
VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

# School of Computing Science and Engineering

VIT Chennai

Vandalur - Kelambakkam Road, Chennai - 600 127

**Review III Report**

**Programme:** **NoSQL Database**

**Course:** **CSE6006-NoSQL Database**

**Slot:** **E1 Slot**

**Faculty:** **Dr. A. Bhuvaneswari**

**Component:** **J**

**Title:** **Fake Profile Detection**

**Team Member(s):**

**Sunil Sai Kumar Mada (20MCB1013)**

**Bhargavinath Dornadula (20MCB1011)**

**Renuga keerthi (20MCB1017)**

# Abstract:

In the present generation, the social life of everyone has become associated with online social networks. These sites have made a drastic change in the way we pursue our social life. Making friends and keeping in contact with them and their updates has become easier. But with their rapid growth, many problems like fake profiles, online impersonation have also grown.There are no feasible solutions exist to control these problems. In this paper, I came up with a framework with which the automatic identification of fake profiles is possible and is efficient. This framework uses classification techniques like Random Forest Classifier to classify the profiles into fake or genuine classes. As this is an automatic detection method, it can be applied easily by online social networks that have millions of profiles whose profiles cannot be examined manually.

# 1. Introduction:

Social networking site is a website where each user has a profile and can keep in contact with friends, share their updates and meet new people who have the same interests. These Online Social Networks (OSN) use web2.0 technology, which allows users to interact with each other. Social networking sites are growing rapidly and changing the way people keep in contact with each other. The online communities bring people with the same interests together which makes it easier for users to make new friends. In the present generation, the social life of everyone has become associated with online social networks. Adding new friends and keeping in contact with them and their updates has become easier.

# 2. Brief Intro about project:

### a. Project Objectives

In today's online social networks there have been a lot of problems like fake profiles, online impersonation, etc. In this project, we intend to give a framework which detects fake profiles in online social media networks. This project can be done so that the social life of people becomes secure and by using this technique we can make it easier for the sites to manage the huge number of profiles, which can't be done manually.

### b. Literature Survey:

For instance, Nazir et al. (2010) [1] describes recognizing and describing phantom profiles in online social gaming applications.The article analyses a Facebook application, the online game "Fighters club", known to provide incentives and gaming advantage to those users who invite their peers into the game. The authors contend that by giving such impetuses the game motivates its players to make fake profiles. By presenting those fake profiles into the game, the user would increase a motivating force of an incentive for him/herself.

Adikari and Dutta (2014) [2] depict recognizable proof of fake profiles on LinkedIn. The paper demonstrates that fake profiles can be recognized with 84% exactness and 2.44% false negative, utilizing constrained profile information as input. Techniques, for example, neural networks, SVMs, and Principal component analysis are applied. Among others, highlights, for example, the number of languages spoken, training, abilities, suggestions, interests, and awards are utilized. Qualities of profiles, known to be fake, posted on uncommon sites are utilized as a ground truth.

Chu et al. (2010) [3] go for separating Twitter accounts operated by humans, bots, or cyborgs (i.e.,bots and people working in concert). As a part of the detection problem formulation, the

Identification of spamming records is acknowledged with the assistance of an Orthogonal Sparse Bigram (OSB) text classifier that uses pairs of words as features.

## 3. Data Set Description:

- The dataset includes user profiles and been divided into 2 files : Users and Fake users
- There are 1338 fake users and 1482 normal users in total.

Attributes of dataset:

```
"id","name","screen_name","statuses_count","followers_count","friends_count","favourites_co
unt","listed_count","created_at","url","lang","time_zone","location","default_profile","def
ault_profile_image","geo_enabled","profile_image_url","profile_banner_url","profile_use_bac
kground_image","profile_background_image_url_https","profile_text_color","profile_image_url
_https","profile_sidebar_border_color","profile_background_tile","profile_sidebar_fill_colo
r","profile_background_image_url","profile_background_color","profile_link_color","utc_offs
et","protected","verified","description","updated","dataset"
```

Twitter Dataset:

- Collecting data from twitter using tweepy model.
- We can get data from twitter app.devloper, where we have to sign in and make an account.

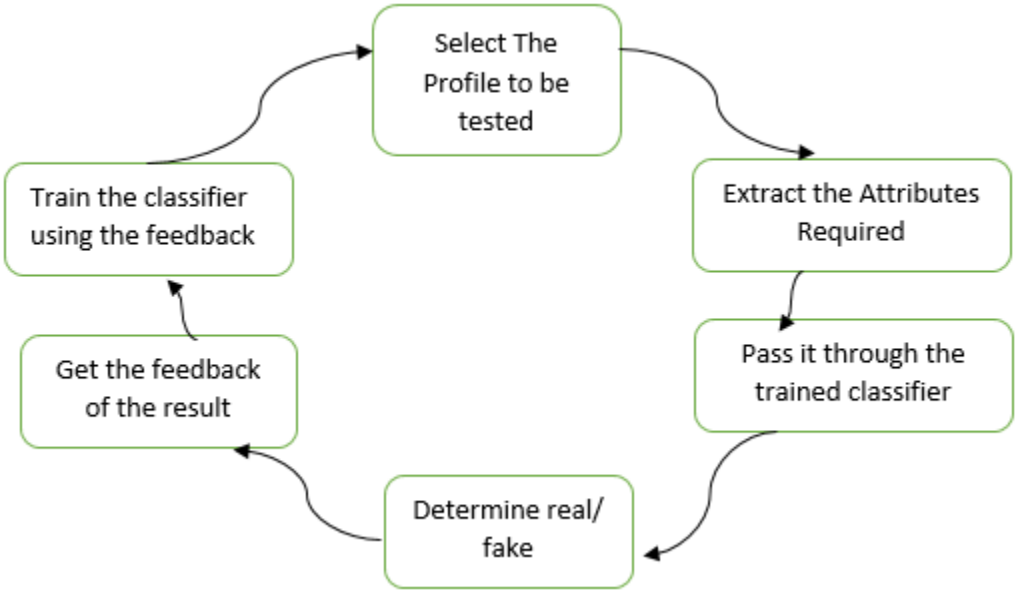
Attributes of dataset:

```
[ ]  1 df.columns
```

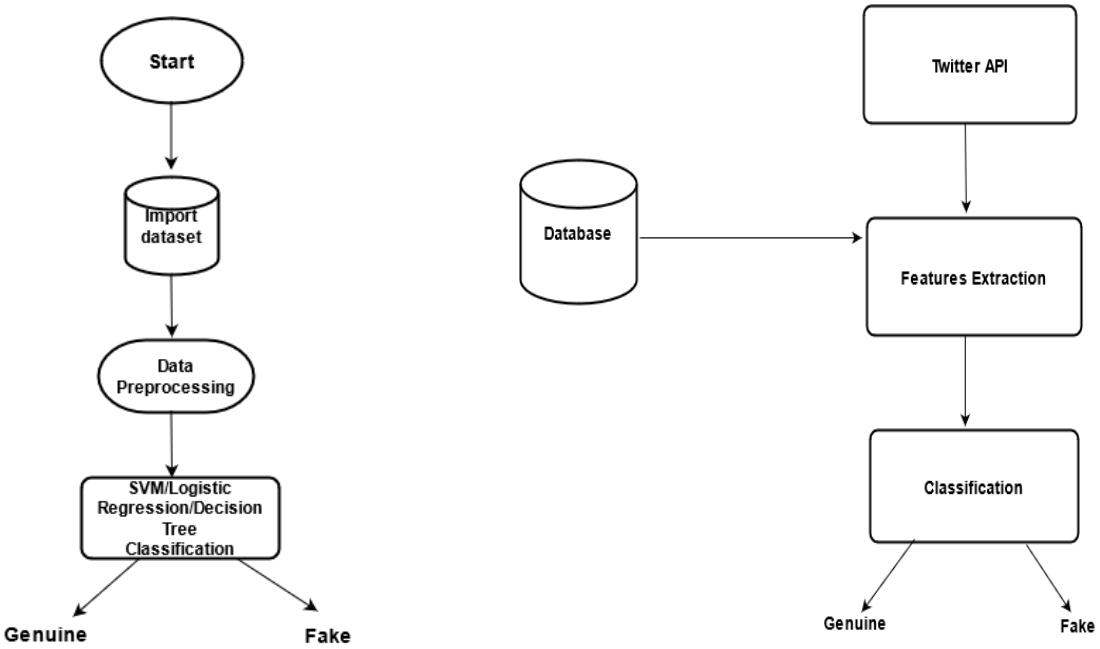
```
Index(['TwittID', 'TextData', 'TweetCreatedAt', 'RetweetCount',
       'TweetFavouriteCount', 'TweetSource', 'UserID', 'UserScreenName',
       'UserName', 'UserCreatedAt', 'UserDescription', 'UserDescriptionLength',
       'UserFollowersCount', 'UserFriendsCount', 'UserLocation', 'HttpCount',
       'HashtagCount', 'MentionCount', 'TweetCount', 'AvgHashtag',
       'AvgURLCount', 'AvgMention', 'AvgRetweet', 'AvgFavCount',
       'SpammerOrNot'],
      dtype='object')
```

## 4. Methodology and Algorithm used:

### a. System Architecture diagram




### b. Flowchart



## 5. Experimental setup

Detection of fake profiles in online social networks :

- ➔ The project shows that fake profiles can be detected using limited profile data as input.
- ➔ Methods such as Decision tree classifier, Support vector machine and random forest classifier are applied.

Storing and Retrieving of Database :

- ➔ MongoDB has been used for storing data.

```
1 from pymongo import MongoClient
2 client = MongoClient()
3 db = client.Project
```

```
[ ]  1 collection = db.Users
     2 OriginalData = pd.DataFrame(list(collection.find()))
```

```
[ ]  1 collection = db.Fusers
     2 FakeData = pd.DataFrame(list(collection.find()))
```

```
[ ]  1 OriginalData.drop("_id",axis=1,inplace=True)
```

Fake profile detection using twitter dataset:

- ➔ Collecting data from twitter using tweepy model.

**Tweepy model :**
Twitter is a popular social network where users share messages called tweets. Twitter allows us to mine the data of any user using Twitter API or Tweepy. The data will be tweets extracted from the user. The first thing to do is get the consumer key, consumer secret, access key and access secret from twitter developer available easily for each user. These keys will help the API for authentication**.**

- ➔ For that we need Consumer key, Consumer secret key, Access token, Access token.

- ➔ We can get that data from twitter app.devloper where we have to sign in and make an account and then twitter generates the keys.

- ➔ Printing all the friends names of the user : For instance, We try printing the friend names of the particular user. Here, in this case we have considered example of Rajinikanth and

have stored the list of names in a file "**Fri_1**"

```
Name of the Friends of user
anirudhofficial
arrahman
dhanushkraja
soundaryaarajni
ash_r_dhanush
SrBachchan
PMOIndia
narendramodi
firstpost
bsindia
airnewsalerts
EconomicTimes
ABPNews
BBCHindi
aajtak
ZeeNewsEnglish
TimesNow
htTweets
IndiaToday
IndianExpress
ndtv
the_hindu
timesofindia
CNNnews18
```

➔ As a result, we use logistic regression model and the spam user profiles are detected

## 6. Results and Discussion

Detect fake profiles in online social networks:

| **Models** | **Accuracy** |
|---|---|
| Support Vector Machine(SVM) | 96.9 |
| Logistic Regression | 99.4 |
| Decision Tree | 99.4 |

All the three models show good accuracy.

Detect fake profiles from twitter:

| **Model** | **Accuracy** |
|---|---|
| Logistic Regression | 95.3 |

## 7. Screenshots of the project

Machine learning models:

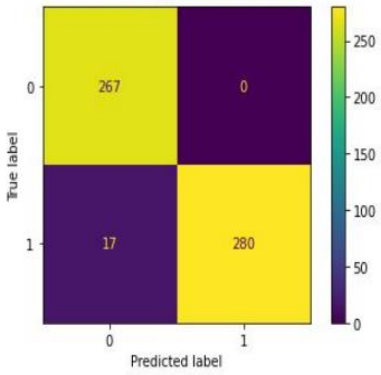
1.Support Vector Machine(SVM):

```
[ ]  1 accuracy_score(y_test, y_pred)
```

```
0.9698581560283688
```

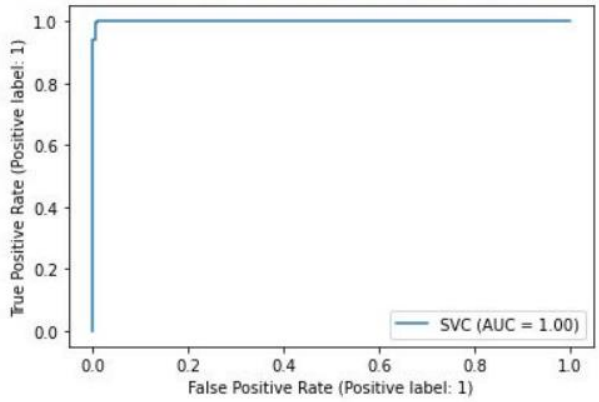
```
1 plot_confusion_matrix(model, x_test, y_test)
```

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7f0021e78550>
```



```
1 plot_roc_curve(model, x_test, y_test)
```

```
<sklearn.metrics._plot.roc_curve.RocCurveDisplay at 0x7f00486e3100>
```




```
1 print(classification_report(y_test, y_pred, target_names=['Fake','Genuine']))
```

```
              precision    recall  f1-score   support

        Fake       0.94      1.00      0.97       267
     Genuine       1.00      0.94      0.97       297
```

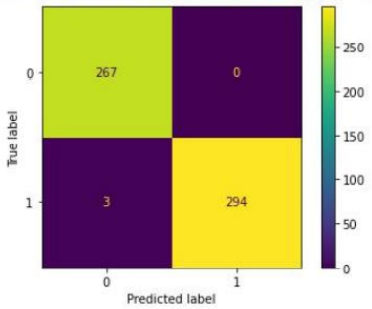
## 2. Logistic Regression:

```
[ ]  1 accuracy_score(y_test, prediction)
```

```
0.9946808510638298
```

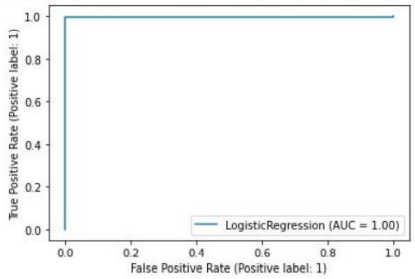
```
[ ]  1 plot_confusion_matrix(model, X_test, y_test)
```

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7f00218f3ee0>
```



```
1 plot_roc_curve(model, X_test, y_test)
```

```
<sklearn.metrics._plot.roc_curve.RocCurveDisplay at 0x7f0021901580>
```




```
1 print(classification_report(y_test, prediction, target_names=['Fake','Genuine']))
```

```
              precision    recall  f1-score   support

        Fake       0.99      1.00      0.99       267
     Genuine       1.00      0.99      0.99       297
```

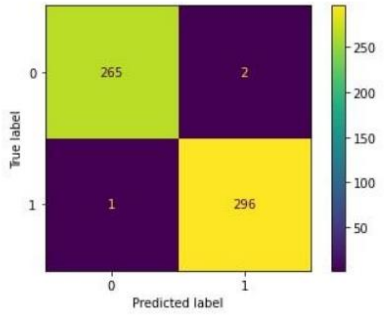
# 3. Decision Tree:

```
In [36]: accuracy_score(y_test, y_pred)
```

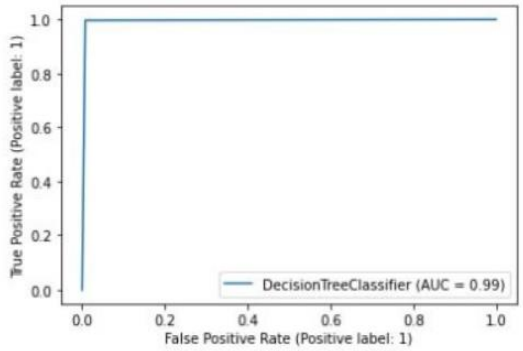
```
Out[36]: 0.9946808510638298
```

```
In [37]: plot_confusion_matrix(classifier, X_test, y_test)
```

```
Out[37]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7f6c720fbe80>
```



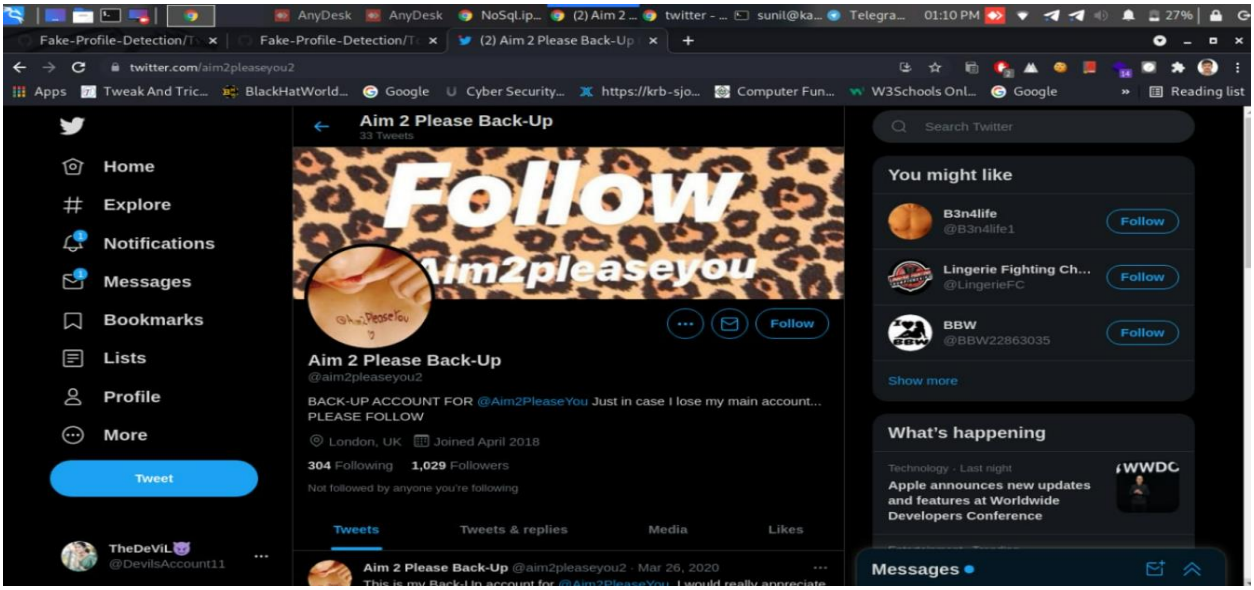
```
In [39]: plot_roc_curve(classifier, X_test, y_test)
Out[39]: <sklearn.metrics._plot.roc_curve.RocCurveDisplay at 0x7f6c7215d580>
```

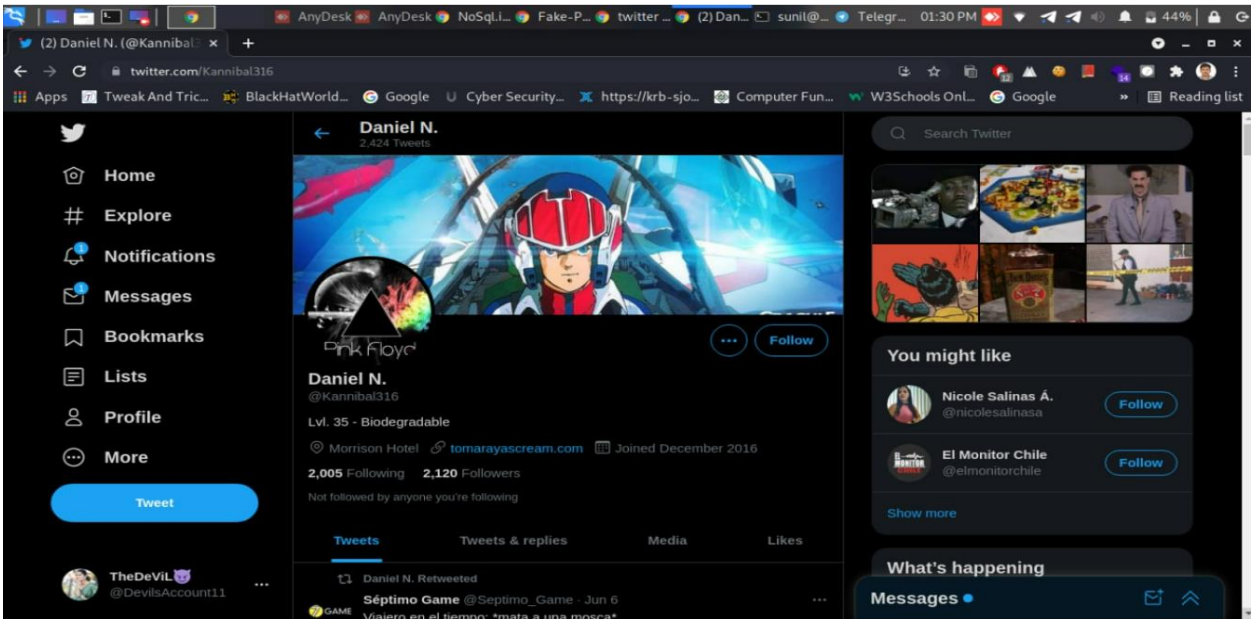



```
In [40]: print(classification_report(y_test, y_pred, target_names=['Fake','Genuine']))
              precision    recall  f1-score   support

        Fake       1.00      0.99      0.99       267
     Genuine       0.99      1.00      0.99       297
```

## Spam profiles detection:






## Genuine Users:

| | TwittID | TextData | TweetCreatedAt | RetweetCount | TweetFavouriteCount | TweetSource | UserID | UserScreenName | UserName | UserCreatedAt | UserDescription | UserDescriptionLength | UserFollowersCount | UserFriendsCount | UserLocation | HttpCount | HashtagCount | Mer |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1399625976800153600 | RT @karthiksubbaraj: Here's #JagameThandhiramT... | 2021-06-01 07:16:51 | 3227 | 0 | Twitter for iPhone | 216447259 | anirudhofficial | Anirudh Ravichander | 2010-11-16 19:13:06 | Music Composer/Singer in the Indian Film Industry | 49 | 8138319 | 207 | Chennai | 1 | 2 | |
| 1 | 1399625927089217536 | RT @dhanushkraja: What would have been a great... | 2021-06-01 07:16:39 | 7527 | 0 | Twitter for iPhone | 216447259 | anirudhofficial | Anirudh Ravichander | 2010-11-16 19:13:06 | Music Composer/Singer in the Indian Film Industry | 49 | 8138319 | 207 | Chennai | 0 | 0 | |
| 2 | 1395367947552522241 | This track and it's video are 🔥 inAwesome work... | 2021-05-20 13:16:58 | 302 | 3645 | Twitter for iPhone | 216447259 | anirudhofficial | Anirudh Ravichander | 2010-11-16 19:13:06 | Music Composer/Singer in the Indian Film Industry | 49 | 8138319 | 207 | Chennai | 1 | 1 | |
| 3 | 1395030557092769794 | #URGENT 🆘inEmergency need of 72 vials of #Amph... | 2021-05-19 14:56:17 | 925 | 2715 | Twitter for iPhone | 216447259 | anirudhofficial | Anirudh Ravichander | 2010-11-16 19:13:06 | Music Composer/Singer in the Indian Film Industry | 49 | 8138319 | 207 | Chennai | 1 | 2 | |
| 4 | 1392771872056479748 | RT @soundaryaarajni: Our Thalaivar gets his va... | 2021-05-13 09:21:05 | 4565 | 0 | Twitter for iPhone | 216447259 | anirudhofficial | Anirudh Ravichander | 2010-11-16 19:13:06 | Music Composer/Singer in the Indian Film Industry | 49 | 8138319 | 207 | Chennai | 0 | 2 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| 715 | 1402144157053853703 | RT @news18dotcom: @AmanKayamHai_ In the detail... | 2021-06-08 06:03:12 | 2 | 0 | TweetDeck | 6509832 | CNNnews18 | News18 | 2007-06-01 20:31:01 | Lightning fast alerts, #BreakingNews from Indi... | 62 | 4597047 | 403 | India | 0 | 0 | |
| 716 | 1402143957325258753 | RT @news18dotcom: @AmanKayamHai_ News18's @Ama... | 2021-06-08 06:02:24 | 3 | 0 | TweetDeck | 6509832 | CNNnews18 | News18 | 2007-06-01 20:31:01 | Lightning fast alerts, #BreakingNews from Indi... | 62 | 4597047 | 403 | India | 0 | 0 | |
| 717 | 1402142378102640643 | It is learnt that the Centre had planned to arm... | 2021-06-08 05:56:08 | 3 | 11 | Twitter Web App | 6509832 | CNNnews18 | News18 | 2007-06-01 20:31:01 | Lightning fast alerts, #BreakingNews from Indi... | 62 | 4597047 | 403 | India | 1 | 0 | |
| 718 | 1402140713614299136 | It was propagated as Modi vaccine: @RajeevRai,... | 2021-06-08 05:49:31 | 1 | 7 | Grabyo | 6509832 | CNNnews18 | News18 | 2007-06-01 20:31:01 | Lightning fast alerts, #BreakingNews from Indi... | 62 | 4597047 | 403 | India | 1 | 0 | |
| 719 | 1402140137312587777 | RT @news18dotcom: An immunocompromised individ... | 2021-06-08 05:47:13 | 1 | 0 | TweetDeck | 6509832 | CNNnews18 | News18 | 2007-06-01 20:31:01 | Lightning fast alerts, #BreakingNews from Indi... | 62 | 4597047 | 403 | India | 1 | 0 | |

Count of spam users and genuine users:

Total count : 930

```
[ ]  1 df.shape

(930, 25)
```

Genuine : 720

```
1 Total_leg_data.shape
```

```
(720, 24)
```

Fake: 210

```
1 Total_spam_data.shape
```

```
(210, 25)
```

## Logistic Regression model:

```
1 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size=0.20, random_state=24)
```

```
1 from sklearn.linear_model import LogisticRegression
2
```

```
1 model=LogisticRegression(max_iter=1000)
```

```
1 model.fit(X_train, y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=1000,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

```
1 prediction=model.predict(X_test)
```

```
1 model
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=1000,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

```
[166] prediction

      array([0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0,
             0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1,
             0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0,
             0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1,
             0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,
             0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0,
             0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1,
             1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
             1, 1, 1, 0, 0, 0, 0, 0, 0, 0])

[170] Acc = accuracy_score(y_test, prediction)

[173] print("LR Model Acc : ",Acc*100)

      LR Model Acc :  95.33
```

## 8. Conclusion

In today's world, social media platforms are being used on a daily basis and have become an important part of our lives. The number of people on social media platforms are incrementing at a greater level for malicious use. The no. of friends to the no. of followers of any account are easily available in the account profiles and no rights are violated of any accounts. In order to accomplish the task of detecting, identifying and eliminating the fake accounts we establish using different machine learning models.

## 9. References


I. X. Chen and O. Martinez, "In a world that counts: Clustering and detecting fake social engagement at scale", *25th International Conference on WWW*, pp. 111-120.

II. D. Song and V. Paxson, "Suspended accounts in retrospect: an analysis of twitter spam", *ACM SIGCOMM conference*, pp. 243-258, 2011.

III. Wiesław Wolny University of Economics in atowice ul. 1 Maja 50, 40-287 Katowice, Poland - Knowledge Gained from Twitter Data

IV. Yusuf rslan ysenur Birturk Bekjan Djumabaev Dilek uc uk -∗Department of Computer Engineering, Middle East Technical University, Ankara, Turkey -Real-Time Lexicon-Based Sentiment Analysis Experiments On Twitter with A Mild (More Information, Less Data)Approach