Final Report

Modelling characteristics which produce influence of a social media user

Alisha Gujarathi (801020254) Chirag Jain (801026633) Nikita Nalawade (801024520)

Abstract:
Social media has become a platform for worldwide discussions. It is becoming the epicenter of world's most trending and important topics. The issues like economic, cultural, political, etc., have taken social media as the platform for world-wide discussions. Many people have become the primary source of such topics of discussion, shaping the thinking of mass crowd. It indicates the significance of studying these characteristic trends of users who can mould and determine how the world takes a specific topic. The investigation tries to understand the features of a user which can be modeled to find out what actually makes a user significantly influential. Which characteristics are required to penetrate the crowd and successfully incline the thinking of people into a particular direction. We have implemented various algorithms like LSTM, SVM, Logistic Regression to find out the most important features that affect the decision variable. Pre-processing was done on the data by normalization. As the marketing strategist and social media marketers are attempting to come up with the best method to publicize a product, this investigation can be of huge significance, not just to marketing field but also to counter the effects of negative influencing from sources like political and terrorist groups.

Introduction:
A person is said to be influential in social media space if the person's credibility and reach has the power to mould other users' thinking. If the user has a large reach which can be used for spreading biased motives, the user has influential powers. Although, the definition is clear enough, the perspective of different people remain scattered about how the influence can be established by some user. As a number of theories have different outcomes for the findings of the phenomenon, the investigation here tries to find out which features are responsible for developing a user's influential power. As we don't have access to time-series and live twitter data, the temporal analysis hasn't been performed which even asks for totally different approach.
The paper is rather determining the features for a user than to classify the levels of influence which can be imparted by a user. Classification has been undertaken by many past studies, which classified the users from "cool" teenagers, local opinion leaders, all the way to popular public figures. However, the scope of marketing through the usage of influential users is mostly out of

the domain of expertise of a few users, we try to focus more on the actions/behaviors which are necessary.

Problem Statement:

The aim of this project is to determine influential characteristics of users on social media. The primary implementation of this project is to train a machine learning model which, for pairs of individuals, predicts the features and characteristics of users about who is more influential. It aims to determine what is required for a user to be influential.

Dataset Description:

The dataset chosen, **Influencers in Social Networks**, comprises a standard, pairwise preference learning task. Each datapoint describes two individuals, A and B. For each person, 11 pre-computed, non-negative numeric features based on twitter activity (such as volume of interactions, number of followers, etc) are provided. The binary label represents a human judgement about which one of the two individuals is more influential. An influencer is a person who attempts to gain compliance from others or uses tactics to shape the opinions, attitudes, or behaviors of others . A label '1' means A is more influential than B. 0 means B is more influential than A. Related observations will also be obtained, analyzed and contemplated.

Motivation:

Determining influencers in twitter space can be important for finding the positive and negative impact of these influencers in the social media world. Influencers analysis can also help companies in something called ‘influencer marketing’. Influencers can help a particular brand filter through that social media noise and get the product in front of the right people. Recently, Jan Koum, co-founder of whatsapp tweeted ‘#deletefacebook’. This trend was followed by a huge number of Koum’s followers. This indicates that an influential user has a huge impact on people, may it be in a positive or negative way. This project topic was undertaken for the same reason so as to find the influencers who may or may not be famous personalities and find the factors that affect in deciding if the user is influential or not.

Review of other researches:

Traditional communication theory states that a minority of users, called influentials, excel in persuading others (Rogers 1962), targeting these influentials can be successful to start a chain-reaction of influence. A more modern view, in contrast, de-emphasizes the role of influentials, that the key factors determining influence are interpersonal relationship among ordinary users and readiness of a society to adopt an innovation (Watts and Dodds 2007; Domingos and Richardson 2001).

Both the research lacks the empirical evidences and need a deeper analysis. These are not well-established till now and the current research mainly focuses on promoting a marketing strategy conforming to the findings of such studies.

Findings by Cha et.al.() were interesting if measures of influence are to be analyzed topic-wise. These were not concretely determining the feature importance for increasing the influence of the user. This was one of the major drawbacks of the study. Other studies got this drawback attempted and achieved to a certain level but the the processing undertaken had minor imperfections which under-stated a few features than the rest.

Open questions in the domain:

1. For achieving better performance or significantly higher accuracy value, coordinate ascent based algorithm can be applied to non-linear models.
2. The models of non-linear models like neural network are expected to perform better than the linear models due to better fitting and training. Although the results were not satisfying due to errors. The biasing errors due to human interpretation can be mitigated by applying decision trees to the dataset.
3. Logistic regression, generates best results while it is not expected as NN algorithms should be performing the best. The question arises how to make changes in the model which can deal with the linearly separable and inseparable datasets.
4. The best choice of attributes can be challenge and may be a big factor in determining the final results. Principal component analysis (PCA) can be applied to further select combinations of best attributes.
5. The model can't be applied to live twitter data and due to the reason, we cannot apply it to real-time users. This can be worked out to implement the model at times of geopolitical scenarios, terrorist situations and marketing campaigns; these being the most common and obvious applications along with others.
6. In a pursuit to improve the feature selection, we can assign every feature with weights according to its importance and then using for model.
7. The significance of network_feature 1,2,3 are questionable.

Short summary of the proposed approach:

The dataset considered is "Influencers in Social Media". The dataset contains data for 2 individuals, A and B. For each person, 11 pre-computed, non-negative numeric features based on twitter activity (such as volume of interactions, number of followers, etc) are provided. The binary label represents a human judgement about which one of the two individuals is most influential. Data preprocessing is required as there was data centralization which was making it inadequate to plot graphs for comparison of the attributes of two individuals. The data is

normalized to get better results. We subtracted the 11 features of A from 11 features of B. The data is then normalized to bring the values of attributes in the range of -1 to 1. After normalizing the data, the attributes are evaluated using descriptive statistics and scatter plots for quality. The scatter plots shows us the dependencies between the attributes. Consequently, we employed machine learning via scikit learn regularized linear regression (i.e. elastic net) to bench-mark classify users based on features. Secondly, a machine learning model is trained using logistic regression and SVM. This created robust dependencies and efficiently predicted the influencers among the two. Applying logistic regression is useful as the independent variables don't have to be normally distributed, or have equal variance in each group. Support Vector Machines is selected as it has a regularisation parameter that avoids over-fitting. Secondly, it uses the kernel, so building in expert knowledge about the problem via engineering the kernel is easy. Lastly, it is an approximation to a bound on the test error rate, which proves to be a good factor. We then applied LSTM on the dataset as LSTM generalizes well - even if the data is widely separated. LSTM works well over a broad range of parameters such as learning rate, input gate bias and output gate bias. LSTM can efficiently handle noise and continuous values. The results of LSTM and SVM are compared with the dataset results and they show that the human judgment about the 'Choice' is close to that predicted by the algorithms.

Backgrounds:

In the article by Cha et.al.[4], the dynamics of user influence across topics has been investigated through three measures of influence: indegree, retweets and mentions. The findings showed that a user with high indegree is not always influential and that influence is not gained spontaneously. It found that the measures were mostly correlated but indegree wasn't related to other measures. Additionally, the number of retweets and mentions were considered directly without normalizing. Other features were not as important. The topic-wise distribution of feature importance was done by calculating the number of retweets and mentions for a given topic without taking indegree in consideration due to its invariability. Three levels of influentials were identified and compared. But this approach was not considering the many other features which contribute to a user's influence. Also, for many calculations and comparisons, direct number was used without normalization or standardization which result in increased effect of a feature.

Another paper by Liu et.al.[2], the prediction of a more influential user among two given presons is done. Logistic Regression, SVM, Naive Bayes and Neural Network have been implemented and compared to predict the influence. Data pre-processing has been done using Hypothesis function to create linear decision boundary. Only 11 features were independent which were taken for the analysis. They also used k-means, feature selection and cross-validation as auxiliary techniques. This pre-processed data was used for normalization and then was fed to Logistic regression and SVM. Finally for Neural network, the MATLAB neural network pattern recognition tool box was used. Forward search was used for feature selection algorithm to rank

the features from most relevant to least. Cross-validation was used on SVM and Logistic regression in hope to get better results. For Naive Bayes method, logarithm processing was used for improving the test results. But the non-linear models were not performing significantly better than linear models.

Traditional theories stated minority of users (opinion leaders in (Katz and Lazarsfeld 1955), innovators in (Rogers 1962), and mavens in(Gladwell 2002)) being influential enough to spread ideas. Modern theories said people make choices based on the opinions of their peers and friends, and probability of acceptance to new innovation (Domingos and Richardson 2001).

The paper by Arora et.al. had four different ML algorithms: Logistic Regression, (SVM), Neural Networks and Gradient Boosting, to model the data. The data preprocessing was similar to Liu et.al.[2]. But the NEural network didn't perform to the expected accuracy level.

Methods:

The methods used are summarized below:

Logistic Regression:

Logistic regression is a predictive analysis that should be performed when the dependent variable is binary. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more independent variables.

Logistic regression generates the coefficients of a formula to predict a *logit transformation* of the probability of presence of the characteristic of interest:

$$logit(p) = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \ldots + b_kX_k$$

where p is the probability of presence of the characteristic of interest. The logit transformation is defined as the logged odds:

$$odds = \frac{p}{1-p} = \frac{probability\ of\ presence\ of\ characteristic}{probability\ of\ absence\ of\ characteristic}$$

and

$$logit(p) = \ln\left(\frac{p}{1-p}\right)$$

Advantages of Logistic Regression:

1. It is more robust: the independent variables don't have to be normally distributed, or have equal variance in each group
2. It does not assume a linear relationship between the IV and DV
3. It handles nonlinear effects
4. The DV need not be normally distributed.

Support Vector Machine:

A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. Given labelled training data, SVM generates a hyperplane that categorizes new examples. SVM supports both regression and classification tasks and can handle multiple continuous and categorical variables. Given a set of training examples, each marked as belonging to one or the other of two class, an SVM training algorithm builds a model that assigns new examples to one category or the other.

Advantages of SVM:

1. It uses a regularisation parameter, which avoids over-fitting.
2. It uses the kernel, so building in expert knowledge about the problem via engineering the kernel is easy.
3. SVM is an approximation to a bound on the test error rate, and it can be considered as one important factor.

Long Short Term Memory:

Long short-term memory (LSTM) units are a building blocks for layers of a recurrent neural network (RNN). A RNN composed of LSTM units is often called an LSTM network.

Any LSTM unit is composed of a **cell**, an **input gate**, an **output gate** and a **forget gate**. The cell is responsible for "remembering" values over various time intervals. LSTMs have a chain like structure but instead of having a single neural network layer like RNN, it has four.

Cell State: The cell state runs straight down the entire chain, with only some minor linear interactions.

Input gate: Used to feed input to the neural network

Forget gate: This gate helps in forgetting the values/attributes that are least significant.

Output gate: This give us the output of the neural network

Advantages of LSTM:

1. LSTM generalizes well - even if the data is widely separated.

2. LSTM works well over a broad range of parameters such as learning rate, input gate bias and output gate bias
3. LSTM can efficiently handle noise and continuous values

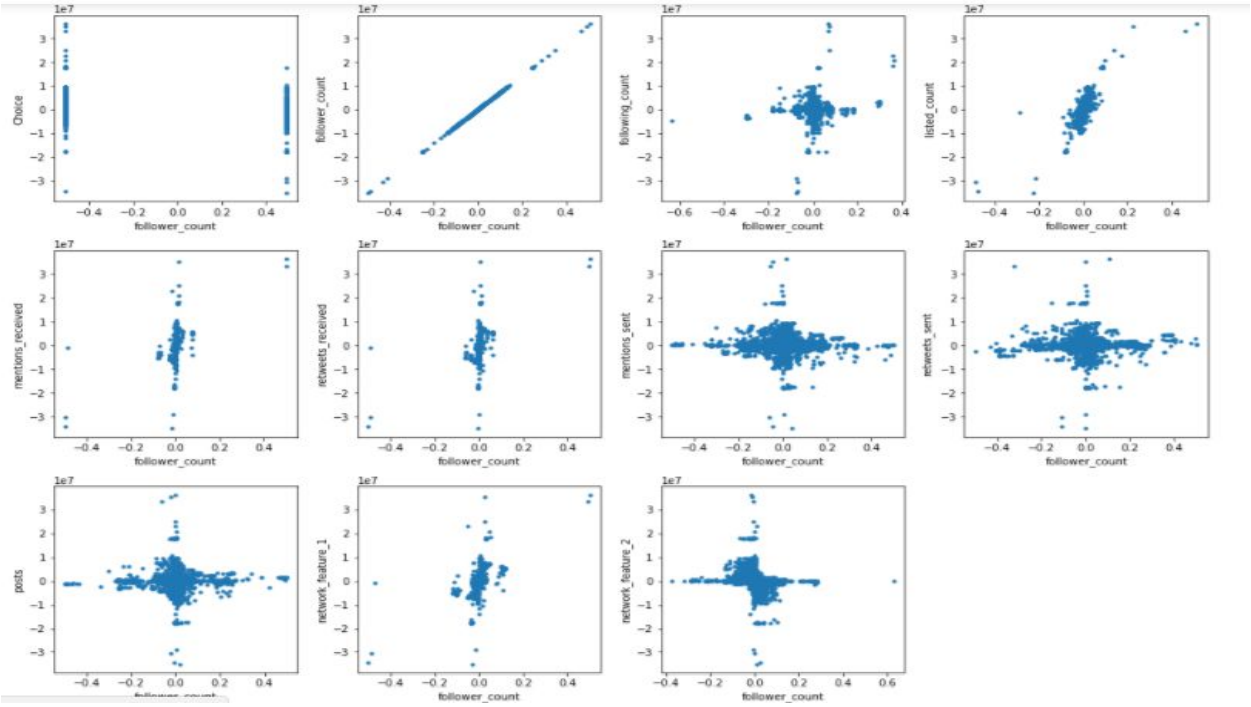
Experiments:

The methods which are started above have showed the following results on various features.The features which were found to be important considering the original dataset were

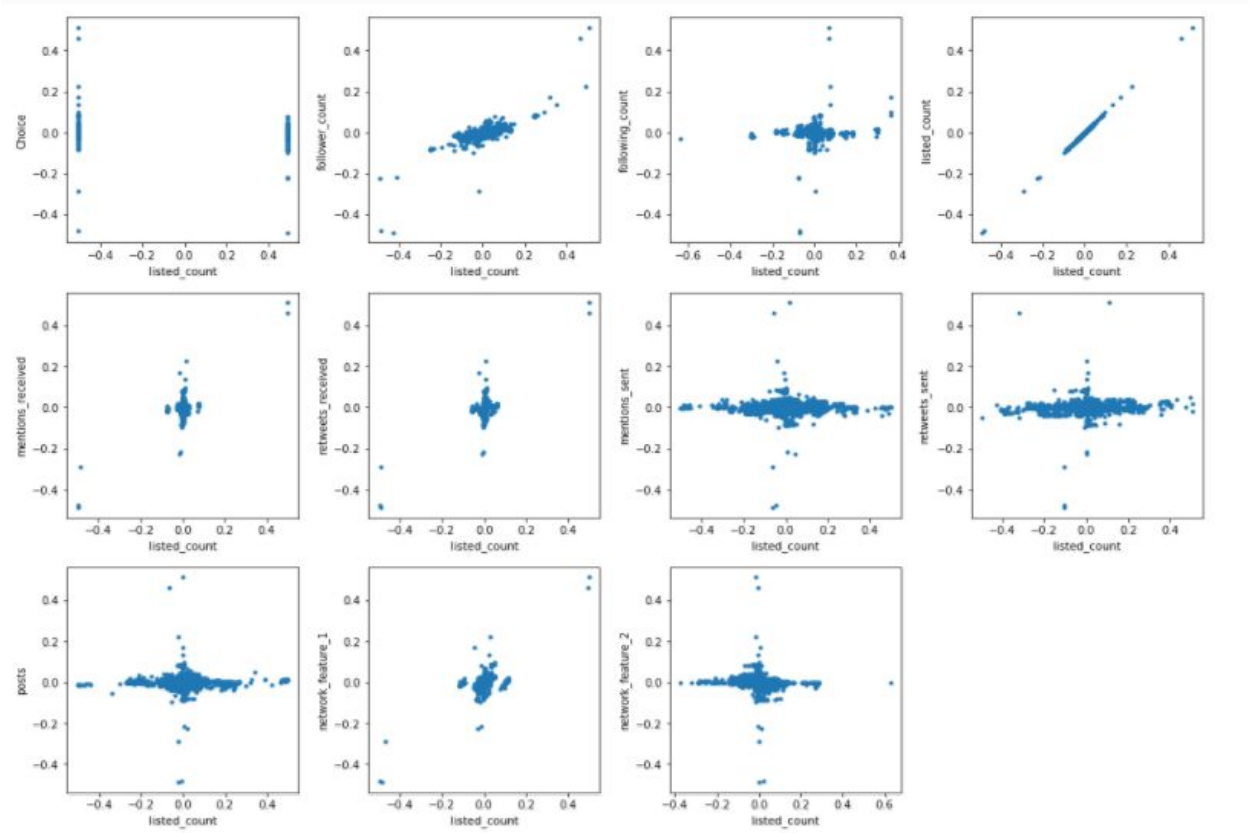
1. Listed_count
2. Follower_count
3. Mentions_sent
4. Retweets_sent
5. Posts

On the basis of these,first with the help of scatter plots we tried to analyse these feature sets in comparison to all the other features so that a pattern could be obtained.

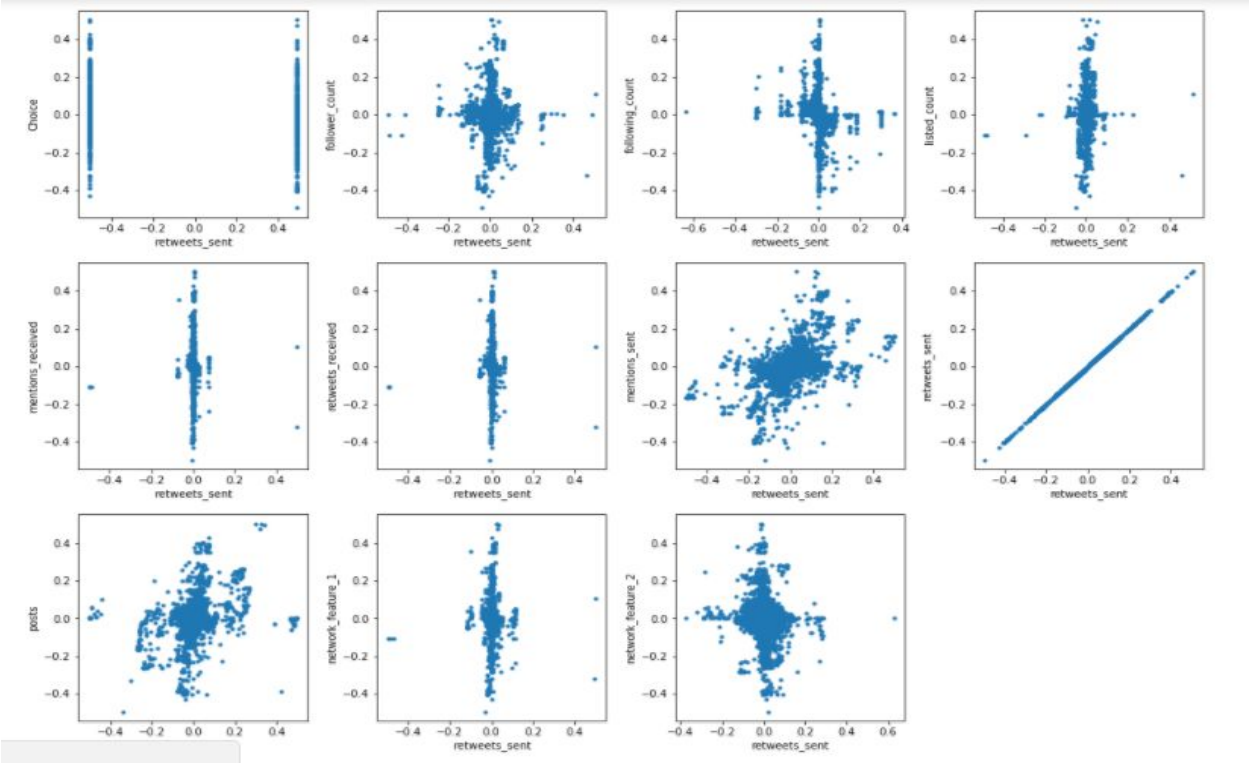
The first scatter plot obtained was with follower count in comparison with all the other features:



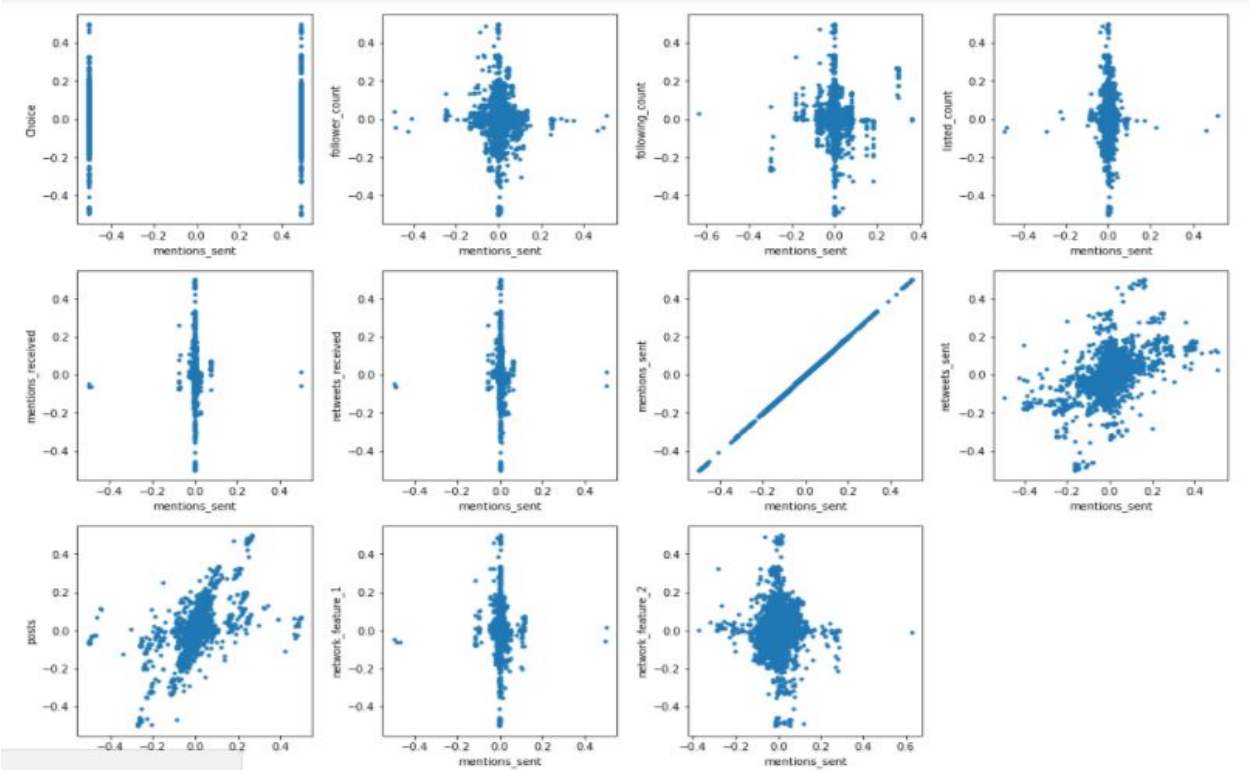
The first scatter plot obtained was with listed count in comparison with all the other features:



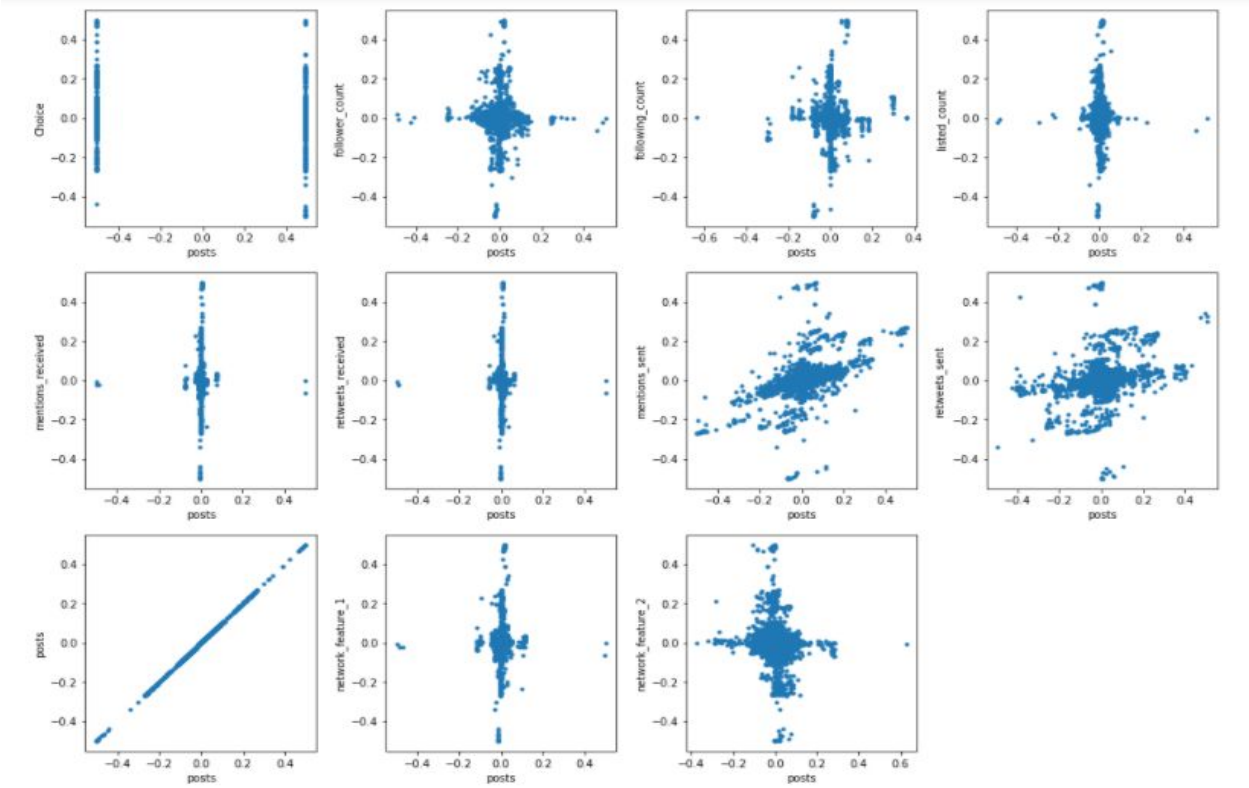
The first scatter plot obtained was with retweets sent in comparison with all the other features:



The first scatter plot obtained was with mentions sent in comparison with all the other features:

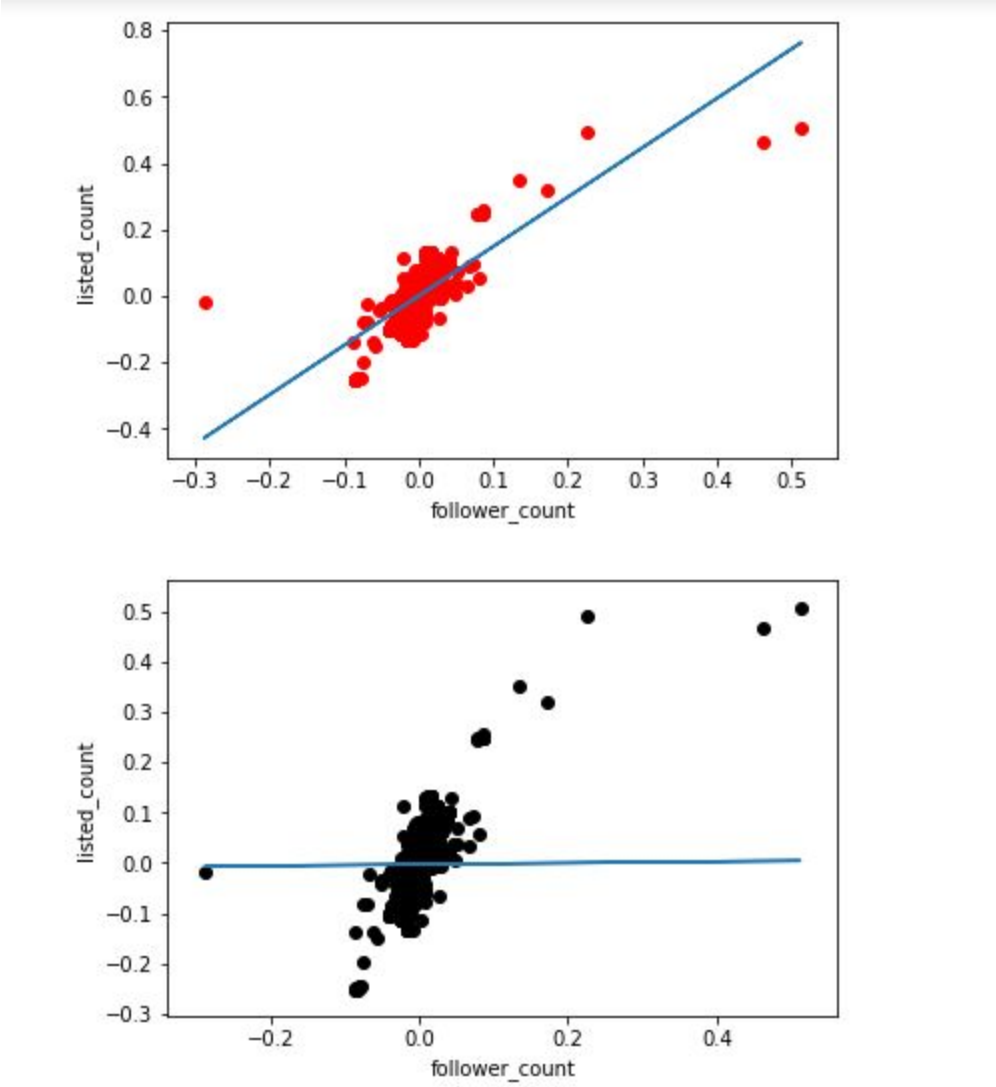


The first scatter plot obtained was with posts in comparison with all the other features:



Observing the above scatter plots,linear regression and least mean squares were applied to these features so that the dependency of features could be understood.From the below plots we understood that the most important features in the dataset were listed count and follower count.The 5 features mentioned above did give us a good importance of features,but the best turned out were listed and follower count.The following plots were observed from the below regression plots:

Linear Regression in the first plot and Least Mean Square in the second.

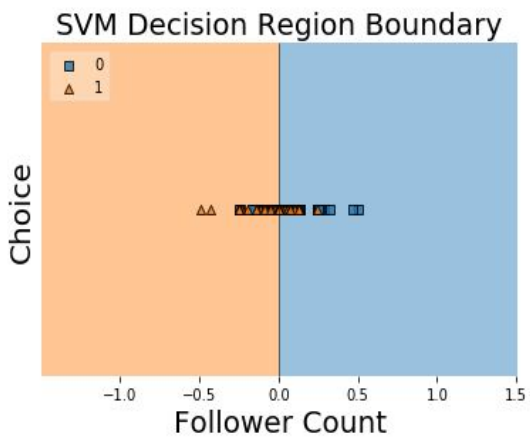


To understand and confirm the best features as listed count and follower count we tried Support Vector Machine and Logistic Regression with 10 fold cross validation to check what accuracies were obtained for classifying the best feature set.

The accuracies obtained for various feature sets are as follows:

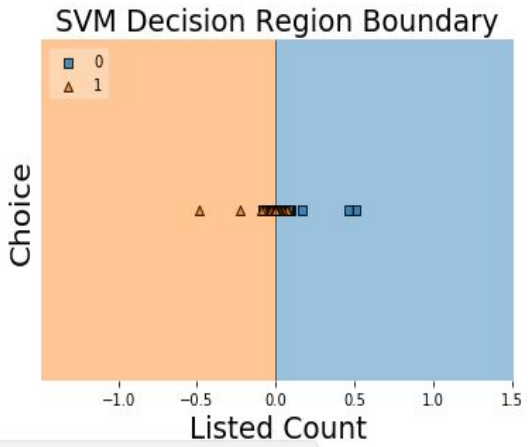
SVM start!
CNN-SVM Accuracy: 62.8%

Text(0.5,1,'SVM Decision Region Boundary')



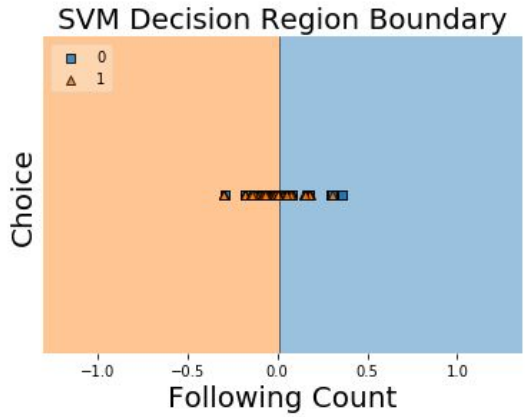
SVM start!
CNN-SVM Accuracy: 70.8%

Text(0.5,1,'SVM Decision Region Boundary')



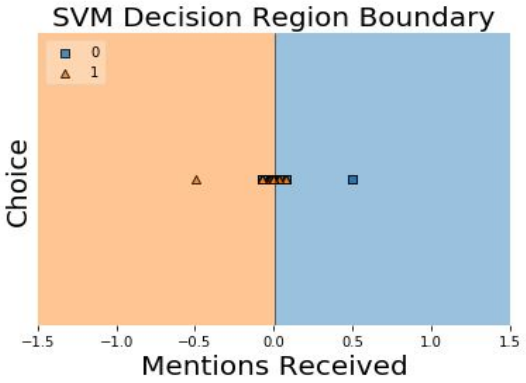
SVM start!
CNN-SVM Accuracy: 51.7%

Text(0.5,1,'SVM Decision Region Boundary')



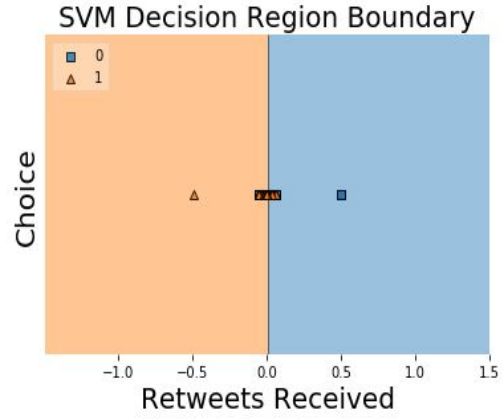
CNN-SVM Accuracy: 52.6%

Text(0.5,1,'SVM Decision Region Boundary')



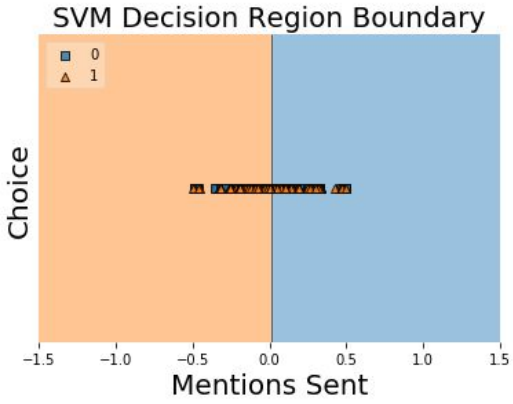
SVM start!
CNN-SVM Accuracy: 53.4%

Text(0.5,1,'SVM Decision Region Boundary')

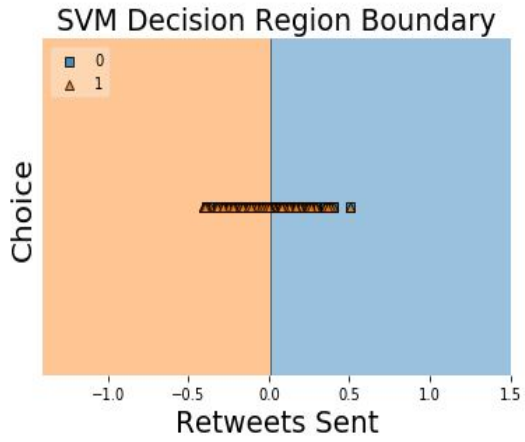


SVM start!
CNN-SVM Accuracy: 64.7%

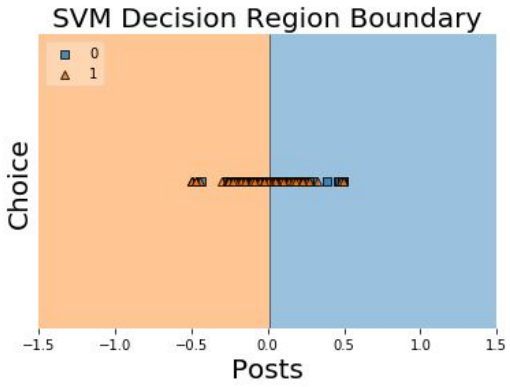
Text(0.5,1,'SVM Decision Region Boundary')



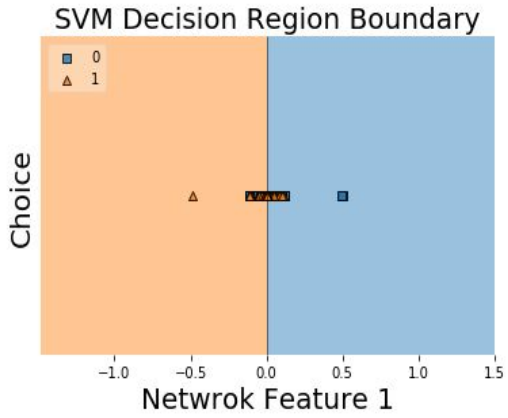
```
SVM start!
CNN-SVM Accuracy: 62.0%

Text(0.5,1,'SVM Decision Region Boundary')
```



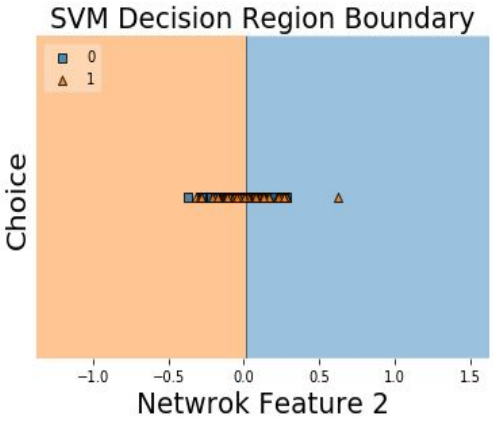

```
SVM start!
CNN-SVM Accuracy: 62.5%

Text(0.5,1,'SVM Decision Region Boundary')
```




```
SVM start!
CNN-SVM Accuracy: 55.4%

Text(0.5,1,'SVM Decision Region Boundary')
```




```
SVM start!
CNN-SVM Accuracy: 53.1%

Text(0.5,1,'SVM Decision Region Boundary')
```



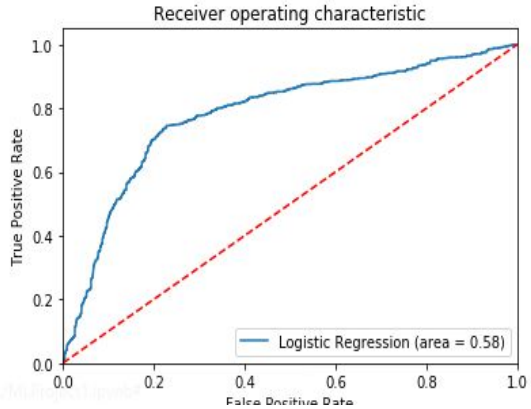
With the help of Support Vector Machines we came up with the conclusion that the most accurately classified feature was Listed Count and with that list was followed by follower count,mentions sent,retweets sent and posts.

Further Logistic Regression and 10 fold cross validations were taken to obtain the accuracy whose results are as follows:

Follower count

```
Accuracy using 10 cross fold cross validation: 0.587

Confusion Matrix
[[ 251  841]
 [  83 1025]]

Classification Report
             precision    recall  f1-score   support

          0       0.75      0.23      0.35      1092
          1       0.55      0.93      0.69      1108

avg / total       0.65      0.58      0.52      2200
```

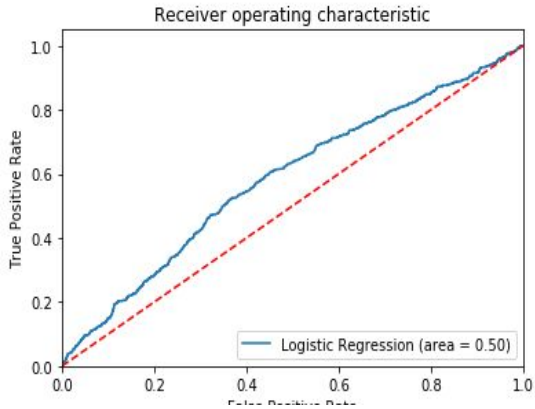
ROC



Following Count

```
Accuracy using 10 cross fold cross validation: 0.521

Confusion Matrix
[[  67 1025]
 [  59 1049]]

Classification Report
             precision    recall  f1-score   support

          0       0.53      0.06      0.11      1092
          1       0.51      0.95      0.66      1108

avg / total       0.52      0.51      0.39      2200
```

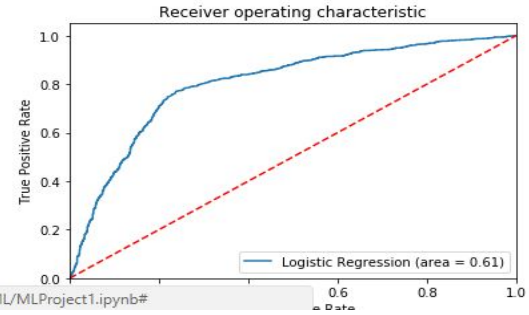
ROC



Listed Count

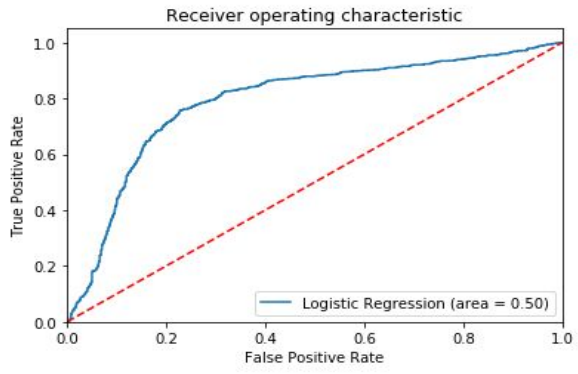
```
Accuracy using 10 cross fold cross validation: 0.614

Confusion Matrix
[[ 286  806]
 [  56 1052]]

Classification Report
             precision    recall  f1-score   support

          0       0.84      0.26      0.40      1092
          1       0.57      0.95      0.71      1108

avg / total       0.70      0.61      0.56      2200
```

ROC



Mentions Received

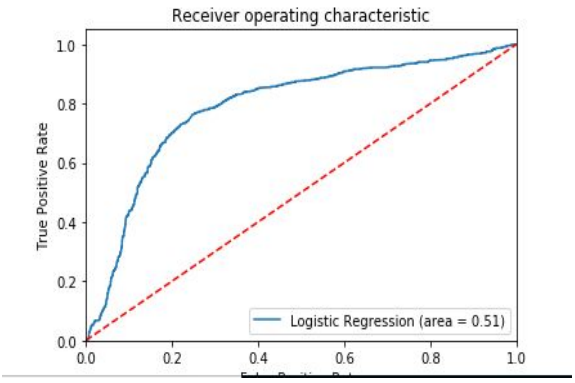
```
10-fold cross validation average accuracy: 0.516
[[   8 1084]
 [   2 1106]]
             precision    recall  f1-score   support

          0       0.80      0.01      0.01      1092
          1       0.51      1.00      0.67      1108

avg / total       0.65      0.51      0.35      2200
```



Retweets Received

```
Accuracy using 10 cross fold cross validation: 0.518

Confusion Matrix
[[  15 1077]
 [   4 1104]]

Classification Report
             precision    recall  f1-score   support

          0       0.79      0.01      0.03      1092
          1       0.51      1.00      0.67      1108

avg / total       0.65      0.51      0.35      2200
```

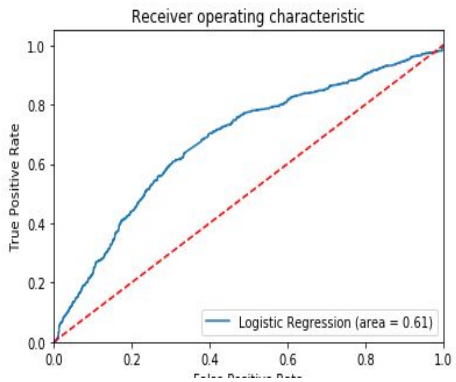
ROC



Mentions Sent

```
Accuracy using 10 cross fold cross validation: 0.625

Confusion Matrix
[[472 620]
 [229 879]]

Classification Report
             precision    recall  f1-score   support

          0       0.67      0.43      0.53      1092
          1       0.59      0.79      0.67      1108

avg / total       0.63      0.61      0.60      2200
```

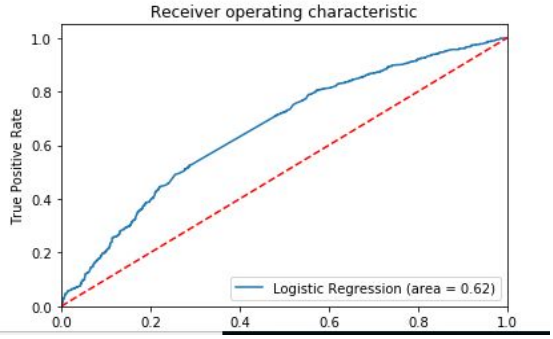
ROC



Retweets Sent

```
Accuracy using 10 cross fold cross validation: 0.608

Confusion Matrix
[[467 625]
 [218 890]]

Classification Report
             precision    recall  f1-score   support

          0       0.68      0.43      0.53      1092
          1       0.59      0.80      0.68      1108

avg / total       0.63      0.62      0.60      2200
```

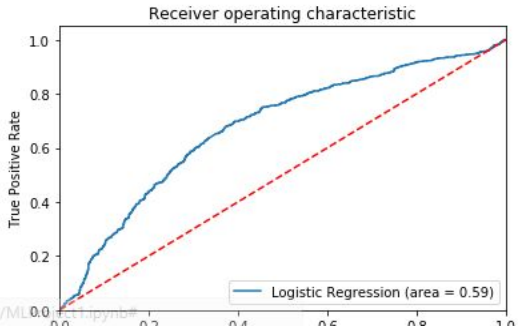
ROC



Posts

```
Accuracy using 10 cross fold cross validation: 0.598

Confusion Matrix
[[346 746]
 [161 947]]

Classification Report
             precision    recall  f1-score   support

          0       0.68      0.32      0.43      1092
          1       0.56      0.85      0.68      1108

avg / total       0.62      0.59      0.56      2200
```

ROC



From the above observations the best features which were obtained were mentions_sent which is followed by listed count,retweets sent,posts and follower count.

Finally,to get the best features,neural networks were also applied to the dataset to come up with the best accuracies of these features.
The accuracies with LSTM(Long Short Term Memory) are as follows:

Follower Count

```
Epoch 92/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2217 - acc: 0.6345
Epoch 93/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2215 - acc: 0.6391
Epoch 94/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2214 - acc: 0.6433A: 1s - l
Epoch 95/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2213 - acc: 0.6439
Epoch 96/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2213 - acc: 0.6424
Epoch 97/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2216 - acc: 0.6421
Epoch 98/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2217 - acc: 0.6403
Epoch 99/100
3300/3300 [==============================] - 5s 2ms/step - loss: 0.2216 - acc: 0.6403
Epoch 100/100
3300/3300 [==============================] - 5s 2ms/step - loss: 0.2216 - acc: 0.6415A: 0s - loss: 0.2210 - acc: 0.6
Accuracy: 63.27%
```

Following Count

```
Epoch 92/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5173
Epoch 93/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2495 - acc: 0.5170
Epoch 94/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5167
Epoch 95/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2495 - acc: 0.5152
Epoch 96/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5145
Epoch 97/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5152
Epoch 98/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5167
Epoch 99/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5170
Epoch 100/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2495 - acc: 0.5185
Accuracy: 52.73%
```

Listed Count

```
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2250 - acc: 0.6267
Epoch 93/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2253 - acc: 0.6252
Epoch 94/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2256 - acc: 0.6197
Epoch 95/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2269 - acc: 0.6233
Epoch 96/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2274 - acc: 0.6130
Epoch 97/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2267 - acc: 0.6173
Epoch 98/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2262 - acc: 0.6209
Epoch 99/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2264 - acc: 0.6209
Epoch 100/100
3300/3300 [==============================] - 8s 2ms/step - loss: 0.2259 - acc: 0.6264
Accuracy: 61.86%
```

Mentions Received

```
Epoch 92/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2248 - acc: 0.6264
Epoch 93/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2247 - acc: 0.6291
Epoch 94/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2247 - acc: 0.6315
Epoch 95/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2245 - acc: 0.6318
Epoch 96/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2245 - acc: 0.6361
Epoch 97/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2246 - acc: 0.6364
Epoch 98/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2244 - acc: 0.6336A
Epoch 99/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2245 - acc: 0.6391
Epoch 100/100
3300/3300 [==============================] - 5s 2ms/step - loss: 0.2241 - acc: 0.6327
Accuracy: 61.09%
```

Retweets Received

```
Epoch 92/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2275 - acc: 0.6248
Epoch 93/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2272 - acc: 0.6300
Epoch 94/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2263 - acc: 0.6309
Epoch 95/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2259 - acc: 0.6309
Epoch 96/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2263 - acc: 0.6233
Epoch 97/100
3300/3300 [==============================] - 5s 2ms/step - loss: 0.2266 - acc: 0.6252
Epoch 98/100
3300/3300 [==============================] - 5s 2ms/step - loss: 0.2265 - acc: 0.6245
Epoch 99/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2263 - acc: 0.6218
Epoch 100/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2262 - acc: 0.6203
Accuracy: 59.18%
```

Mentions sent

```
Epoch 92/100
3300/3300 [==============================] - 8s 2ms/step - loss: 0.2243 - acc: 0.6324
Epoch 93/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2240 - acc: 0.6376
Epoch 94/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2241 - acc: 0.6285
Epoch 95/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2227 - acc: 0.6409
Epoch 96/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2218 - acc: 0.6452
Epoch 97/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2214 - acc: 0.6436
Epoch 98/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2211 - acc: 0.6461
Epoch 99/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2205 - acc: 0.6427
Epoch 100/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2205 - acc: 0.6455
Accuracy: 63.73%
```

Retweets Sent

```
Epoch 92/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2235 - acc: 0.6376
Epoch 93/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2236 - acc: 0.6379
Epoch 94/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2236 - acc: 0.6394
Epoch 95/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2237 - acc: 0.6394
Epoch 96/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2239 - acc: 0.6379
Epoch 97/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2241 - acc: 0.6421
Epoch 98/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2241 - acc: 0.6415
Epoch 99/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2243 - acc: 0.6373
Epoch 100/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2245 - acc: 0.6348
Accuracy: 63.18%
```

Posts

```
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2252 - acc: 0.6364
Epoch 90/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2250 - acc: 0.6370
Epoch 91/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2249 - acc: 0.6367
Epoch 92/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2248 - acc: 0.6364
Epoch 93/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2247 - acc: 0.6355
Epoch 94/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2246 - acc: 0.6370
Epoch 95/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2246 - acc: 0.6376
Epoch 96/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2251 - acc: 0.6352A: 0s - loss: 0.225
Epoch 97/100
3300/3300 [==============================] - 8s 2ms/step - loss: 0.2256 - acc: 0.6355
Epoch 98/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2257 - acc: 0.6336
Epoch 99/100
3300/3300 [==============================] - 7s 2ms/step - loss: 0.2256 - acc: 0.6330
Epoch 100/100
3300/3300 [==============================] - 6s 2ms/step - loss: 0.2257 - acc: 0.6321
Accuracy: 61.73%
```

The best feature which was obtained by LSTM was mentions sent followed by followers count,retweets sent,posts and listed count.

After experimenting on the features and lising down various accuracies we have come up with conclusion that Support Vector Machine has been the best among all the three models as it has given the best accuracy.

Conclusion:

We have achieved to get the best features as Listed Count and Follower count through are models as per stated in the feature selection.As stated in paper [2] the feature selection and the features obtained through the models did not match.The accuricies given by them are high as compared to our model but the paper could not come up as per the plan.We have achieved to get the features which were thought to be important.

The features are given these numbers to understand the accuracies in the ascending format of the models.

1 follower_count
2 following_count
3 listed_count
4 mentions_received
5 retweets_received
6 mentions_Sent
7 retweets_sent
8 posts

9 network_feature_1
10 network_feature-2
11 network_feature_3

According to Support Vector MAchine,the highest to lowest accurate features are
3 1 6 8 7 11 9 5 10 4 2
According to Logistic Regression,the highest to lowest accurate features are
6 3 7 8 1 11 9 10 5 4 2
According to LSTM,the highest to lowest accurate features are
6 1 8 3 4 5 7 2 9 11 10

We conclude with Mentions sent,Listed Count and Follower Count to be the most important features to determine which users are influential.And therefore,when we use these and apply it on the dataset we will get the user which be the highest influential user.

References:

1. https://www.statisticssolutions.com/what-is-logistic-regression/

2. http://cs229.stanford.edu/proj2014/Ruishan%20Liu,%20Yang%20Zhao,%20Liuyu%20Zhou,%20Predict%20Influencers%20in%20the%20Social%20Network.pdf

3. http://colah.github.io/posts/2015-08-Understanding-LSTMs/

4. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/viewFile/1538/1826,2011

5. http://nbviewer.jupyter.org/url/webpages.uncc.edu/mlee173/teach/itcs6156/notebooks/notes/Note-Support%20Vector%20Machines.ipynb

6. http://scikit-learn.org/stable/

7. https://www.medcalc.org/manual/logistic_regression.php