# Comparative study of the Twitter Sentiment of a set of football superstars

GROUP 2:

Moghe Ritwik Prashant

Pranita Khandelwal

Riju Bhattacharyya

Sushant Rajput


LEO
MESSI


LAURENCE GRIFFITHS/GETTY IMAGES EUROPE/GETTY IMAGES

**Obvious Owen**
@MrObviousOwen

Jamie Vardy has now scored in 11 consecutive matches

That means in the last 11 games he's scored in every one

Incredible

11:27 PM - 28 Nov 2015

641

Lineker

Follow

Lineker

indisputably the greatest player

**sportingintelligence**
@sportingintel

On this corresponding weekend four years ago, Vardy scored in a 1-1 draw at Gateshead for Fleetwood. In the conference. Now he's worth £30m.

11:42 PM - 28 Nov 2015

100

@noahlove

Yep, Messi about to lose La Liga on the last

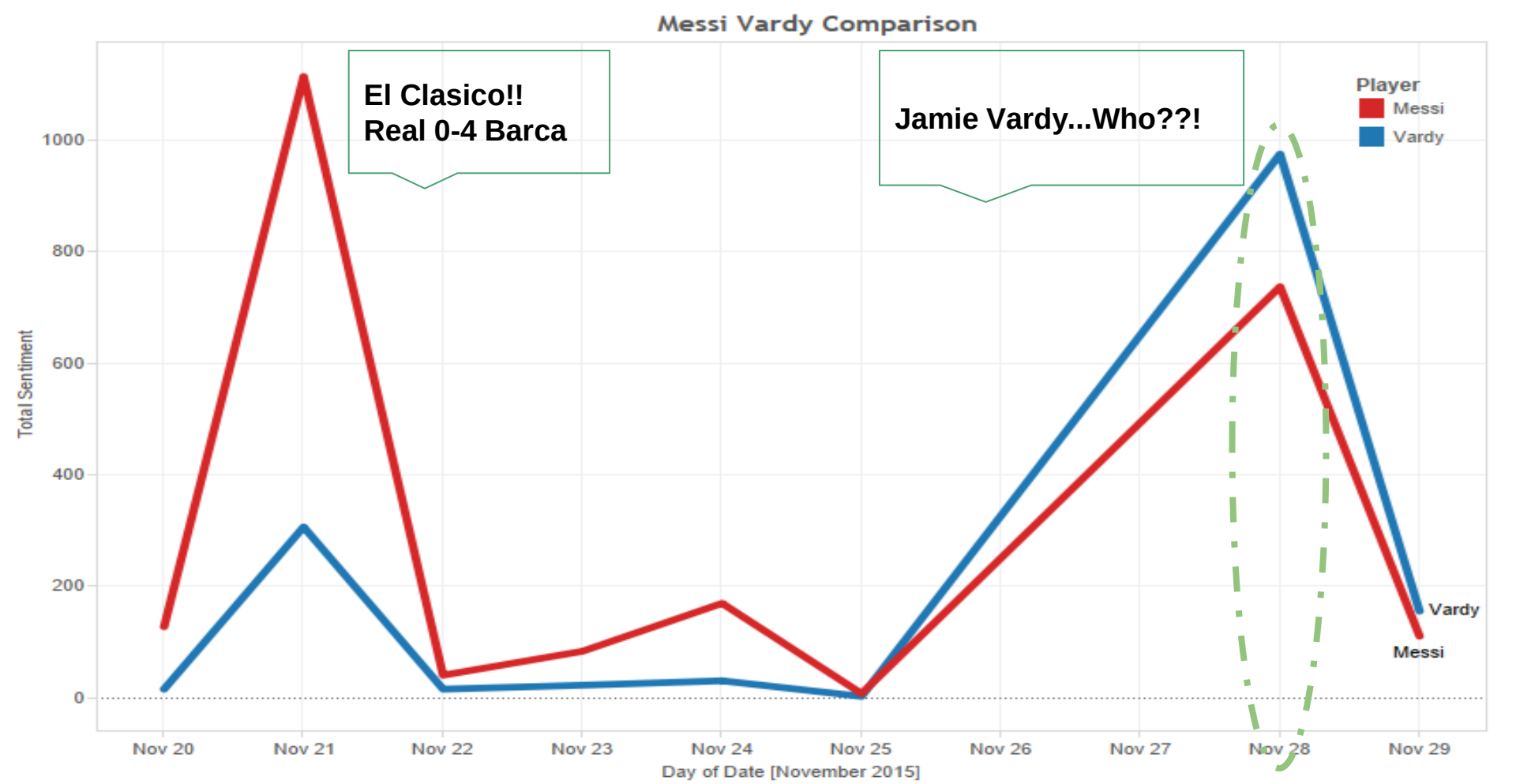
Favorited ••• More

**Gary Lineker**
@GaryLineker

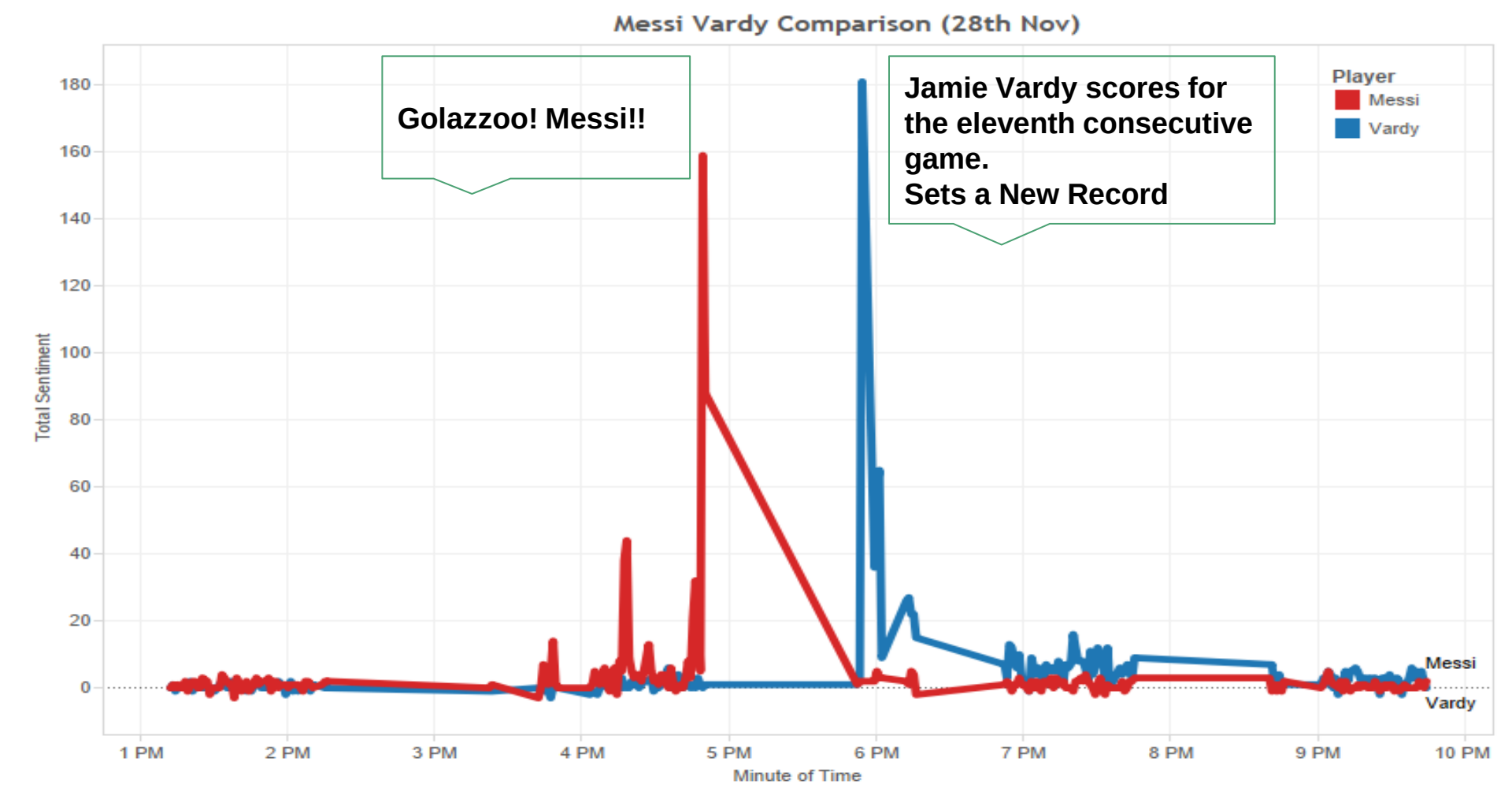
Vardy! He scores when he wants.

11:26 PM - 28 Nov 2015

5,119




The trend of sum of Rating for Date Day. Color shows details about Player. The marks are labeled by Player. The data is filtered on Date Day, which keeps 8 of 17 members. The view is filtered on Player, which keeps Messi and Vardy.




The trend of sum of Rating for Time Minute. Color shows details about Player. The marks are labeled by Player. The data is filtered on Date Day, which keeps 28. The view is filtered on Player, which keeps Messi and Vardy.

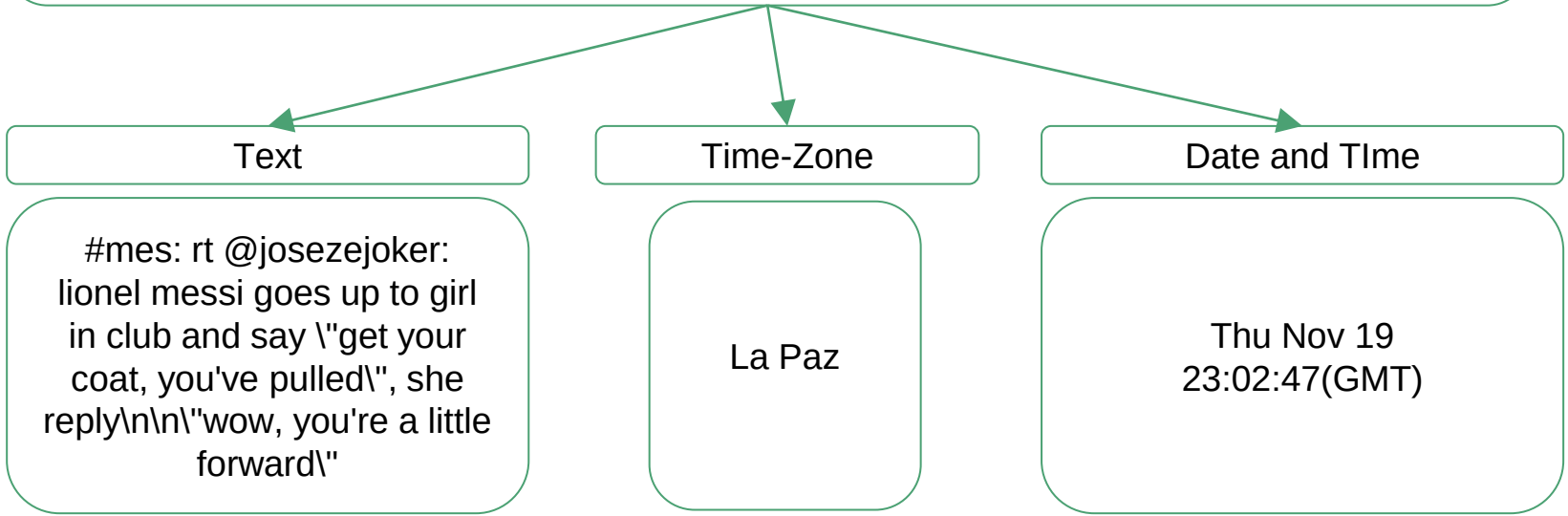
Twitter Streaming API on Python

Real Time Unlimited Data

# Data Extraction

Lots of Meta-Deta

{"*created_at*":"Thu Nov 19 23:02:47 +0000 2015","*id*":667478151535136768,"*id_str*":"667478151535136768","*text*":"RT @JoseZeJoker: Lionel Messi goes up to girl in club and say \"Get your coat, you've pulled\", she reply\n\n\"Wow, you're a little forward\"", "*time_zone*":"La Paz","*geo_enabled*":true,"*lang*":"en"….}



# Data Transformation- Unsupervised

#mes: rt @josezejoker: lionel messi goes up to girl in club and say \"get your coat, you've pulled\", she reply\n\n\"wow, you're a little forward\"

| | |
|---|---|
| girl | 1 |
| club | 1 |
| coat | 1 |
| little | 1 |
| forward | 1 |

Lexicon Score* = Sum of Scores of each word in the Tweet

= Score(girl) + Score(club) + … + Score(forward)

= 1

*Bing Liu Opinion Lexicon [6786 words : 2006 +ve , 4783 -ve]

# Data Transformation- Unsupervised

#mes: rt @josezejoker: lionel messi goes up to girl in club and say \"get your coat, you've pulled\", she reply\n\n\"wow, you're a little forward\"

Lexicon Score = 0

#kan: rt @bentrivett1: fucking jamie vardy's an overrated piece of shit!  you f***king harry khiladi, one season wonder c*** #oneseasonwonder"

Lexicon Score = -2

#ron: rt @realmadrid @cristiano i love cristiano ronaldo

Lexicon Score = 1

#cou: And he’s slotted it home after a quick break! Counter attack FTW! Klopp FTW!

Lexicon Score = 0

# Data Classification- Supervised Classification

Bag of Words Representation: Assumes that position of the word in the text is not important

#cou: And he's slotted it home after a quick break! Counter attack FTW! Klopp FTW!

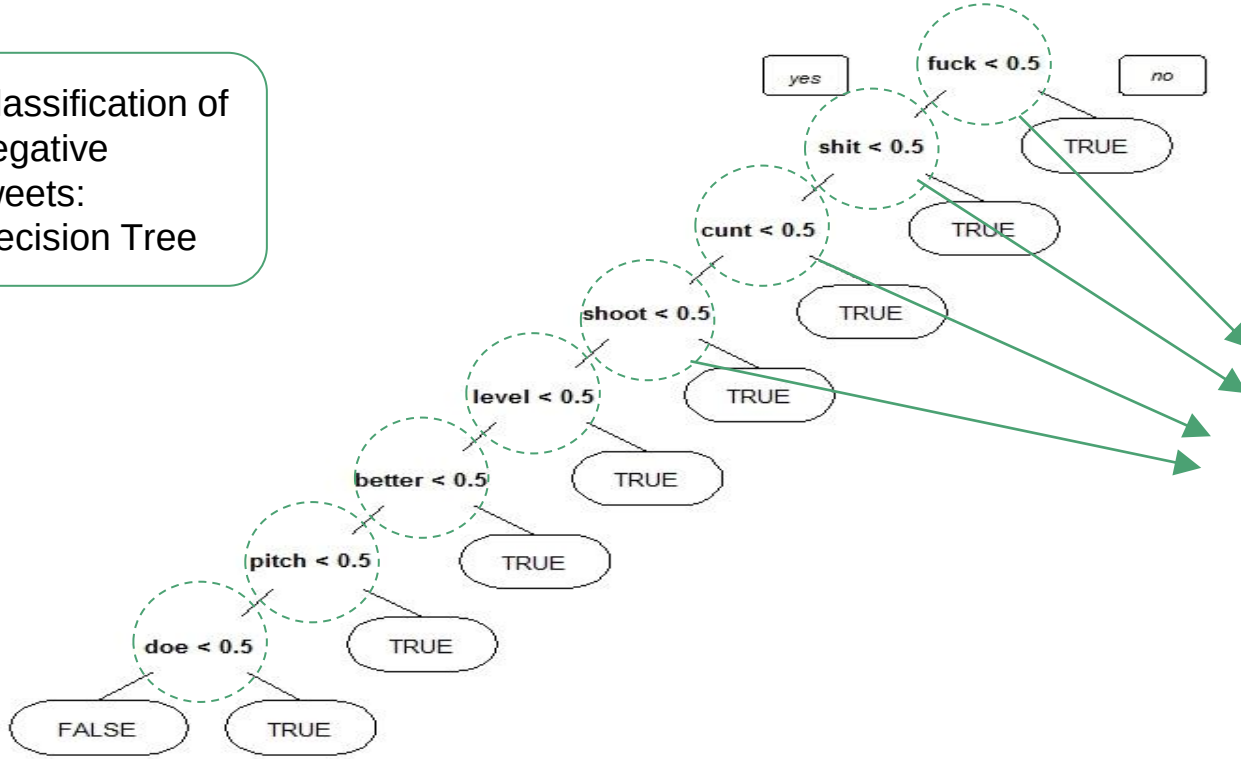
Human Rating = Positive

| | attack | break | counter | coat | FTW | f***king | …….. | ynwa | RATING |
|---|---|---|---|---|---|---|---|---|---|
| tweet1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | -1 |
| tweet2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| tweet3 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| tweet4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Term Document Frequency Matrix

# Data Classification- Supervised Classification

Classification of negative tweets: Decision Tree




Words used as features in the decision tree

# Supervised Classification- Tweets with Comparison?

#mes: rt @opta: **Ozil** is way **better** than **Coutinho** in terms of Big Chances Created per game"

**Ozil…..better**
**Better after Ozil**
**Positive for Ozil**

**In all the tweets about Ozil**
**Replace better by**
**KPOS**

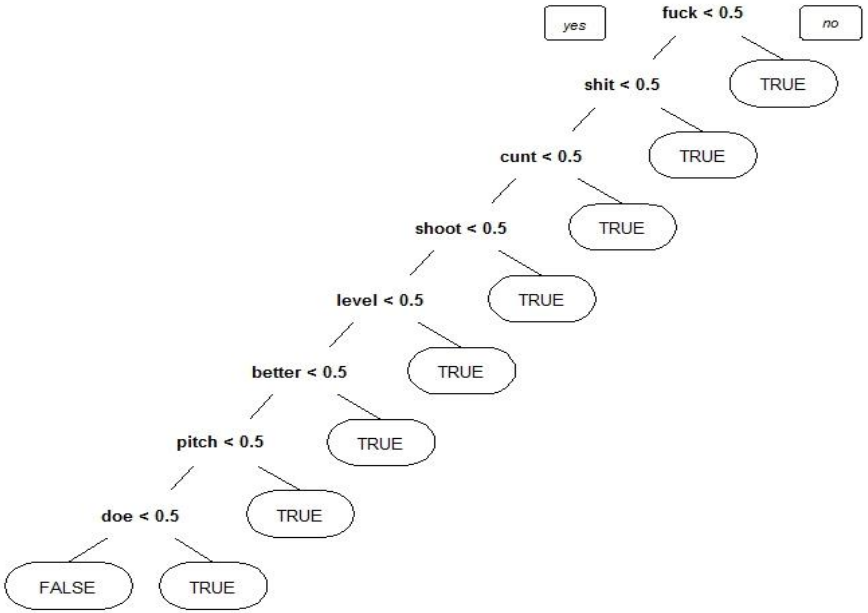
**Opposite if worse is present**

**better….Coutinho**
**Better before Coutinho**
**Negative for Coutinho**

**In all the tweets about Coutinho**
**Replace better by**
**KNEG**

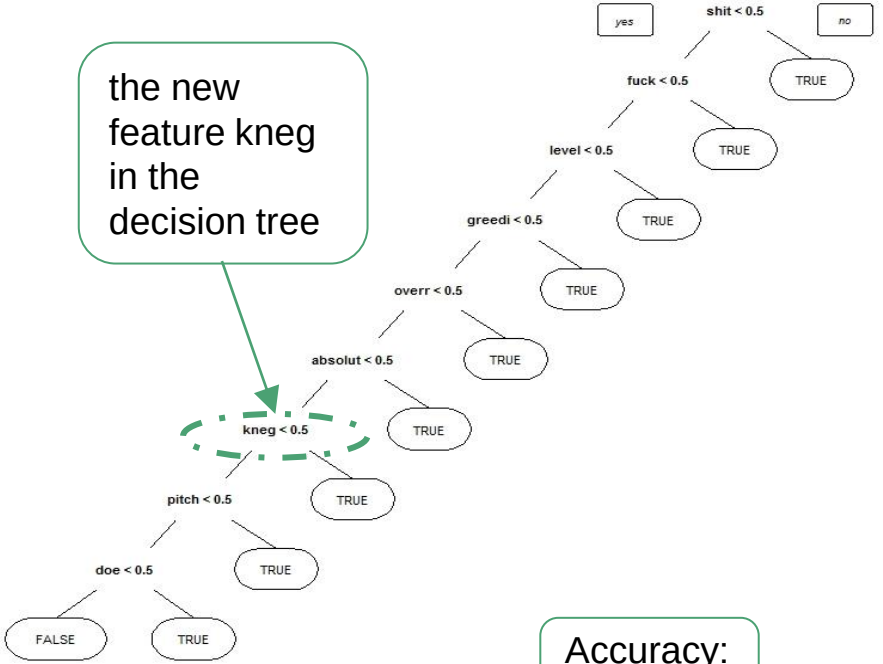
**Opposite if worse is present**

# Supervised Classification- Tweets with Comparison?




Accuracy: 75%

the new feature kneg in the decision tree




Accuracy: 78%

# Supervised Classification- Tweets with Emoticons? Slang Words? Repeating letters?

#mes: god back **\ud83d\ude09\ud83d\ude39\ud83d\ude39\ud83d\ude24** @culedefcb

#kan: **Awesoooome** Goal! **FTW** Hurricane

| | | | |
|---|---|---|---|
| 😘 | kissing_heart | \ud83d\ude18 | 4 |
| 😉 | wink | \ud83d\ude09 | 4 |
| 😋 | yum | \ud83d\ude0b | 4 |
| 😄 | triumph | \ud83d\ude24 | 5 |
| 😢 | cry | \ud83d\ude22 | -2 |
| 😞 | disappointed | \ud83d\ude1e | -2 |
| 😳 | flushed | \ud83d\ude33 | -2 |
| 😨 | fearful | \ud83d\ude28 | -2 |

- **Emoticon Mapping**

#mes: god back **‘wink’ ‘wink’’triumph’**@culedefcb

- **Slang Removal**

#kan: Awesoooome Goal! For The Win hurricane

- **Repeated Letters removal**

#kan: Awesoome Goal! FTW hurricane

# Lexicon Mapping

#cou: And he's slotted it home after a quick break! Counter attack FTW! Klopp FTW!

Lexicon Score = 0

**Lexicon Mapping:** The Lexicon Scores of Each Tweet are added as a feature in the Term Document Matrix

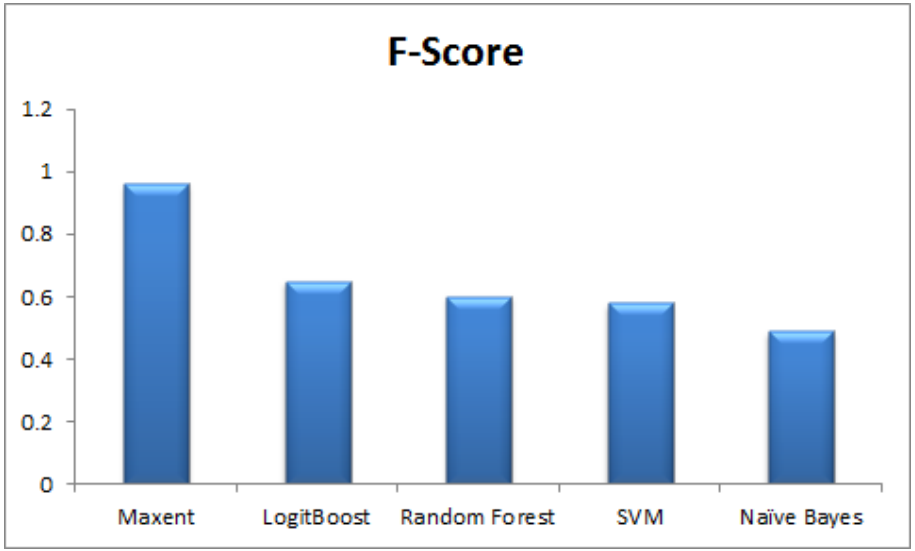
Rationale: All the words relevant to the real-time test data may not be present in the train data

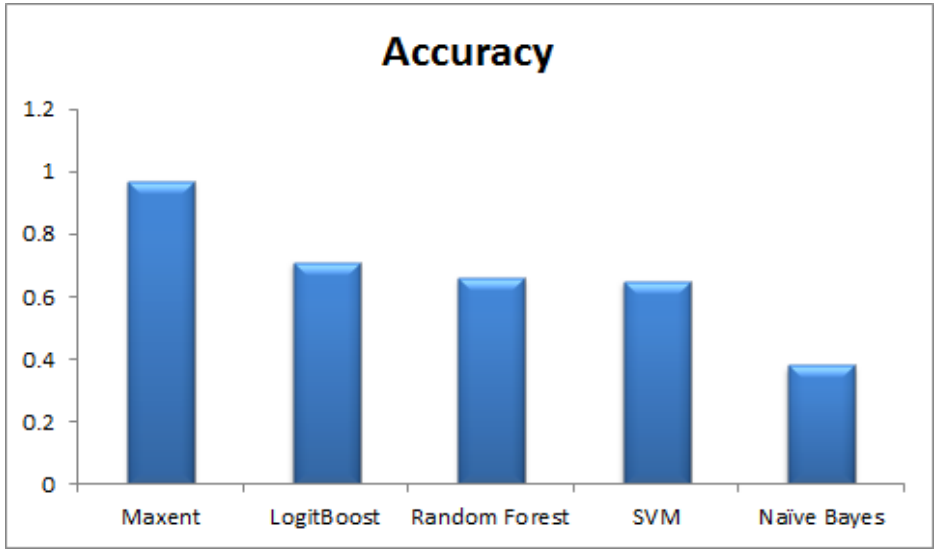
| | Lexicon Rating | attack | break | counter | coat | FTW | f***king | love | one | RATING |
|---|---|---|---|---|---|---|---|---|---|---|
| tweet1 | -4 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | -1 |
| tweet2 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| tweet3 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| tweet4 | +2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Modified Term Document Frequency Matrix

# Comparison of Various Classifiers

Train Data: 4669 unique tweets. Collected uniformly over time for each player. Number of tweets about each player in the train data proportional to the total number of tweets about the player.
Train Data: 781 Negative, 1267 Neutral, 2621 Positive
Classification: Three-Way Classification



# From generative models to discriminative models

Generative modes like Naive Bayes gives the joint probability of the features and tries to maximize the joint likelihood of the data

Assumptions

- Conditional independence
- Position of the word doesn't matter

Cons- Overcounts evidence

For a tweet t and class c

$$P(c \mid t) = \frac{P(t \mid c)\, P(c)}{P(t)}$$

best class that the tweet t belongs to given by

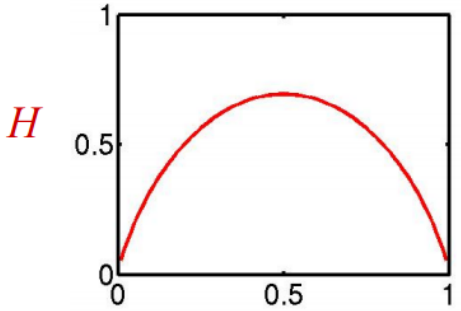
$$C_{best} = \underset{c \in C}{\operatorname{argmax}}\, P(c \mid t) = \underset{c \in C}{\operatorname{argmax}}\, P(x_1, x_2 \ldots x_n \mid c) * P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} \prod P(x_i \mid c) * P(c)$$

# Maximum Entropy Classifier

- Used when we can't assume any probability distribution or conditional independence for our model.
- We want the probabilities to be as uniform as possible.
- uniformity ⇒ high entropy

$$H(p) = E_p[\log_2(1/p_x)] = -\Sigma\, p_x \log_2 p_x$$



A coin-flip is most uncertain for a fair coin.

**Maximize entropy H subject to feature based constraints**

- Adding constraints (features):
  - lowers maximum entropy
  - increases maximum likelihood of data
  - brings distribution closer to data

# Maximum Entropy Classifier

1. For each word w and class c, define a joint feature f(c,w)= N, where N is the number of times the feature appears in the class c

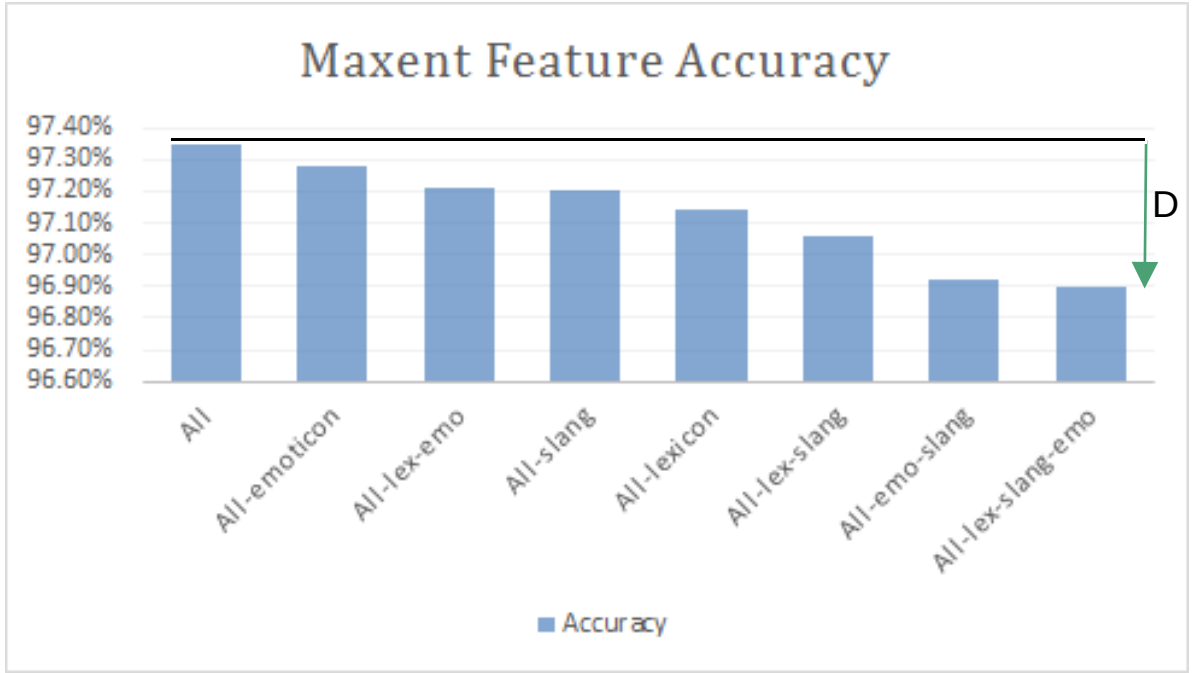
2. Using iterative optimization assign weights to the features in order to maximize the log-likelihood of the training data

3. Probability of a class c given tweet t is given by

$$P(c|d,\lambda) \stackrel{def}{=} \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c' \in C} \exp \sum_i \lambda_i f_i(c',d)}$$

Maxent models don't double count correlated features- this is done by weighing the features so that model expectations match observed expectations

# Effect of Emoticon/Slang/Lexicon




D:
Decrease in Accuracy when the particular cleaning-method is not implemented.

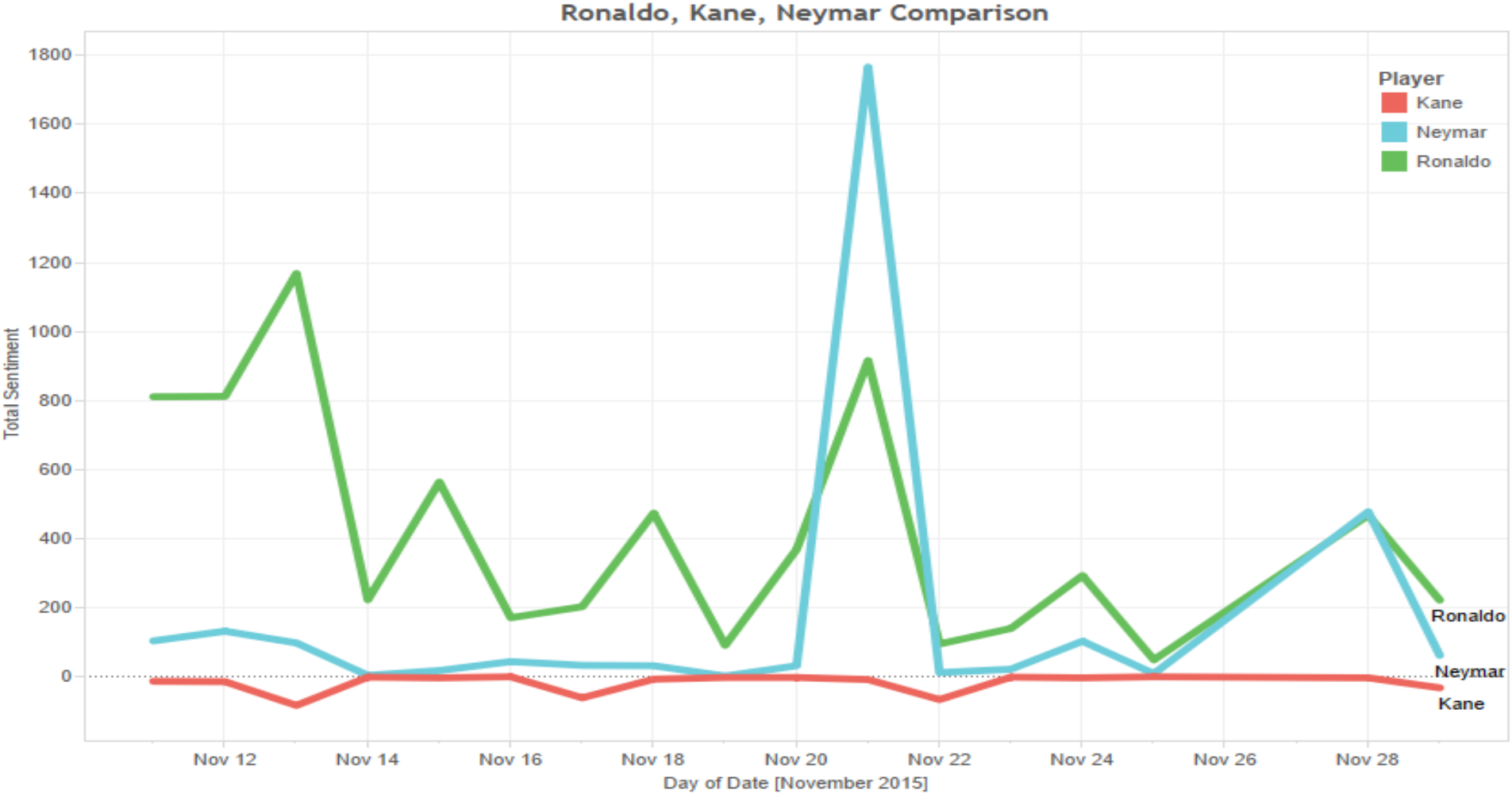
A method with higher D has a higher importance

# Business Applications

Real time quadrant charts for player classification

Comparing popularity of players

Use as a factor in predicting possible ROI (as an endorsee)

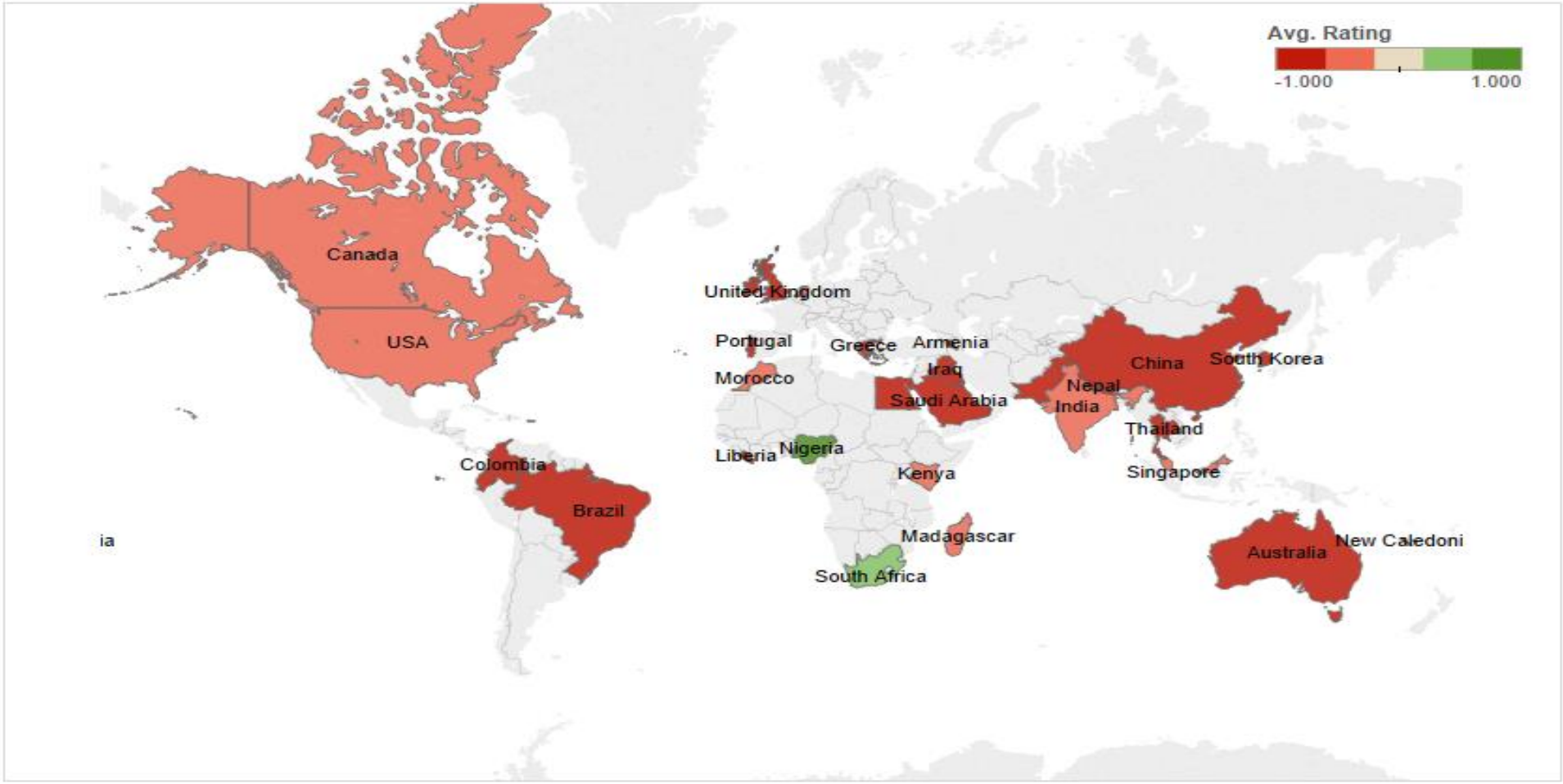



The trend of sum of Rating for Date Day. Color shows details about Player. The marks are labeled by Player. The data is filtered on Date Day, which keeps 17 of 17 members. The view is filtered on Player, which keeps Kane, Neymar and Ronaldo.

# Tweet Count Map


Map based on Longitude (generated) and Latitude (generated). Size shows count of Rating. The marks are labeled by Country. The data is filtered on Player and Date Day. The Player filter keeps 8 of 8 members. The Date Day filter ranges from November 11, 2015 to November 29, 2015.

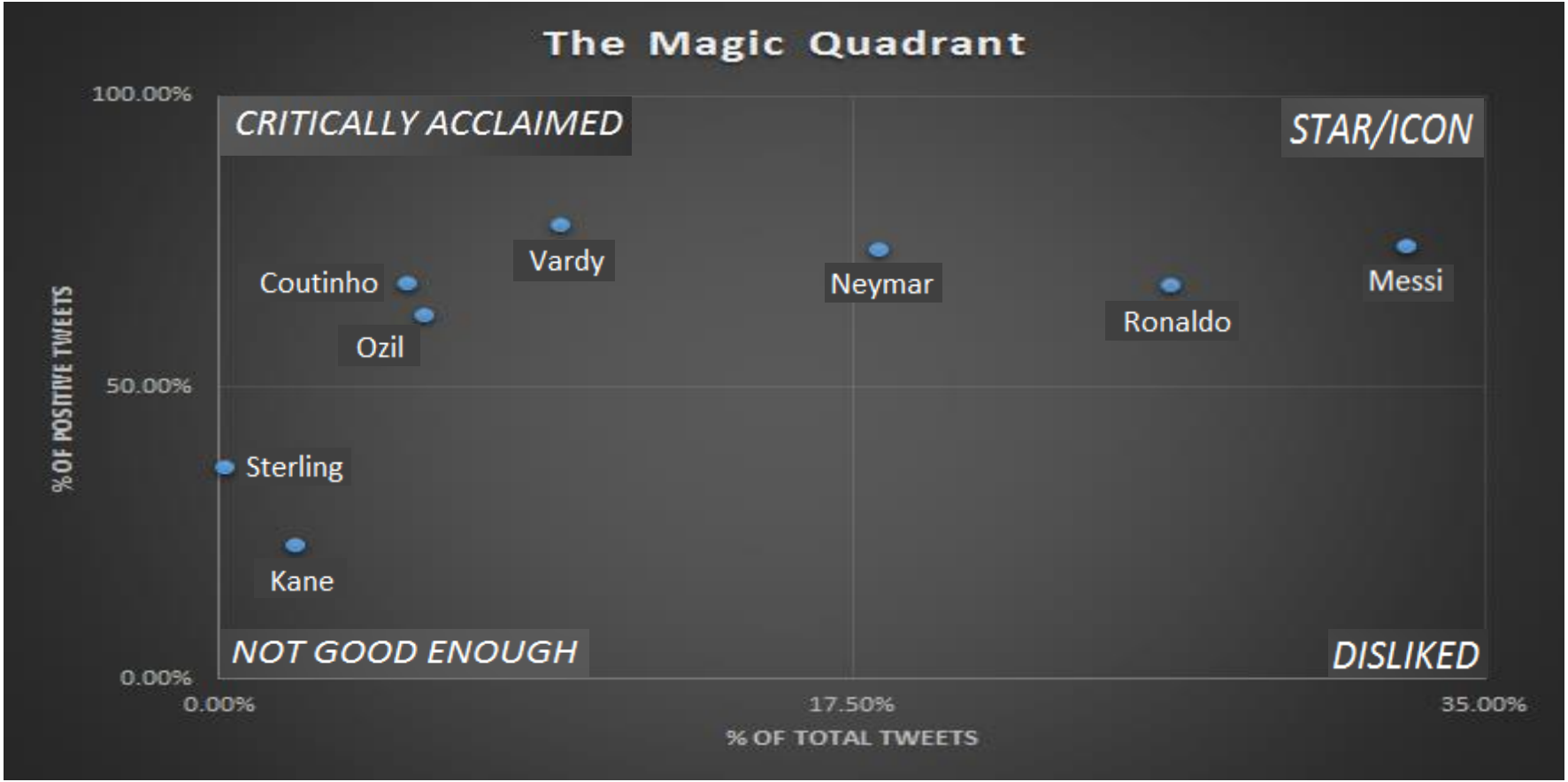
# Kane Sentiment Heat Map




Map based on Longitude (generated) and Latitude (generated). Color shows average of Rating. The marks are labeled by Country. The data is filtered on Player and Date Day. The Player filter keeps Kane. The Date Day filter ranges from November 11, 2015 to November 29, 2015.

# Ronaldo Sentiment Heat Map


Map based on Longitude (generated) and Latitude (generated). Color shows average of Rating. The marks are labeled by Country. The data is filtered on Player and Date Day. The Player filter keeps Ronaldo. The Date Day filter ranges from November 11, 2015 to November 29, 2015.


The Magic Quadrant
100.00%
50.00%
0.00%
% OF POSITIVE TWEETS
CRITICALLY ACCLAIMED
STAR/ICON
NOT GOOD ENOUGH
DISLIKED
Vardy
Coutinho
Ozil
Neymar
Ronaldo
Messi
Sterling
Kane
0.00%
17.50%
35.00%
% OF TOTAL TWEETS