IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Special Issue-15, May 2019 E-ISSN: 2347-2693

A Framework for Detection of Accuracy of Spam in Twitter

Pooja Naik[1], Monisha S[2], Supritha Shetty [3], Pooja NR[4], Anoop N Prasad[5]

[1,2,3,4,5] Student, Department of Computer Science, East west Institute of Technology, Bangalore, India

DOI: https://doi.org/10.26438/ijcse/v7si15.105110 | Available online at: www.ijcseonline.org

***Abstract*—** With millions of users tweeting around the world, real time search systems and different types of mining tools are emerging to allow people tracking the repercussion of events and news on Twitter. Trending topics, the most talked about items on Twitter at a given point in time, have been seen as an opportunity to generate traffic and revenue. Spammers post tweets containing typical words of a trending topic and URLs, usually obfuscated by URL shortness, that lead users to completely unrelated websites. This kind of spam can contribute to de-value real time search services unless mechanisms to fight and stop spammers can be found. To solve this issue, we propose to take tweet text features along with user-based features. We have evaluated our approach with natural language processing and the naïve-Bayes machine learning algorithm.

***Keywords*—** Twitter, tweets, spam, navie bayes, natural lanugage processing

I. INTRODUCTION

ONLINE interpersonal organizations (OSNs, for example, Twitter, Facebook, and some venture interpersonal organization [1], have turned out to be to a great degree mainstream in the most recent couple of years. Twitter, which was established in 2006, has got to be a standout amongst the most well known microblogging administration web page. Nowadays, 200 million Twitter users generate over 400 million new tweets per day [2].

Given that spammers are increasingly arriving on Twitter, the success of real time search services and mining tools relies at the ability to distinguish valuable tweets from the spam storm. In this paper, we firstly address the issue of detecting spammers on Twitter. To do it, we propose a 4-step approach. First, we crawled a near-complete dataset from Twitter, containing more than billion tweets. Second, we created a labeled collection with users "manually" classified as spammers and non-spammers. Third, we conducted a study about the characteristics of tweet content and user behavior on Twitter aiming at understanding their relative discriminative power to distinguish spammers and non-spammers. Lastly, we investigate the feasibility of applying a super-vised machine learning method to identify spammers. We found that our approach is able to correctly identify the majority of the spammers (90%), misclassifying only 10% of non-spammers. Tingmin Wu [3] performed spam tweet detection based on deep learning. They used word vector to train their model, but they have not explored user or tweet based features to address the problem. On the other side, Chao Chen [4] used lightweight features (user's and tweet's specific feature) that are suitable for real-time spam tweet detection. We also investigate different tradeoffs for our classification approach namely naïve Bayes, the attribute importance and the use of different attribute sets. Our results show that even using different subsets of attributes, our classification approach is able to detect spammers with high accuracy. the expansion of Twitter additionally adds to the development of spam. Twitter spam, which is alluded as spontaneous tweets containing noxious connections that coordinates casualties to outer destinations containing malware downloads, phishing, drug deals, or tricks, and so forth [5], has not just influenced various genuine clients additionally dirtied the entire stage. Amid the time of Australian Prime Minister Decision (August 2013), the Australian Constituent Commission (AEC) distributed a ready that affirmed its Twitter account @AusElectoralCom was hacked. A large portion of its devotees got immediate spam messages which contained malevolent connections [6]. The capacity to deal with helpful data is basic for both the scholarly world and industry to find shrouded bits of knowledge and foresee patterns on Twitter. However, spam significantly brings noise into Twitter [7].Thusly, the exa mination group, and Twitter itself, has proposed some spam discovery plans to make Twitter as a sans spam stage. For example, Twitter has connected some "Twitter guidelines" to suspend accounts on the off chance that they carry on unusually. Those records, which are oftentimes asking for to be companions with others, sending copy content, saying others clients, or posting URL-just substance, will be suspended by Twitter [8]. In this paper, we give a framework based on naïve Bayes machine learning approach that deals with various problems including accuracy shortage, time lag (BotMaker) and high processing time to handle thousands of tweets in 1 sec. Firstly, we have collected 400,000 tweets from HSpam14 [5] dataset. Then we further characterize the 150,000 spam tweets and 250,000 non-spam tweets. We also

derived some lightweight features along with the Top-30 words that are providing highest information gain from Bag-of-Words model. This approach has been detailed in section II. The architechture has been illustrated in section III

II. RELATED WORK

We prepare our dataset by collecting tweets corresponding to 400,000 tweet ids from HSpam14 [9]. We then created the features set mentioned in Table I on our dataset. In order to get information from tweets' text, we want to extract those words that can be strong indicators to classify the tweets as spam and non-spam. Features of Twitter-

Twitter lets accounts to "follow" other accounts which they are interested in. Unlike other social media platforms, the relationship between users is bi-directional instead of unidirectional links which mean one user may not be following one of his followers. The user can "like" or "retweet (RT)" a tweet which means sharing that tweet with his "followers". The relationship between users in Twitter is presented in Fig. 1. Each user has a unique Twitter username, and users can post tweets that refer others by adding their usernames with starting "@" character which is called as "mention" on Twitter. Users are immediately informed with notifications when a mention, like, or RT happens to one of his tweets.

A. Information Gain from Bag-of-Word Model After characterizing the spam and non-spam tweets' text into two separate documents, we construct the following sets*:*

US = Collection of unique words in the spam tweets' text. UNS = Collection of unique words in the non-spam tweets' text. For each word T in US and UNS we calculate the following probablity values: P(T|U S)= # of Spam tweets that contain T total # of Spam tweets (1) P(T|UNS)= # of Non-Spam tweets that contain T total # of Non-Spam tweets (2)
We calculate the information gain γT for each word T as follows:

$$P(T|Us) = \frac{\#\text{ of spam tweets that contain T}}{\#\text{total of spam tweets}}$$

$$P(T|NUs) = \frac{\#\text{ of non-spam tweets that contain T}}{\#\text{total of non-spam tweets}}$$

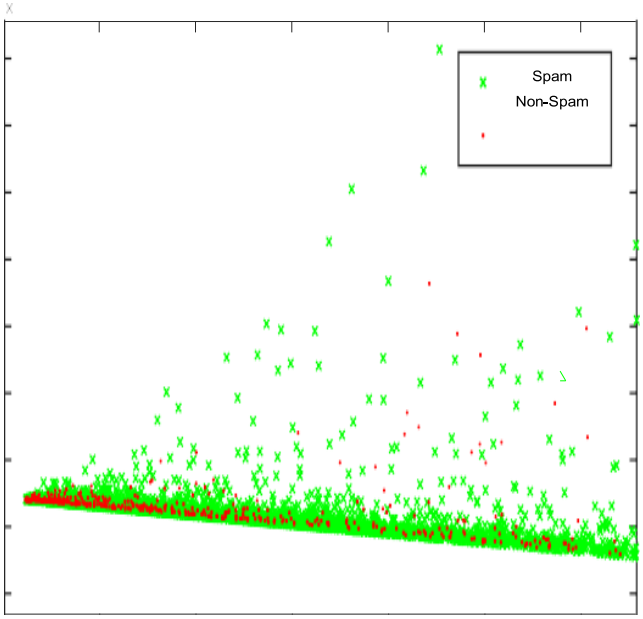


Fig. 1: Scatter plot of dataset showing distribution of two classes namely, spam(x) and non-spam(.)

P(T|Us) and P(T|NUs)

B. Extracting Lightweight Features- After collecting 400,000 labelled tweets, we extracted around 350,000 English tweets. Since we are receiving an arbitrary independent tweet from Twitter API, so we could not obtain the complete social graph of Twitter's users. Consequently, we take the feature set from work [1] that is more suitable for timely detection of Twitter spam. However, we add one more feature, i.e., no of non ASCII on top of those 12 features. From our analysis, we found that 88% of spam tweets use non-ASCII values to post a tweet.

We classify our feature sets in 3 categories.
We examine that features' values are not in same range.

D ^{I} = Matrix representation of Dataset-1 of size M*N.
N = number of features, M=number of tweets.
For representing the data using Feature-set-1, we normalize the data so that each of the feature has zero mean and unit standard deviation. We represent each feature using its bag-of-words representation. Here each feature is a word and corresponding value is the frequency of the word in the tweet.

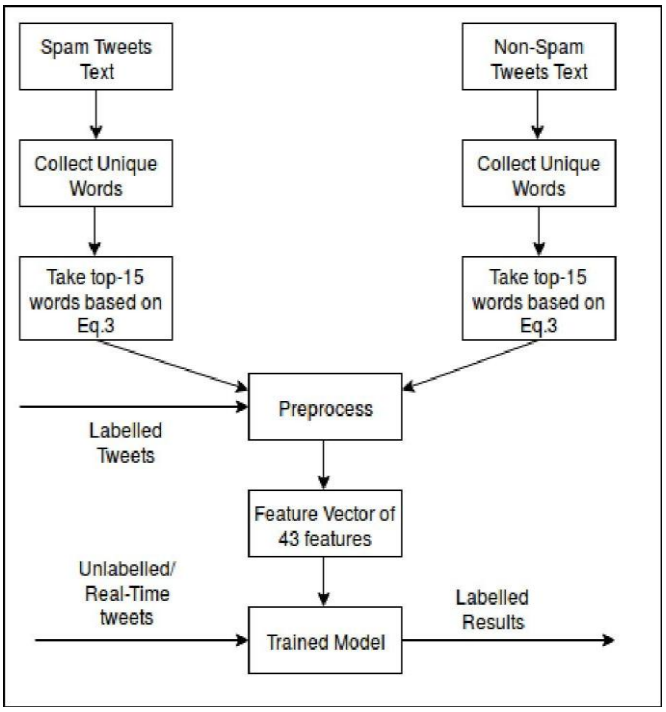


Fig. 1: Flow Diagram to pre-process the dataset for Information gain.

III. METHODOLOGY

This paper gives different 8 lightweight features extracted for tweet representation. Spam detection transformed to a binary classification problem in the feature space and can be solved by conventional machine learning algorithms. The auther evaluated the impact of different factors to the spam detection performance, which included spam to non-spam ratio, feature discretization, training data.

TABLE I: Feature Set

Feature Name	**Description**
account age	The age (days) of an accounl since its creation until the time of sending the most recent tweet
no_follower	The number of followers of this Twitter user
no_following	The number of followings/friends of this Twitter user
no_userfavourites	The number of favourites this Twitter user received
no_lists	The number of lists this Twitter user added
no_tweets	The number of tweets this Twitter user sent
no_retweets	The number of retweets this tweet

From the dataset, investigate the distribution of stop words and identify what we hope will be sets of frequent hashtags that are indicative of positive, negative and neutral messages. These stopwords are used to select the tweets that will be used for development and training. Pre-processing is one of the important steps in text mining, Natural Language Processing (NLP) and information retrieval (IR). which gives tokenization, normalization i.e. remove @,remove #and URL. Data preprocessing is used to extract interesting and non-trivial knowledge from unstructured text data. Information Retrieval is important for deciding which documents in a collection should be retrieved so that we can satisfy a user's need for information.

The top-20 most popular hashtags. All these 20 hash- tags appeared as trending up/down hashtags which were used for sampling the tweets.

Hashtag (#)	Freq.	Hashtag (#)	Freq.
TEAMFOLLOWBACK	666,328	SougoFollow	197,525
TFBJP	527,176	ipad	195,375
gameinsight	510,504	FOLLOWBACK	177,109
android	341,240	THF	165,762
OPENFOLLOW	332,857	FOLLOWNGAIN	149,916
FF	293,748	500aday	146,005
androidgames	286,706	AUTOFOLLOW	141,214
RETWEET	250,992	MUSTFOLLOW	138,040
RT	235,137	TEAMHITFOLLOW	136,043
IPADGAMES	232,335	MUSIC	129,852

Upload Input Data Set

This function will upload the dataset (tweets downloaded) for a particular #hash Tag.

Pre Processing Technique

Pre-processing techniques are applied on dataset to get clean data.

Remove @

The first pre-processing technique is remove @ which means it scans the whole document of input dataset and after comparing it with @ it deletes @ from every available comment with @.

Remove URL

The next step of pre-processing is remove URL where the whole input document gets scanned and compared with http:\\... and the comments having URL are deleted.

Remove Stop Words

Further move on to stop word removal being the next step in data pre-processing. Stop word removal exactly means that from the whole statement after scanning it removes the words like and, is, the, etc and only keeps noun and adjective from the statement.

Porter stemming algorithm

Porter stemmer' is a method for removing the commoner morphological and in flexional endings from words in English. Following are the steps of this algorithm:-

1. Gets rid of plurals and -ed or -ing suffixes.
2. Turns terminal y to i when there is another vowel in the system.
3. Maps double suffixes to single ones: -ization, ational, etc.
4. Deals with suffixes, -full, -ness etc.
5. Takes off -ant, -ence, etc. Removes a final –e.

Data Crawling

To collect data for hashtag-oriented spam tweet research, one key issue is to identify candidate hashtags. The tweets containing these candidate hashtags are then collected and labelled. Without the luxury of accessing all tweets or all hashtags, we rely1 on the trend reports trending hashtags in three categories: trending up, trending down, and most popular hashtags. As expected, we observe that the hashtags in most popular hashtags category do not change much for many days. To cover more varieties of hashtags in our data collection, we used the hashtags in trending up and trending down categories as query keywords to search for tweets on daily basis. On each day, we collect the trending hashtags in these two categories and we then use each collected hashtag as a query key2-word to collect tweets using Twitter's streaming API for that day.

A tweet is collected through the API if it contains the query key- word as a hashtag, word, or link in its content. On average 135 hashtags were used as query keywords on each day.

For time period spam detection, we have a tendency to any extracted twelve lightweight options for tweet illustration. Spam detection was then reworked to a binary classification drawback within the feature space and may be solved by typical machine learning algorithm. we have a tendency to evaluated the impact of various factors to the spam detection performance, including spam to non -spam ratio, feature discretization, coaching information size, information sampling, time-related information, and machine learning algorithms. The display the streaming spam tweet detection continues to be an enormous challenge and a robust detection technique ought to take under consideration the 3 aspects of knowledge, feature, and model. Twitter's Gushing Programming interface can be utilized for grouping. With a specific end goal to better comprehend ML calculations' power in grouping gushing spam tweets; we gave a principal assessment in this work. To accomplish this objective, we gathered countless. This information contained more than 600 million tweets We additionally removed some direct components for each tweet and inspected some ML calculations' execution on the identification of spam from different perspectives. Machine learning algorithms and Data mining can effectively reduce spam content by taking benefit of the gigantic quantity of information on the social media sites. In this paper naïve- Bayes a machine learning algorithms is used to categorize similar type of spam in twitter. Naïve Bayes was developed based on statistical theory.

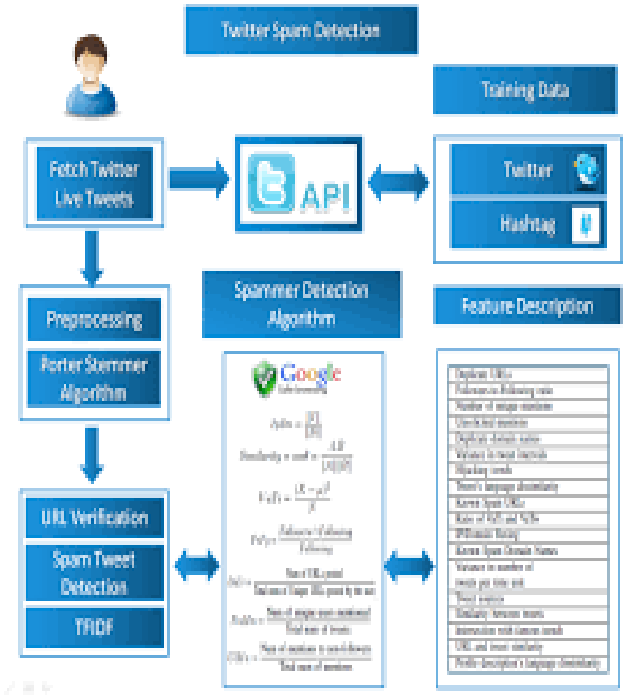


Fig.2: Architecture diagram for Twitter Spammer Detection

TABLE II: Sample Top-10 Words

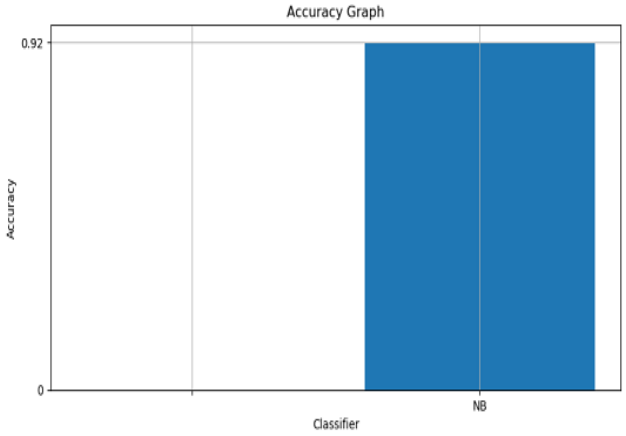
Top 5 Words from Spam Tweets	Top 5 Words from Non-Spam Tweets
harvested	rain
Tribez	asleep
Coins	rather
collected	college
unfollower	fell
openfollow	follback
Inspi	dinos
Build	bullshit
Smurf	child
Brainy	couch

IV. RESULTS AND DISCUSSION

In this section, we will measure the Twitter spam detection performance on our dataset by using the machine learning algorithm Naïve Bayes classifier along with natural language processing. We also extracted 8 lightweight features which can differentiate between that spam and the non-spam tweets from the labelled datasets. We also recognized that Features discretization was an important pre- process to ML- based spam detection. Secondly, increasing training data only cannot bring more benefits to detect Twitter spam after a certain number of training samples. We should try to bring more discriminative features or better model to further improve spam detection rate. Third, classifiers can detect more spam tweets when the tweets were sampled continuously rather than randomly selected tweets. From the third point of view, we have overall analyzed the reasons why classifier's performance reduced when training and testing data were in different days from three points of views. We came to the conclusion that the performance decreases due to the fact that the distribution of features changes of later days datasets, whereas the distribution of training datasets stays the same. Further, to illustrate the characteristics of extracted features, we used cdf figure. We averaged these features to machine-learning based spam classification later in our experiments. We also classify spam class as a positive class and non-spam class as a negative class. We determine the Accuracy as follows:

$$\text{Accuracy} = \frac{TP+TN}{\text{TP+TN+FP+FN}}$$

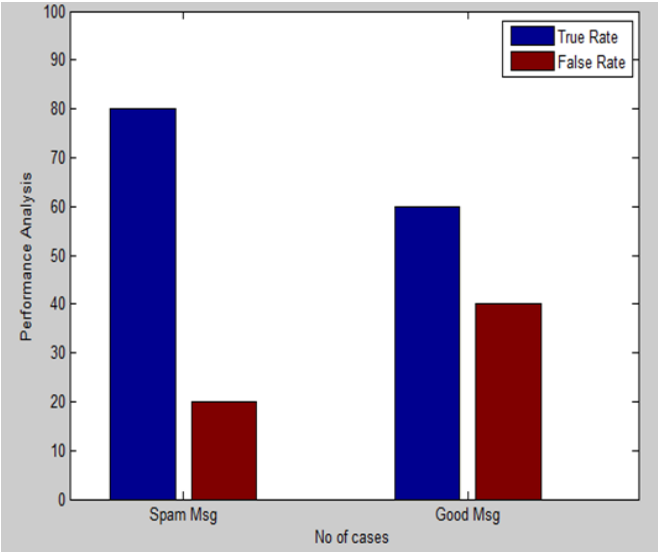
Naïve bayes classifier gives the finest accuracy i.e. 90-95%.



Accuracy Evaluation on Feature-set.

TABLE III: Performance Evaluation on Feature-set

Unit	Feature Set
Classifier	Naïve-Bayes
Accuracy	90-95%



V. CONCLUSION AND FUTURE SCOPE

In order to detect and prevent spammers in social networks several methods have been proposed and developed by many researchers. During our survey it is seen that spam detection in social networks using Naïve Bayesian approaches is highly effective and a combination of spam prevention filters will give higher accuracy.

In the future, we will keep on updating our Bag-of-Words model based on new spam tweets by implementing self-learning-algorithm. This problem will exist in streaming spam tweets detection, as the new tweets are coming in the forms of streams, but the training dataset is not updated. In future, we will be working on this issue. Also,we observe in our dataset that79% of spam tweets contain a malicious link. Frequent Pattern Mining of tweets' text can also be the vital aspect to distinguish Twitter spam in real-time. We will consolidate these three approaches to handle Spam Drift problem.

REFERENCES

[1] C. P.-Y. Chin, N. Evans, and K.-K. R. Choo, "Exploring factors influencingthe use of enterprise social networks in multinational professionalservice firms," J. Organizat. Comput. Electron. Commerce, vol. 25, no. 3, pp. 289–315, 2017.

[2] H. T sukayama, "Twitter turns 7: Users send over 400 million tweets per day," Washingon Post, Mar. 2017.

[3] T. Wu, S. Liu, J. Zhang, and Y. Xiang, "Twitter spam detection based on deep learning," in Proceedings of the Australasian Computer Science Week Multiconference, ser. ACSW '17. New York, NY,USA.

[4] C. Chen, J. Zhang, X. Chen, Y. Xiang, and W. Zhou, "6 million spam tweets: A large ground truth for timely twitter spam detection," in 2016 IEEE International Conference on Communications (ICC), June 2015, pp. 7065–7070.

[5] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammer on Twitter," presented at the 7th Annu. Collab. Electron. Messaging Anti-Abuse Spam Conf., Redmond, WA, USA, Jul. 2017.

[6] L. Timson, " Electoral commission Twitter account hacked, voters asked not to click," Sydney Morning Herald, Aug. 2013 [Online]. Available: http://www.s mh.com.au/it-pro/security- it/electoral-commission-twitteraccount- hacked-voters-asked- not-to-click-20130807-hv1b 5.html

[7] Z. Miller, B. Dickinson, W. Deitrick, W. Hu, and A. H. Wang, "Twitter spammer detection using data stream clustering," Inf. Sci., vol. 260, pp. 64– 73.

[8] K. Thomas, C. Grie r, D. Song, and V. Pa xson, Suspended accounts in retrospect: An analysis of Twitter spam," in Proc. ACM SIGCOMM Conf. Internet Meas., pp. 243– 258.

[9] K. Thomas, C. Grie r, D. Song, and V. Pa xson, Suspended accounts in retrospect: An analysis of Twitter spam," in Proc. ACM SIGCOMM Conf. Internet Meas.,pp. 243– 258.

[10] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammers on twitter," in In Collaboration, Electronic messaging, Anti- Abuse and Spam Conference.

[11] C. Pash., "The lure of Naked Hollywood Star Photos Sent the Internet into Meltdown in New Zealand, Bus. Insider, accessed on Aug. 1, 2015," https://tinyurl.com/yc93ssj4,2017.

[12] "BotMaker," blog.twitter,us/a/2014fighting-spam-with-botmaker.

[13] Witten and E. Frank. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016.

[14] C. Wu, K. Cheng, Q. Zhu, and Y. Wu. Using visual features for anti-spam filtering. In IEEE Int'l Conference on Image Processing (ICIP), 2017.

[15] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten, and I. Osipkov. Spamming botnets: Signatures and characteristics. In ACM SIGCOMM,.

[16] D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In Int'l Workshop on the Web and Databases (WebDB),.

[17] S. Garriss, M. Kaminsky, M. Freedman, B. Karp,

[18] D. Mazi`eres, and H. Yu. Re: Reliable email. In USENIX Conference on Networked Systems Design & Implementation (NSDI), 2016.

[19] Z. Gy¨ongyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In Int'l. Conference on Very Large Data Bases (VLDB), 2017.

[20] P. Heymann, G. Koutrika, and H. Garcia-Molina. Fighting spam on social web sites: A survey of approaches and future challenges. IEEE Internet Computing, 11, 2017.

[21] A Framework for Real-Time Spam Detection in Twitter Himank Gupta, Mohd. Saalim Jamal, Sreekanth Madisetty and Maunendra Sankar Desarkar Department of Computer Science and Engineering,Indian Institute of Technology Hyderabad, India,2018.

