March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

Detection of Fake Profile on Social Media Using Machine Learning and Feature Selection Techniques

Himanshu Kumar[1], Amrita[2]

[1,2]Department of Computer Science and Engineering, SET, Sharda University, Greater Noida, Uttar Pradesh, India

Article Info
Volume 83
Page Number: 9187 - 9198
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 09 April 2020*

Abstract

Social media facilitates the sharing of not only feelings and expressions but ideas and information too. Meanwhile, when someone attempts to clone one's profile with a malevolent intention, it not only breaches its privacy but can sabotage in other senses too. Many researchers in the past have endeavoured to prevent these kinds of malicious pursuits on the internet. With the help of Machine Learning (ML) and Feature Selection techniques, fake profiles can be detected at an introductory stage so that one is not capable of performing scurrilous efforts on the site. This work is an endeavour to detect whether a profile is fake or not based on users profile information. In this research, a model is proposed based on data pre-processing, feature selection and ML techniques to detect fake profile. The data pre-processing encompasses the measures like removing null values, encoding, and feature scaling. Feature selection is performed as an endeavour to reduce the dimension of the dataset and avoid overfitting. After applying feature selection, the number of features gets reduced from 34 to 11. Seven single ML techniques are employed to evaluate the effectiveness of data-pre-processing and reduced 11 features to detect fake profile. The results evince that data pre-processing and feature selection techniques improve the accuracy, precision, recall, and F1-Score of ML techniques and hence the performance of model.

Keywords: *fake profile detection, feature selection, machine learning, profile cloning, twitter dataset analysis.*

INTRODUCTION

With the expeditious growth in the services provided by the internet that too at a cheaper rate, an average human being spends most of its time connecting with people on social media through electronic devices. The technological preferment has made the devices smarter than ever before. Also, social media is the platform where a human being can come and express one's feeling and life events.

There are various types of social media which include social networks, video conferencing, blogging, business networks, social gaming, etc. YouTube, Facebook, Google+, Linked In, and Twitter stands top in the ranking of the most widely used social media platforms in the world. All these social media platforms operate under a similar transmission model. Social media, in contrast to traditional media, operates under a dialogic transmission model (many sources to many receivers) [1].

Social media is no longer stuck to a particular domain. It is used in multiple fields like posting job vacancies, raising awareness, spreading environment concerns, marketing, sales, etc. Meanwhile, it is used for evil purposes like election manipulation, muckraking, swindling, etc. These are a few of the activities that have become very common nowadays. Profile cloning is one such activity that involves creating a fake profile and using someone else's identity to post in their feed. Cloning one's profile for pecuniary interests or some other vicious intentions is also against the cyber laws. Thus

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

cloning of social media profiles is quite a concern nowadays.

The social media platforms must protect the privacy of their users. This can be accomplished by detecting the fake profiles and destroying them as soon as it is created on the platform. In an attempt to protect the privacy of users, a model is proposed to detect the fake profile on Twitter using data pre-processing, feature selection and machine learning (ML) technique.

The paper is organized as follows. A review of related work is presented in Section 2. In section 3, the proposed methodology is introduced. The experimental results and analysis is presented in Section 4 and conclusion and future work in Section 5.

II. LITERATURE REVIEW

Few works that are related to this research are:

In [2], Natural Language Processing (NLP) is used as a pre-processing technique. Steps such as stemming, tokenization, stop word removal, etc. are performed which are time taking on huge datasets. Classification algorithms such as Support Vector Machine (SVM), Naïve Bayes (NB), iSVM are used which achieved an accuracy of 77.4%, 77.3%, and 90% respectively.

In [3], the twitter dataset is gathered from April 2010 to July 2010 to perform analysis. The dataset is quite imbalanced as the ratio of fake to legitimate users is 1:10 and thus is changed to an equal ratio by removing tweets that are not in English dialects. Tweets are divided into three categories i.e. within a month, within two months and four months. Features extraction and model building time in seconds(s) and classification Accuracy (Acc) are used as evaluation measures for comparison, shown in Table I.

Table I: Performance measure of the model

	Tweets within		
	1 Month	2 Months	4 Months
Features Extraction Time(s)	35.00	59.00	87.00
Model Building Time(s)	4.15	5.41	6.96
Acc (%)	94.30	94.80	94.9

In [4], the verification is done manually on 13,000 purchased fake followers and 5,386 genuine followers. The features passed to an ML model to classify users into fake and genuine are: statuses count, followers count, followees count, favourites count, followers, listed count. The values of Cumulative Distribution Frequency for fake and genuine users are quite different. Three ML algorithms: SVM, Simple logistic and k-Nearest Neighbor (k-NN) (k=1) are used using 10 fold cross-validation. The results are shown in Table II.

Table II: Accuracy of ML algorithms

The algorithm	Acc (%)
SVM	60.48
Simple Logistic	90.02
k-NN (k=1)	98.74

A model based on a similarity between the users' friends' networks is proposed to discover fake accounts in social networks in [5]. Similarity measures such as common friends, cosine, Jaccard, L1-measure, and weight similarity are calculated from the adjacency matrix of the corresponding graph of the social network. The SVM using medium Gaussian is employed to evaluate the proposed model using the Twitter dataset. It achieved an area under the curve of 100% and has a low false-positive rate of 2%. In the proposed method, the user-friend network structure is analyzed and the fake users are predicted by computing similarity and the classifier algorithms. The fake accounts must work in the network for the possibility of recognizing them as genuine or fake, by scrutinising their friend's networks.

A new algorithm, SVM-Neural Network (SVM-NN), is proposed in [6] to provide an efficient detection for fake Twitter accounts and bots. Few feature selection and dimension reduction techniques such as Principal Component Analysis

Published by: The Mattingley Publishing Co., Inc.

(PCA), Correlation, etc. and three classification algorithms such as SVM, NN, and SVM-NN are used to identify the target account as genuine or fraudulent. The Acc, False Positive (FP) and False Negative (FN) are shown in Table III.

An ML and NLP system is presented to observe the fake profiles in online social networks in [7]. Moreover, the SVM classifier and NB algorithm are added to increase the detection accuracy rate of the fake profiles.

The paper reports on a study that focused on detecting fake accounts created by humans, as opposed to those created by bots in [8]. Investigations are conducted to examine whether the results from past studies to detect bot accounts could be applied successfully to detect fake human accounts. A corpus of human accounts is enriched with engineered features that had previously been used to successfully detect fake accounts created by bots. These features are applied to various supervised ML models such as linear SVM, AdaBoost, and Random Forest (RF). The models are trained to use engineered features without relying on behavioural data. This made it possible for these ML models to be trained on very little data, compared to when behavioural data is included. The Acc, F1-Score (F1-S) and Precision Recall Curve (PRC) are shown in Table IV.

Table IV: Results of supervised machine learning models

Model	Acc(%)	F1-S(%)	PRC(%)
Linear SVM	68.05	32.16	27.76
RF	87.11	49.75	49.90
AdaBoost	85.91	47.54	49.53

III. PROPOSED METHODOLOGY

The below mentioned are the steps performed towards building the proposed model whose diagram is depicted in Fig.1.

Step 1: Data Pre-processing.

Step 2. Feature Selection.

Step 3.Detection of fake or genuine profile.

Step 4.Evaluation of the proposed method.

Step 1: Data Pre-processing

The foundation step towards building an ML model is data preparation. The raw data can be gathered from multiple sources like sensors, records, databases, etc. and then prepared according to the requirement. In this work, dataset [9] is used, which consists of 1,337 and 1,481 samples of fake and genuine users respectively. It is then merged into a single dataset. Each sample from the dataset consists of 34 features. The dataset initially has five colour columns that contained colour codes in the hexadecimal (#RRGGBB) format. It gets converted into (RGB) format which brought it to a range of (0-255). After performing this step, the number of features increases by 10. Further, One Hot Encoding is performed for nominal data and Label Encoding performed for ordinal data to convert them into numerical values. The column "name" is assigned a respective gender value to it. This is done to find a pattern between a fake user and its associated gender.

Table III: Results after applying SVM, NN, SVM-NN on the proposed feature sets

Feature Set	SVM			NN			SVM-NN		
	Acc	FP	FN	Acc	FP	FN	Acc	FP	FN
[11]	0.886	0.111	0.001	0.737	0.059	0.203	0.912	0.086	0.001
PCA	0.914	0.039	0.653	0.653	0.278	0.067	0.922	0.033	0.043
Correlation	0.923	0.036	0.822	0.079	0.097	0.097	0.983	0.033	0.003
Regression	0.947	0.035	0.888	0.040	0.071	0.071	0.96	0.027	0.011
Wrapper-SVM	0.956	0.039	0.833	0.052	0.114	0.114	0.965	0.027	0.007

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

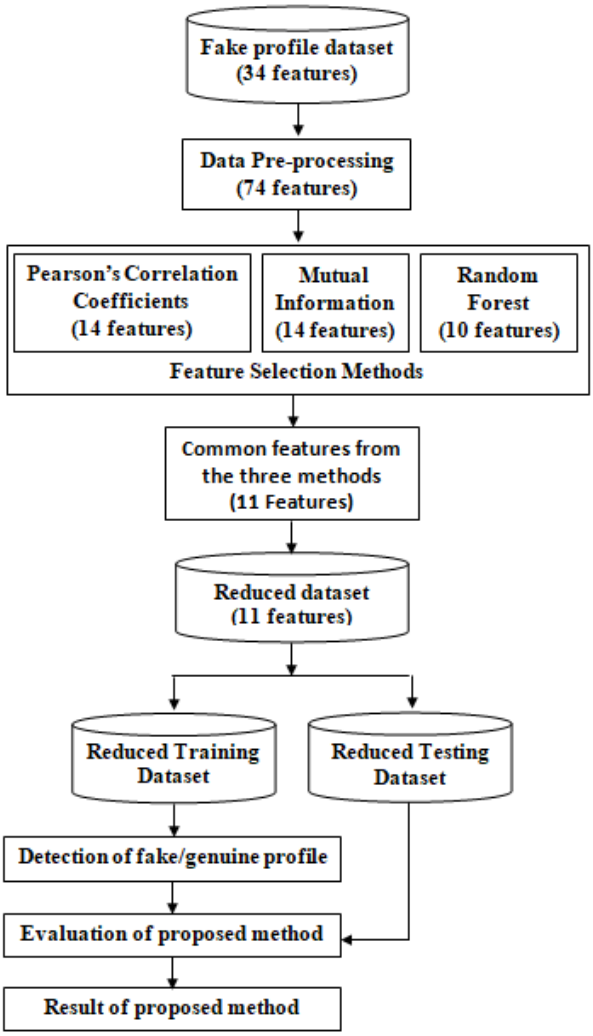

Fig. 1: Proposed methodology for the model

To scale the feature values, feature scaling is performed. For the model to achieve higher accuracy the features are needed to be scaled to almost an equal range. MinMaxScaler [10] is used in this work whose formula is given in Eq. 1.

$$y=\text{minrange}+\frac{(x_i-x_{min})}{(x_{max}-x_{min})}(maxrange-minrange) \quad (1)$$

where y, xi, xmax, and xmin are y-scaled value, current value, maximum value, and minimum value in the current column respectively. minrange and maxrange are 0 and 1 respectively by default, if not provided in argument.

Step 2: Feature selection

Features contribute immensely to the accuracy of an ML model, but not all features are equally important. The ones not considered significant can be termed as noise. Thus it's essential to find an optimal set of features which reduces the noise to its minimum. This is where feature selection comes into play. Feature selection [12] is the method of selecting the best possible features from a set of available ones using various methods such as:

Filter method [13]: This method is not dependent on any ML algorithm. It tells about the dependency between two variables.

Pearson's correlation coefficients (PCC) [14]: It gives a value between -1 to +1 where -1 tells about least dependency and +1 says most dependency. Given a pair of random variables (X,Y), the formula for PCC is given in Eq.2 :

$$\rho_{X,Y}=\frac{cov(X,Y)}{\sigma_X\sigma_Y} \quad (2)$$

wherecov, σX, and σY are covariance, standard deviation of X and standard deviation of Y respectively.

Mutual Information (MI) [15]: It is rank based feature selection technique which uses the measure of dependency between two variables. It is a measure of the amount of information that one random variable has about another variable.

Wrapper method [16]: It is an iteration method that starts with having either all the features or no features at all in the beginning. It then either increases or decreases the number of features depending on the feature score. At the end of this process, the 'k' best features are selected.

Embedded method [17]: This method is completely algorithm dependent. It has the advantages of both the Filter and Wrapper methods. RF classifier, Decision Tree (DT) have their feature selection methods.

The PCC, MI, and RF feature selections techniques are employed to obtain the features based on their

Published by: The Mattingley Publishing Co., Inc.

top ranks. Then, common features from these three sets are selected based on their rank to create final features set. This feature selection approach selects the relevant features and removed redundant and irrelevant features. Finally, 11 features are selected from 34 features.

Step 3: Detection of fake or genuine profile

After finishing the pre-processing and feature selection steps, the next step is to apply ML techniques on pre-processed and selected features. In this step, supervised ML techniques as k-NN[19], DT[20], RF[21], Logistic Regression (LR) [22], Stochastic Gradient Descent (SGD) Classifier[23], SVM[24], and Bernoulli NB (BNB)[25] are used for building the model. Since this is a supervised learning classification approach [18], the pre-processed data is split into training dataset and testing dataset. The training dataset is used to train the model, whereas testing dataset is used to test the model. The model, when trained, is evaluated after passing the testing data through it. The testing data is obscure to the model and when tested upon gives the output as to whether the user is genuine or fake.

Step 4: Evaluation of the proposed method

The evaluation metrics used in the proposed method are Acc, Precision, Recall, and F1-S. The results for the same are presented in Section 4.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

Several experiments have been performed to evaluate the performance of proposed model presented in Section 3 in terms of Acc, Precision, Recall, and F1-S. The steps of the proposed model have been carried out using Python [26] on the given dataset. The features present in the dataset before performing any data pre-processing step has its feature name (Name) and feature number (#) shown in Table V.

Table V: Feature # and Name before data pre-processing

#	Name	#	Name
1	id	18	profile_banner_url
2	name	19	profile_use_background_image
3	screen_name	20	profile_background_image_url_https
4	statuses_count	21	profile_text_color
5	followers_count	22	profile_image_url_https
6	friends_count	23	profile_sidebar_border_color
7	favourites_count	24	profile_background_tile
8	listed_count	25	profile_sidebar_fill_color
9	created_at	26	profile_background_image_url
10	url	27	profile_background_color
11	lang	28	profile_link_color
12	time_zone	29	utc_offset
13	location	30	protected
14	default_profile	31	verified
15	default_profile_image	32	description
16	geo_enabled	33	dataset
17	profile_image_url	34	updated

Then data pre-processing step is performed to fabricate the features and it ended up having 74 features whose name and the number are given in Table VI.

Table VI: Features number (#) and name after data pre-processing

SNo	#	Name	SNo.	#	Name
1	2.1	sex_code	38	13.19	London
2	4	statuses_count	39	13.20	Madrid
3	5	followers_count	40	13.21	Melbourne
4	6	friends_count	41	13.22	Moscow
5	7	favourites_count	42	13.23	Mountain Time(US & Canada)
6	8	listed_count	43	13.24	New Delhi
7	9.1	created_	44	13.25	Pacific Time

Published by: The Mattingley Publishing Co., Inc.

		at_minute			(US& Canada)
8	9.2	created_at_hour	45	13.26	Paris
9	9.3	created_on_date	46	13.27	Prague
10	9.4	created_in_month	47	13.28	Quito
11	9.5	created_in_year	48	13.29	Rome
12	11.1	de	49	13.30	Tehran
13	11.2	en	50	13.31	Vienna
14	11.3	Es	51	13.32	West Central Africa
15	11.4	Fr	52	13.33	Yerevan
16	11.5	Gl	53	14	default_profile
17	11.6	It	54	15	default_profile_image
18	11.7	Nl	55	16	geo_enabled
19	11.8	Tr	56	19	profile_use_background_image
20	13.1	Abu Dhabi	57	21.1	profile_text_color_red
21	13.2	Amsterdam	58	21.2	profile_text_color_green
22	13.3	Arizona	59	21.3	profile_text_color_blue
23	13.4	Athens	60	23.1	profile_sidebar_border_color_red
24	13.5	Berlin	61	23.2	profile_sidebar_border_color_green
25	13.6	Bern	62	23.3	profile_sidebar_border_color_blue
26	13.7	Brasilia	63	24	profile_background_tile
27	13.8	Brussels	64	25.1	profile_sidebar_fill_color_red
28	13.9	Casablanca	65	25.2	profile_sidebar_fill_color_green
29	13.10	Chennai	66	25.3	profile_sidebar_fill_color_blue
30	13.11	Copenhagen	67	27.1	profile_background_color_red
31	13.12	Eastern Time (US & Canada)	68	27.2	profile_background_color_green
32	13.13	Edinburgh	69	27.3	profile_background_color_blue
33	13.14	Greenland	70	28.1	profile_link_color_red
34	13.15	Guadalajara	71	28.2	profile_link_color_green
35	13.16	Hawaii	72	28.3	profile_link_color_blue
36	13.17	International Date Line West	73	29	utc_offset
37	13.18	Istanbul	74	33	dataset

To reduce the dimension of the dataset, PCC, MI, and RF rank based feature selections techniques are employed. The features are arranged in descending order based on their ranks. The top 14, 14, and 10 ranks of features along with its feature number (#) obtained by PCC, MI, and RF are shown in Tables VII, VIII, and IX respectively. The reason for selecting the 'n' number of features is the huge difference between the scores of nth and (n+1)th feature. Then, common features from these three sets are selected based on their rank to create final features set. Finally, 11 features are selected. The final feature set is reduced to 33% of the original feature set.

Table VII: Ranks of the top 14 features using PCC

SNo	#	Coefficient	SNo	#	Coefficient
1	11.6	0.877	8	13.4	0.278
2	13.29	0.572	9	21.1	0.208
3	16	0.553	10	13.14	0.195
4	24	0.429	11	9.4	0.172
5	28.1	0.402	12	2.1	0.165
6	4	0.316	13	21.3	0.163
7	13.2	0.290	14	21.2	0.159

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

Table VIII: Ranks of top 14 features using MI

SNo	#	Variable Importance	SNo	#	Variable Importance
1	26	0.677	8	9.4	0.405
2	13.10	0.674	9	9.3	0.367
3	4	0.535	10	14	0.366
4	11.2	0.503	11	21.1	0.300
5	7	0.492	12	21.2	0.280
6	11.6	0.491	13	21.3	0.300
7	5	0.446	14	9.5	0.280

Table IX: Ranks of top 10 features using RF

SNo	#	Feature Importance	SNo	#	Feature Importance
1	4	0.270	6	14	0.056
2	11.6	0.151	7	11.2	0.041
3	29	0.130	8	25.3	0.031
4	13.10	0.118	9	16	0.029
5	9.4	0.057	10	13.29	0.026

k-NN, DT, RF, LR, SGD, SVM, and BNB are used for building the model. The performance of model is evaluated in terms of Acc, Precision, Recall, and F1-S. The performance of these classifiers in terms of Acc, Precision, Recall, and F1-S on Full Features, selected features from PCC, MI, RF, and common features from three sets are shown in Tables X, XI, XII, XIII and XIV respectively.

Table X: Performance of various ML techniques on Full Features

ML Techniques	Acc (%)	Precision (%)	Recall (%)	F1-S (%)
k-NN	96.4	99.5	93.5	96.4
DT	99.7	100	99.5	99.7
RF	99.8	99.7	100	99.8
LR	100	100	100	100
SGD	51.4	51.5	99.5	67.9
SVM	99.7	99.5	100	99.7
BNB	99.7	99.7	99.7	99.7

Table XI: Performance of various ML techniques on 14 features selected using PCC

ML Techniques	Acc (%)	Precision (%)	Recall (%)	F1-S (%)
k-NN	99.7	99.7	99.7	99.7
DT	99.3	99.5	99.7	99.5
RF	99.7	99.7	99.7	99.7
LR	99.5	99.5	99.5	99.5
SGD	100	97.8	96.1	98.0
SVM	100	99.7	99.5	99.7
BNB	99.1	99.2	99.5	99.3

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

Table XII: Performance of various ML techniques on 14 features selected using MI

ML Techniques	Acc (%)	Precision (%)	Recall (%)	F1-S (%)
k-NN	99.6	99.3	100.0	99.6
DT	99.8	100.0	99.7	99.8
RF	99.8	99.7	100.0	99.8
LR	99.5	99.5	99.5	99.5
SGD	91.0	96.0	86.7	91.1
SVM	99.5	99.1	100.0	99.5
BNB	98.3	99.7	97.1	98.4

Table XIII: Performance of various ML techniques on top 10 features selected using RF

ML Techniques	Acc (%)	Precision (%)	Recall (%)	F1-S (%)
k-NN	99.2	99.3	99.3	99.3
DT	99.7	99.7	99.7	99.7
RF	99.7	99.7	99.7	99.7
LR	99.4	99.5	99.3	99.4
SGD	86.2	99.4	74.4	85.3
SVM	99.6	99.3	100	99.6
BNB	99.5	99.5	99.5	99.5

Table XIV: Performance of various ML techniques on 11 features using Common Features selected from three techniques

ML Techniques	Acc (%)	Precision (%)	Recall (%)	F1-S (%)
k-NN	99.7	99.7	99.7	99.7
DT	99.3	99.5	99.7	99.5
RF	99.7	99.7	99.7	99.7
LR	99.5	99.5	99.5	99.5
SGD	100.0	97.8	96.1	98.0
SVM	100.0	99.7	99.5	99.7
BNB	99.1	99.2	99.5	99.3

The model employing 11 common features are compared in terms of Acc, Precision, Recall, and F1-S with the model employing features from Full Features, PCC, MI, and RF using seven ML techniques. The performances of these seven ML using Full Features, common features from three sets, PCC, MI, and RF in terms of Acc are presented in Table XV, in terms of Precision in Table XVI, in terms of Recall in Table XVII, and in terms of F1-Score in Table XVIII. Results show that common selected features from three sets outperform PCC, MI, and RF by many ML techniques depicted in bold and also perform better or near to equal of Full Features as shown in Tables XV to XVIII.

Table XV: The performance of various ML techniques using common features from three sets, PCC, MI, and RF in terms of Acc

ML Techniques	Full Features	Common Features	PCC	MI	RF
k-NN	96.4	**99.7**	98.5	99.2	99.6

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

DT	99.7	99.3	98.6	99.7	99.8
RF	99.8	99.7	99.4	99.7	99.8
LG	100.0	**99.5**	98.3	99.4	99.5
SGD	51.4	**100.0**	53.4	86.2	91.0
SVM	99.7	**100.0**	99.4	99.6	99.5
BNB	99.7	99.1	98.2	99.5	98.3

Table XVI: The performance of various ML techniques using common features from three sets, PCC, MI, and RF in terms of Precision

ML Techniques	Full Features	Common Features	PCC	MI	RF
k-NN	99.5	**99.7**	98.8	99.3	99.3
DT	100.0	99.5	99.3	99.7	100.0
RF	99.7	**99.7**	99.5	99.7	99.7
LG	100.0	**99.5**	99.5	99.5	99.5
SGD	51.5	97.8	53.4	99.4	96.0
SVM	99.5	**99.7**	99.3	99.3	99.1
BNB	99.7	99.2	99.5	99.5	99.7

Table XVII: The performance of various ML techniques using common features from three sets, PCC, MI, and RF in terms of Recall

ML Techniques	Full Features	Common Features	PCC	MI	RF
k-NN	93.5	99.7	98.4	99.3	100.0
DT	99.5	**99.7**	98.2	99.7	99.7
RF	100.0	99.7	99.3	99.7	100.0
LG	100.0	**99.5**	97.3	99.3	99.5
SGD	99.5	96.1	100.0	74.4	86.7
SVM	100.0	99.5	99.5	100.0	100.0
BNB	99.7	**99.5**	99.1	99.5	97.1

Table XVIII: The performance of various ML techniques using common features from three sets, PCC, MI, and RF in terms of F1-Score

ML Techniques	Full Features	Common Features	PCC	MI	RF
k-NN	96.4	**99.7**	98.6	99.3	99.6
DT	99.7	99.5	98.7	99.7	99.8
RF	99.8	99.7	99.4	99.7	99.8
LG	100.0	**99.5**	98.4	99.4	99.5
SGD	67.9	**98.0**	69.6	85.3	91.1
SVM	99.7	**99.7**	99.4	99.6	99.5
BNB	99.7	99.3	99.3	99.5	98.4

The Figs 2 to 5 show comparative charts for ML techniques performance in terms of Acc, Precision, Recall, and F1-S on Full Features, reduced feature sets obtained by common features from three sets, PCC, MI, and RF respectively. As it can be seen

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

from the Figs 2 to 5 that common features selected from three sets has consistent and better performance over other feature selection techniques and Full Features.

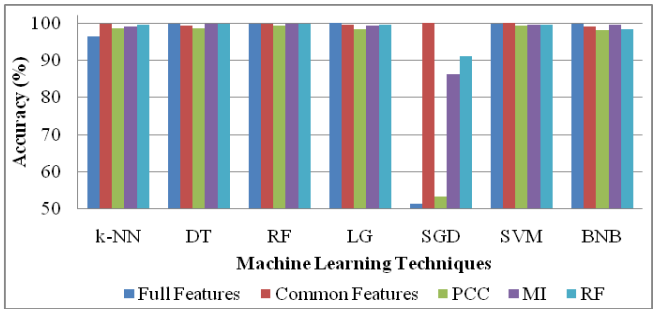

Fig. 2: Accuracy of different ML algorithms using selected features from different feature selection techniques

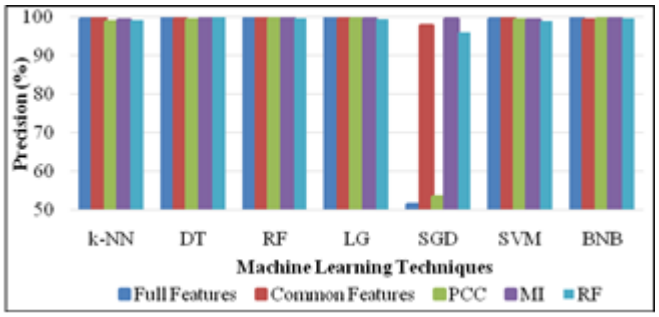

Fig. 3: Precision of different ML algorithms using selected features from different feature selection techniques

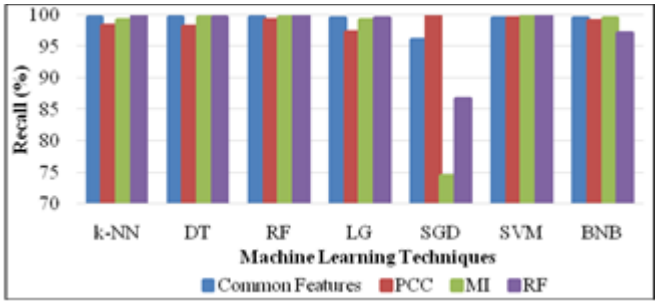

Fig. 4: Recall of different ML algorithms using selected features from different feature selection methods

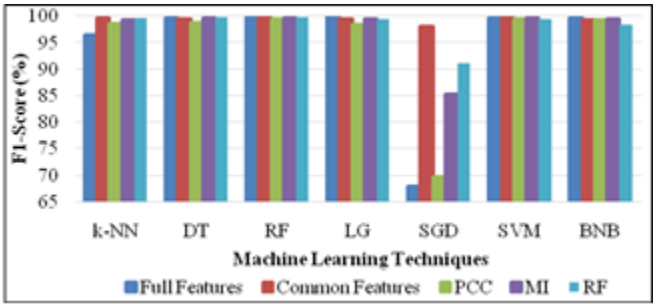

Fig. 5: F1-Scores of different ML algorithms using selected features from different feature selection techniques

V. CONCLUSION AND FUTURE SCOPE

In this proposed work, detection of fake profiles on social media platforms is performed. Various steps have been implemented for building the Machine Learning model which includes: data pre-processing, feature selection, model building, and model evaluation. Using data pre-processing maximum information can be extracted from the data. The same information when converted into knowledge can be used to solve various existing problems such as detecting fake users on social networking sites. The purpose of performing feature selection is to reduce the computational cost. The supervised learning algorithms used in this work performed almost equally on a similar set of features which also proves that the performance of a model is dependent more on the quality of the data passed to it than the algorithm used.

The feature selection techniques can be improved further to find such a set of features which can improve the performance of the model. The performance of model can also be improved by using hybrid, ensemble of classifiers or deep learning algorithm.

REFERENCES

[1] J. V. Pavlik and S. McIntoch "Converging Media: A New Introduction to Mass Communication", 4th Edition. New York, NY: Oxford University Press. ISBN 978-0-19-934230-3, 2015.

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

[2] A. K. Ojo, "Improved Model for Detecting Fake Profiles in Online Social Network: A Case Study of Twitter", Journal of Advances in Mathematics and Computer Science, vol. 33, no. 4, pp. 1-17, 2019.

[3] M. M. Swe and N.N. Myo, "Fake Accounts Classification on Twitter", International Journal of Latest Engineering and Management Research (IJLEMR), vol 3, no. 6, pp. 141-146, June 2018.

[4] A. Khalil, H. Hajjdiab and N. Al-Qirim, "Detecting Fake Followers in Twitter: A Machine Learning Approach" International Journal of Machine Learning and Computing, vol. 7, no. 6, pp. 198-202, December 2017

[5] M. Mohammadrezaei, M. E. Shiri, and A. M. Rahmani, "Identifying Fake Accounts on Social Networks Based on Graph Analysis and Classification Algorithms", Security and Communication Networks, pp. 1-8, 2018.

[6] S. Khaled, H. M. O. Mokhtar, and N. El-Tazi, "Detecting Fake Accounts on Social Media", IEEE International Conference on Big Data (Big Data), December 2018.

[7] P. S. Rao, J. Gyani, and G. Narsimha, "Fake Profiles Identification in Online Social Networks Using Machine Learning and NLP", International Journal of Applied Engineering Research, vol. 13, no. 6, pp. 4133-4136, 2018.

[8] E. V. D. Walt and J. Eloff, "Using Machine Learning to Detect Fake Identities: Bots vs Humans", IEEE Access, vol. 6, pp. 6540-6549, March 2018.

[9] S. Cresci, R. D. Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "Fame for sale: efficient detection of fake twitter followers", Decision support systems, vol. 80, pp. 56–71, 2015.

[10] S. G. K. Patro1, K. K. Sahu, "Normalization: A Preprocessing Stage", International Advanced Research Journal in Science, Engineering and Technology, vol. 2, no. 3, pp. 20-22, March 2015

[11] C. Yang, R. Harkreader, and G. Gu, "Empirical evaluation and new design for fighting evolving twitter spammers", IEEE Transactions on Information Forensics and Security, vol. 8, no. 8, pp. 1280–1293, 2013

[12] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning", Artificial Intelligence, vol. 97, no. 1-2, pp. 245-271, 1997.

[13] H. Liu and H. Motoda, "Feature Selection for Knowledge Discovery and Data Mining", Boston: Kluwer Academic, 1998.

[14] S. Senthilnathan, "Usefulness of Correlation Analysis",
July 2019, Available at SSRN: https://ssrn.com/abstract=3416918

[15] T. M. Cover and J. A. Thomas, "Elements of Information Theory", 2nd edn. Wiley-Interscience, New Jersey, JA, 2006.

[16] R. Kohavi and G. H. John, "Wrappers for feature subset selection", Artificial Intelligence, vol. 97, no. 1-2, pp. 273-324, 1997.

[17] S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection", in Proceedings of the Third International Conference on Machine Learning, pp. 74-81, Dalian, China, June 2001.

[18] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers", IBM Journal of Research and Development, vol. 3, no. 3, pp. 210–229, July 1959.

[19] S. Thirumuruganathan, "A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm", World Press, 2010.

[20] J. R. Quinlan, "C4.5: Programs for Machine Learning", San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[21] L. Breiman, "Random forests, Machine Learning", vol. 45, no. 1, pp. 5-32, 2001.

[22] J. S. Cramer, "The origins of logistic regression", 119. Tinbergen Institute, pp. 167–178, 2002.

[23] L. Bottou, "Large-scale machine learning with stochastic gradient descent", in Proceedings of the19th International Conference on Computational Statistics (COMPSTAT'10), pp. 177-187, Paris France, Aug. 2010.

Published by: The Mattingley Publishing Co., Inc.

March - April 2020
ISSN: 0193-4120 Page No. 9187 - 9198

[24] C. Cortes and V. Vapnik, “Support-vector networks”, Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.
[25] http://scikit-learn.org/stable/modules/naive_bayes.html
[26] https://www.python.org/