Seahawks, and has one of the best stats makes him a good candidate for best receiver. The third importan player is Marshawn Lynch, who is also selected as Rushing Leader of the Game for his team and has better game stats than his Patriots counterpart, L. Blount.

LR coeff plots for each Sentiment type¶ In this part, we show the coefficient plots of each class for best LR models' on each feature type. I only show non-zero coefficients in the plot.



- We can see that in all 4 feature sets, Tom Brady get the highest positive coeff value for assigning Patriots as leading game, and highest negative coeff value for class Seahawks. For Tie assignment, Tom Brady is not as important.

Patriots Coefficients:

- In all 4 models, negative #gohawks and positive Tom Brady coeffs are found the most relevant informations for assigning Patriots as class. This means there is a negative correlation on positive overall positive Seahawks team and positive correlation on positive Tom Brady sentiments to predict Patriots leading the game. One interesting thing is that AllenNLP was able to catch correct players for each team: assigned positive coeff. values to Patriots players and negative coeff. values to Seahawks players

Seahawks Coefficients:

- We can see Tom Brady as the highest negative coeff. value for the players for all 4 LR model coefficients on assigning Seawhawks class. In all 4 models, the prediction is mainly driven on overall team emotions and Tom Brady. One interesting thing to notice is that both of the Roberta feature sets gave a higher positive value to Chris Matthews (and also only positive coeff. to a player). We can interpret this as perceived emotion against Chris Matthews is positively correlated in predicting the times when Seahawks leading the game and Chris Matthews played an important role as a receiver for Seahawks team for them to score.

Overall, from these plots, if we would guess and MVP it would be Tom Brady, who had high absolute value of coefficient amongs all players for each class. Chris Matthews can potentially be selected as best receiver for the Seahawks team.

Conclusion¶

I learned more about American Football, Patriots and Seahawks than I could ever imagine. Probably now I should go and try to see a game. The results for both Character-Centric Time-Series Tracking and Winner Prediction tasks were very interesting and full of insights. Fan based emotions about players in time windows are indeed good signals for understanding game status, scores and even the MVP of the game. Can we tell which feature type is better for the two tasks? For the Character-Centric Time-Series Tracking, both of the Roberta models offer more accurate and relevant correlations between scores and the emotions perceived by fans for players and teams. Rule-based nltk and textblob, seems to underperform compared the other two feature types, textblob seem to be the worst one, in this context and dataset. For Winner Prediction task in a given time interval, CardiffNLP seem to be the best performing feature set combined with a Random Forest model that leads almost 75% validation and 79.5% test accuracy.

This project should be taken as an preliminary exploration for the given tasks and can be improved in many different ways, we can deep dive even further in almost every step.

Appendix¶

Winner Prediction Experiments with different number of players¶

Results for experiment with 30% density requirement on columns¶

```
-- Info about data:

X shape: (216, 11)

'Class Distribution:'

Tie         98
Patriots    67
Seahawks    51
Name: leading, dtype: int64

'Class Distribution (normalized):'

Tie         0.453704
Patriots    0.310185
```

```
Seahawks    0.236111
Name: leading, dtype: float64
```

```
   Dropped columns where less than 30.0% is empty
    X shape after dropping columns: (216, 11)
        Remaining columns:                  ['#gohawks',
'#gopatriots', 'Chris Matthews', 'Earl Thomas', 'Jamie Collins',
'Julian Edelman', 'Marshawn Lynch', 'Richard Sherman', 'Rob
Gronkowski', 'Russell Wilson', 'Tom Brady']
    Filling NaN emotions scores with 0..
Model Experiments for : sent_nltk_vader_compound_1_mins data:
Model Experiments for : sent_textblob_polarity_1_mins data:
Model Experiments for : sent_cardiffnlp_roberta_pos_1_mins data:
Model Experiments for : sent_allennlp_roberta_pos_1_mins data:
```

	Mean Train Score			Mean Validation Score			Mean Test Score		
Model Name	LR	NN	RF	LR	NN	RF	LR	NN	RF
Sentiment Feature									
Nltk Vader Compound	0.716	0.711	1.000	0.656	0.616	0.750	0.773	0.682	0.727
Textblob Polarity	0.660	0.713	0.988	0.623	0.605	0.727	0.705	0.614	0.705
Cardiffnlp Roberta Pos	0.738	0.650	0.983	0.675	0.583	0.750	0.727	0.727	0.773
Allennlp Roberta Pos	0.735	0.615	0.990	0.669	0.528	0.722	0.682	0.500	0.659

Results for experiment with 80% density requirement on columns¶

```
-- Info about data:
```

```
X shape: (216, 5)
```

```
'Class Distribution:'
```

```
Tie         98
Patriots    67
Seahawks    51
Name: leading, dtype: int64
```

```
'Class Distribution (normalized):'
```

```
Tie         0.453704
Patriots    0.310185
Seahawks    0.236111
Name: leading, dtype: float64
```

```
   Dropped columns where less than 80.0% is empty
    X shape after dropping columns: (216, 5)
        Remaining columns:                  ['#gohawks',
'#gopatriots', 'Rob Gronkowski', 'Russell Wilson', 'Tom Brady']
    Filling NaN emotions scores with 0..
Model Experiments for : sent_nltk_vader_compound_1_mins data:
Model Experiments for : sent_textblob_polarity_1_mins data:
Model Experiments for : sent_cardiffnlp_roberta_pos_1_mins data:
Model Experiments for : sent_allennlp_roberta_pos_1_mins data:
```

	Mean Train Score			Mean Validation Score			Mean Test Score		
Model Name	LR	NN	RF	LR	NN	RF	LR	NN	RF
Sentiment Feature									
Nltk Vader Compound	0.702	0.709	0.767	0.685	0.644	0.732	0.773	0.750	0.705
Textblob Polarity	0.686	0.694	0.860	0.669	0.658	0.692	0.727	0.682	0.682
Cardiffnlp Roberta Pos	0.745	0.682	0.770	0.721	0.645	0.751	0.750	0.523	0.682
Allennlp Roberta Pos	0.715	0.645	0.921	0.693	0.617	0.710	0.727	0.523	0.682