International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-5S, May 2019

# Adaptive k-Nearest Centroid Neighbor Classifier for Detecting Drifted Twitter Spam

**L A Lalitha, Vishwanath R Hulipalled**

***Abstract*—With the growth of Internet and its related technologies have resulted in increased usage of smart and Internet connected devices and large amount of time is spent on Social Network. Nonetheless, because of increase in attractiveness of Social Network, cyber offenders are spreading spam on these networks to exploit possible targets. The spammers trap users to malware downloads or external phishing URLs, which has been an enormous problem for online safety and user quality of exposure. However, the existing research fails to detect spam in Twitter and has become a key issue in recent times. Recent work [14], focused on using Machine Learning (ML) approach for detecting spam in Twitter, by making use of the statistical features of Twitter data. However, adoption of such method affects the classification accuracy of ML algorithm. Because the Statistical Feature characteristics of spam tweets vary with respect to time. This problem is known as "Twitter Spam Drift". To address this problem, we present a novel non-parametric Adaptive K-Nearest Centroid Neighbor (AKNCN) Classifier. Further, for meeting real-time requirement the AKNCN is trained using one million spam tweets and one million non-spam tweets data. The AKNCN model can discover spam more efficiently than the state-of-the-art model. Experiment outcome shows the AKNCN attains significant performance with reference to Accuracy (A), F-Measure (F) and Detection Rate (DR) in real-world scenarios.***

***Keywords*— Nearest Centroid Neighbor, Machine Learning, Social Networks, Statistical Features, Spam Drift, Twitter Spam Detection.**

## I. INTRODUCTION

Twitter was founded in March 2006 in Sanfransisco, California. Since its inception Twitter is a successful Social Media platform in terms of its fast reachability to millions of users, ease of use and quick connectivity with people just by following them on Twitter. Twitter has emerged as a most popular and widely used Social Network platform in last decade among people from various sector, especially teenagers [1]. However, the rapid evolution of Twitter also resulted in increased number of spamming activities in Social Network. Twitter Spam is denoted as uninvited tweets composed of malicious or spam links that makes user or victims to access external links that is composed of phishing, drug sales, malware downloads, or scams, and so on [2], it not only affects subscriber experience, but also affects the entire network health. In late 2014, the Australian continent specifically New Zealand's Internet was seriously affected due to spread of spam URL and malware downloading spam. These spams are composed of links that claims to contain international actor photos, but in fact attract subscriber and direct them to download malware to carryout Distributed Denial of Service (*DDOS*) attacks [3]. Subsequently, Twitter and security companies are working to remove spammers to make spam free Twitter Social Network. A service Web Reputation Technology (*WRT*) by TrendMicro is a platform to blacklist user and eliminate spam URLs from subscribers who have its product installed [4]. Twitter also uses similar type of blacklisting filtering services namely BotMaker [5]. However, it fails to protect its subscribers from new type of spam which changes with respect to time [6]. Extensive analysis shows that, more than 90% victims or users visits these new kind of spam URLs before its being blocked by blacklist filtering services [7]. To get over the drawback of blacklists, state-of-the-art model have employed Machine Learning (*ML*) based detection approaches. These approaches make use of spam tweets or spammers historical or statistical features to identify spam without examining the URLs [8]. *ML* based detection models are composed of following steps. Firstly, historical or statistical feature are used to distinguish between non-spam and spam, which are extracted from Twitter users or tweets (such as user account age, number of characters in tweets, and number of friends or followers). Then a small set of data is labeled with classes, that is, non-spam and spam, as training data. Further, *ML* based classification algorithm is trained using these trained labeled sets, and lastly, the trained classification model is utilized to identify spam. Recently, number of *ML* based approaches have been presented in recent times which are composed of both Semi-Supervised or Supervised approaches [9], [10], [11], [12], and [13]. However, the extensive analysis presented in [14] shows that characteristic features of spam tweet varies with respect to time [15]. This problem is referred here as a "Twitter Spam Drift". The state-of-the-art *ML* classifiers are up to date with modified spam tweets set. As a result, the performance of them is inefficient as it is affected by "Spam Drift" while identifying new type of incoming spam tweets.

**Manuscript published on 30 May 2019.**
***** Correspondence Author (s)
**L A Lalitha**, School of C&IT, REVA University/Bengaluru
**Dr. Vishwanath R Hulipalled** School of C&IT, REVA University/Bengaluru

The reason is that the spammers are finding a way to escape from being detected and at the same time researchers also are working to detect spam. This leads spammers to intrude the exiting detection model by creating spam with similar features and semantic representation using different text or through posting more tweets [9], [16].

To address the "Spam Drift" problem, mean values of statistical features are used and distribution of the data is used to model changes is statistical features of the dataset [17]. There are two kinds of methods used in these distributions such as Parametric Model and Non-Parametric Model.

Parametric Model is used when there is a specific distribution in training set and when distribution is already known. However, these methods cannot be applied to Twitter spam detection, since Twitter spam data distribution is unknown. In Parametric Models we need to know the parameters of the model. On the other side, Non-Parametric Models to forecast new data, we need to know the data based on the parameters of the model along with the current state of the data. This method doesn't make rules of the dataset distributions which are utilized by state-of-the-art methods [18]. However, the spam tweet features distribution is changing impulsively from time to time [19], [20] and training sample distribution is not changed. As the *ML* algorithm learns feature from the unchanged training sample that are not changed are used to classify new tweet, the performance of state-of-the-art *ML* algorithm becomes inaccurate [19], [20]. To overcome the challenges, this work adopts similar architecture as [14]. Further, to attain superior classification performance than [14], this work presents a novel non-parametric Adaptive K-Nearest Centroid Neighbor Classifier model for Twitter Spam Detection.

The Contribution of our work is as follows:

This work presents a novel classification model namely AKNCN for twitter spam detection. Our model can address the Twitter Spam-Drift problem more efficiently than state-of-the-art technique [14]. Our model attains good performance in terms of Accuracy, Detection rate and F-measure. The rest of the paper is organized as follows. Research survey is carried out in section II. The proposed adaptive k-nearest centroid neighbor classifier to detect spam in Twitter is presented In section III. In last but one section, the experimental study is discussed. And in the last section, conclusion and future work is discussed.

## II. LITERATURE SURVEY

Rapid growth and increased Twitter popularity, spammers have transformed from social network such as blogs and emails to Twitter social network platform. Security organization and various research community is working hard to remove spam to make Twitter as a communication platform for social interaction. Security organization like Trend Micro [4] focused on blacklisting to waive of spam URLs. Though, due to time lag they failed to protect user on time. To overcome these issues is blacklisting [21] presented a heuristic rule (rules such as username pattern matching, keyword detection, and suspicious URL searching) that remove spam in Twitter. In [22] they considered a tweet with more than 3 hashtags to eliminate spam to address the impact of spam.

Later, [2], [10], [11], [12], [13] applied ML algorithm using feature such as content-based features and user account such as length of the tweets, number of followers, users account age, etc. to identify non-spammer and spammer. Further in [12] used Bayesian based classification model to detect spammer and [2] used Support Vector Machine (*SVM*) to detect non-spammer and spammer. In [23] used Random Forest classifier to detect spam in social network such as Myspace, Facebook and Twitter. In [24] used honeypots to obtain information about spammers and their statistical feature were extracted for spam detection using Random Subspace, Decorate, and J48 *ML* algorithm.

Features used in [2], [10], [11], [12], [13] can be faked or cheated by posting more tweets, purchasing more followers, or mixing normal tweets with spam tweets [9]. As a result, some exiting work [9], [16] presented robust feature that rely on social graph to avoid fabrication of features. In [16] they extracted connectivity and distance among its receiver and sender to estimate whether a tweet is spam or not. This updating process aided in improving the classification accuracy of several classifier. In [9], presented a more efficient features such as Bidirectional Links Ratio, Betweenness Centrality, Local Clustering Coefficient, and attend better performance than [2] classier approach.

In [19] and [25], they have considered embedded links for identifying spam. In [26] multiple URL feature such as path tokens, query parameters, and domain tokens along with some feature form the domain information, DNS information, and landing page. Further, in [25] they have analyzed the properties of Correlated link Redirect Chain, and further obtained relevant features, like Relative number of different initial URLs, URL redirect chain length, etc. These features showed their superior performance when performing classification of spam. However, none of the above approach have addressed the "Spam Drift" problem. As a result, the state-of-the-art *ML* based classifier model accuracy is reduced with respect to time. Since spammers are changing methods to escape being identified. In [27] came up with model to scape "Spam Drift" using Posting Time model and Language Model, for each subscriber. However, their method fails to identify spamming accounts created by spammers falsely and they can only identify only user account was compromised or not.

To address [14] presented a model that learns from the unlabeled twitter tweets, can address "Spam Drift" problem, and detect Twitter spam. They attained better result than [2], [10], [11], [12], and [13]. However, the accuracy of their classification model degrades as day's progress. To overcome the research problem, this work adopts similar architecture as [14] to address Spam drift problem. Further, a novel Adaptive K-Nearest Centroid Neighbor (*AKNCN*) is presented in next section below.

## III. ADAPTIVE K-NEAREST CENTROID NEIGHBOUR

This work presents an Adaptive K-Nearest Centroid Neighbor (*AKNCN*) Classifier to detect twitter spam on Social Networks which works on the basis of Nearest Centroid Neighbor (*NCN*) and K-Nearest Centroid Neighbor (*KNCN*) Algorithms. The basic idea of *NCN* is to distribute the neighbors as geometrically as possible around the query objects along with placing them close to the query objects. Suppose for the query object *x*, *NCN* should be subjected to the two constraints: The Distance Criterion - which says that the centroid neighbors should be close to *x* as much as possible and the Symmetry Criterion – which says that the centroid neighbors should be placed around *x* as homogeneously as possible. The centroid of a set of points *X* where $X = \{x_1, x_2, \ldots\ldots, x_r\}$ is defined as

$$x_r^c = \frac{1}{r}\sum_{i=1}^{r} x_i$$

*b) Architecture of twitter spam detection model using adaptive k-Nearest Centroid Neighbor Classifier:*

In contrast with K-Nearest Neighbor (*KNN*), the *KNCN* predicts the class label of the query object in terms of both the proximity and symmetrical distribution of the neighbor by the Majority Voting.

Under certain circumstances, we can combine different classification rules in such a way that it produces a classifier that is superior to any of the individual rules. Each value of *X*, classify to the classes that receives the largest number of votes or classifications, such family of classifiers are called as Majority Vote Classifier. And the process is known as Majority Voting.

*KNCN* algorithm first calculates the distance of training samples in each class *C* to the query x, next it finds the first nearest centroid neighbor of x in each class *C* and then it searches k-nearest centroid neighbor of x except the first one on the class C. It then calculates the centroid in the set *S* for *x* and the distance between them. And lastly it assigns the class label to majority voting. This paper explains how Twitter Spam Drift Problem can be addressed by using Adaptive K-Nearest Centroid Neighbor algorithm. Firstly, the architecture of proposed twitter spam detection model using *AKNCN* classification is described. Secondly, feature extraction process for training *AKNCN*. Lastly, *AKNCN* Classifier construction for detecting twitter spam is shown.

This section defines the working methodology of proposed spam detection model using Adaptive K-Nearest Centroid Neighbor Classifier. Fig. 1 depicts the working steps involved in building a machine learning based twitter spam detection classifier model. Before initializing classification process, a classifier that possess the knowledge outline must be trained with labeled twitter data. Then, the classification approach gains knowledge pattern of these training labelled data or feature, which is then utilized to classify future tweet. The entire methodology is composed of feature extraction (i.e., the training phase) and classification (i.e., testing phase) which is discussed in below sections. The architecture of proposed twitter spam detection model using *AKNCN* classification is shown in Fig. 1 below.

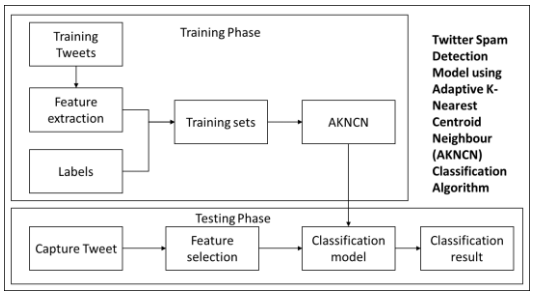


*Fig. 1. Architecture of Twitter Spam Detection Model using* AKNCN.

*b) Feature extraction for training classifier:*

For building Adaptive K-Nearest Centroid Neighbor Classifier, firstly, a neighbor should be close to a feature data as possible and at the same time, the neighbors should be around that feature data as conceivable. Then, another scenario is that the resultant of the initial step is in the limiting circumstance. But in real-world conditions, the geometrical placement can be more significant than actual distances to properly represent a feature by its neighborhood. As the Nearest Neighborhood (*NN*) considers only the initial characteristics, if the neighborhood in the training feature set is not spatially same, the nearest neighbors might not be positioned correspondingly around the feature data. It is stated here that considering finite feature set is considered along with the use of local distance measure which helps in enhancing the behavior or performance of classifier [28]. Therefore, to address the above defined problem, this work presents a spam detection classifier model that makes use of novel neighborhood classification, namely, Adjacent Neighborhood (*AN*) of a feature, i.e., building an efficient nearest neighborhood model that considers not only nearness but also considers the spatial distribution corresponding to that feature.

It must be noted that there is no theoretical model to justify the use of close to (neighbor around) a feature rather than of nearest neighbors, this aid in special circumstances finite sample case., in which classifier model do not entirely describes the distance used or the underlying statistics which shows some detrimental characteristics.

The *AN* is also called as Nearest Centroid Neighborhood (*NCN*) [29]. Let $q$ be a point whose $l$ neighbor must be identified in a set of training set (points) which is represented as follows

$$\mathcal{Y} = \{y_1, y_2, \dots, y_o\}. \quad (1)$$

These $l$ neighbors are such position that their centroid is as close to $q$ as probable and similarly, they should be as close to $q$ as probable. These are satisfied using iterative approach as follows

Firstly, the initial neighbor of $q$ should be its nearest neighbor $r_1$. Then the $j^{th}$ neighbor, $r_j$, $j \geq 2$, is such that the formerly chosen neighbors and centroid of this, $r_1, r_2, \dots, r_j$ is the nearest to $q$. This assumption address in building neighborhood model that takes into account both spatial distribution and closeness because of centroid measure.

Alternatively, *AN* can be derived from information obtained from geometrical structure [30], which is used here to obtain closeness graph. Let consider a set of points in $\mathcal{S}^e$ as follows

$$Y = \{y_1, y_2, \dots, y_o\}. \quad (2)$$

The closeness graph is represented as follows

$$\mathcal{H} = (\mathcal{W}, \mathcal{F}) \quad (3)$$

which is an undirected (directionless) graph with a set of edges $\mathcal{F}$, and set of vertices $\mathcal{W} = Y$, such that $(y_j, y_k) \in \mathcal{F}$ considering if $y_j$ and $y_k$ satisfy some neighborhood relation. Considering the case, $y_j$ and $y_k$ are graph neighbors. Subsequently, the Graph neighborhood of a feature data can be defined as the combination of its entire graph neighbors.

Let $e(*,*)$be the Euclidean distance in $\mathcal{S}^e$. The Proximity Graph (*PG*) can be described as follows

$$(y_j, y_k) \in \mathcal{F} \leftrightarrow e^2(y_j, y_k) \leq e^2(y_j, y_l) + e^2(y_k, y_l)\ \forall y_l \in Y, l \neq j, k. \quad (4)$$

Therefore, $y_j$ and $y_k$ are considered to be Graph Neighborhood (*GN*). In other terms, two points are *GN*s if there is no other point from $Y$ whose diameter is the distance among them and is falling in the hypersphere centered at their center point. Similarly, the set of edges in the Adaptive Neighborhood Graph (*ANG*) can be described as follows

$$(y_j, y_k) \in \mathcal{F} \leftrightarrow e(y_j, y_k) \leq \max e(y_j, y_l), e(y_k, y_l), \forall y_l \in Y, l \neq j, k. \quad (5)$$

Its geometric representation is based on concave-convex area bounded by two circular arcs [31], which can be described as the disjoint intersection among two hyperspheres centered at $y_j$ and $y_k$ and whose radius are equivalent with distance among them, and if their concave-convex area does not hold other point from $Y$, then those two points are considered to be adaptive neighbor. From Eq. (4) and (5), it can be stated that, considering those two points are graph neighbors, if there is no other point that falls inside a given area of impact among hypersphere in the Proximity graph [30] and a concave-convex area bounded by two circular arcs in the adjacent neighboring graph. Therefore, it is likely to fully enclose a feature by means of its *GN*.

*Classifier Construction:*

This section presented a non-parametric based classifier model which is based on method of computing the class of a feature set from its neighbors, by considering a way a neighborhood permits to check near and appropriately small around the feature set. This helps the model around the feature set to take part in process of classification. This is done by utilizing the Adjacent Neighborhood definition described in this work (i.e., the graph neighborhood and nearest centroid neighborhood) which is used here to construct graph neighbors and *KNCN* rules, respectively. From this, the objective of this work is to detect spam on Twitter by applying *AN* in the classification model to obtain an alternative information rather than just obtaining the means of nearest neighborhood for enhancing reliability of classification considering real-time scenario. Therefore, the classification is modeled to perform decision of class membership of the respective feature set after obtaining the spatial distribution of twitter spam model around that feature set. This shows that an input feature set is classified using both nearest neighbors and also utilize how the model is placed around it. Considering this *GN* and nearest centroid neighborhood method is utilized to attain an enhanced non-parametric twitter spam detection classifier that considers a relatively close and number of feature around, rather than just close to feature set to compute to which class it belongs to.

Let consider a set of twitter spam detection function $Y = \{y_1, y_2, \dots, y_o\}$ with $\mathbb{L}$ number of classes, and a new feature data $q$. let us describe the novel adaptive k-nearest centroid neighbor (*AKNCN*) classification rule as follows

Firstly, establish the $k$ nearest centroid neighbors (*KNCN*) of $q$, as follows

$$Y^q = \{y_1^q, y_2^q, \dots, y_l^q\} \ where\ l \leq o. \quad (6)$$

Secondly, assign to $q$ the class with higher weights from its *KNCN*'s in the set $Y^q$ (addresses ties arbitrarily). An important thing to be seen here is that the Graph Neighborhood rule, considering proximity graph and adaptive neighborhood graph, we have two varied but similar proximity graph neighborhood and the Adaptive neighborhood graph, neighborhood rules, respectively. From, experiment analysis, our classifier chose the most effective adjacent neighborhood based on features selection of each problem (i.e., dimensionality of the feature space) which aids in achieving better accuracy spam detection. Further, the accuracy and performance of *AKNCN* classifier can be improved by using the model presented in the next section.

f) AKNCN based Spam Detection Algorithm and Working Model to address spam drift problem:

This section presents proposed algorithm and its description to detect spam efficiently and address spam drift problem. The spam behavior changes with respect to time. As a result, affect performance of existing classifier due to spam drift problem. Therefore, it is important study distribution of data considering different period. The distribution can be computed using distance distribution methods [19], [26], and [32] which can be described as follows

$$\mathcal{D}_{lm}(\mathbb{P}\|\mathbb{Q}) = \sum_{j} \mathbb{P}(j) \log \frac{\mathbb{P}(j)}{\mathbb{Q}(j)}. \quad (7)$$

It is utilized for obtaining a comparative analysis of probability distributions. This work maps the data point into distribution parameter. Let consider a multi-set spam $Y = \{y_1, y_2, \ldots, y_o\}$ from a finite set $F$ composed of feature parameter, and represent $O(y|Y)$ the number of appearances of $y \in Y$, thus the relative ratio of each $y$ is represented as follows

$$\mathbb{P}_Y = \frac{O(y|Y)}{o}. \quad (8)$$

Therefore, the distance among successive days twitter tweets, $\mathcal{D}_1$and $\mathcal{D}_2$ is computed as follows

$$\mathcal{D}(\mathcal{D}_1\|\mathcal{D}_2) = \sum_{y\in F} \mathbb{P}_{\mathcal{D}_1}(y) log \frac{\mathbb{P}_{\mathcal{D}_1}(y)}{\mathbb{P}_{\mathcal{D}_2}(y)} \quad (9)$$

This work compute distance distribution (Eq. (9)) of each feature of non-spam and spam data considering successive days. The higher the value, the more complex the distribution are. From this the distance distribution values, we can obtain the distribution of spam tweets features which is changing rapidly with respect to time. Further, for non-spam data this work assumes there is no changes in distribution. However, the distribution of training sample is not changed/modified. Since the knowledge base that learns from unmodified training sample is not modified while being utilized for classifying upcoming tweets, as a result, accuracy of classification model will be affected. For addressing problems of collecting modified data to update classifier model is a key factor. By collecting such unlabeled incoming twitter tweets and address spam drift problem, this work presents a model *AKNCN* which composed of two element such as to learn from detected spam tweets and further learn from manually labeled data. The algorithm to address spam drift problem is presented in Algorithm 1.

**Algorithm 1: Proposed *AKNCN* based twitter spam detection model for addressing spam drift problem**

**Input:** Labeled / classified twitter training set $\{\alpha_1, \ldots, \alpha_2\}$ unclassified twitter tweets $U_{ulbl}$, a binary *AKNCN* classifier model $\beta$,

**Output:** Manually classified/labeled chosen twitter tweets $u_n$.

**Step 1: Start**

**Step 2:** $U_{lbl} \leftarrow \bigcup_{j=1}^{o} \alpha_j$

**Step 3:** $\mathbb{C} \leftarrow \beta: U_{lbl}$

**Step 4:** $U_{S'} + U_{S''} \leftarrow U_{ulbl}$

**Step 5:** $U_R \leftarrow U_{lbl} + U_{S'}$

**Step 6:** $\mathbb{C} \leftarrow \beta: U_R$

**Step 7:** $V \leftarrow \emptyset$

**Step 8: For** $j = 0$ **to** $l$ **do**

**Step 9:** **if** $V_j$ meets the selection condition $T$ **then**

**Step 10:** $V \leftarrow (V \cup V_j)$

**Step 11:** **end if**

**Step 12: End For**

**Step 13:** $U_n \leftarrow \emptyset$

**Step 14: For** $j = 1$ **to** $l$ **do**

**Step 15:** manually classifies each $v_j$

**Step 16:** $U_n \leftarrow (U_n \cup v_j)$

**Step 17: End For**

**Step 18: Stop**

The algorithm takes labelled twitter set $\{\alpha_1, \ldots, \alpha_2\}$ unclassified twitter tweets $U_{ulbl}$, a binary AKNCN classifier model $\beta$ as input and obtain an output of manually labeled chosen tweets $u_n$. In step 2, we initialize labeled training tweet data $U_{lbl}$ . In step 3, using $\beta$ we construct a classification model $\mathbb{C}$ from $U_{lbl}$. In step 4, the unlabeled data $U_{ulbl}$ is classified as spam $U_{S'}$ and non-spam $U_{S''}$. In step 5, the spam tweets $U_{S'}$ classified by classification model $\mathbb{C}$ are grouped in to labeled data $U_{lbl}$. In step 6, utilizing $U_R$ the classification model $\mathbb{C}$ is further retrained. In step 7, we establish the freshly coming twitter data's fitness for selection process. In step 8 to 12, we obtain tweets that meets selection condition $T$. In step 13 to 17, we manually label or classify each tweet data $v_j$. In this way the training data is updated. This aids in addressing spam drift problems and attaining better accuracy performance which is experimentally proven in below section.

## IV. RESULT ANALYSIS

In this section we present experiment analysis of proposed spam detection model performance using *AKNCN* classifier over exiting classifier model [14].

The system requirement for experiment analysis is Windows 10 enterprises edition, Intel Pentium I-7 class Quad core processor, 16 GB RAM, and NVIDIA CUDA enabled GPU. The *AKNCN* model is implemented using MATLAB 16b and above and Python 3. The performance is evaluated in terms of Accuracy, F-Measure and Detection Rate. This work considers [14] which is the comparison paper. Since it attains better performance than state-of-art model [2], [10], [11], [12], and [13]. This work uses Accuracy, F-measure and Detection Rate to evaluated performance. The Accuracy $A$ is calculated as follows

*a)*

$$A = \frac{T^P * T^N}{T^P + T^N + F^P + F^N} \quad (7)$$

where $T^P$ is the True Positive, $T^N$is the True Negative, $F^P$ is the False positive, and $F^N$ is the False Negative. The F-measure $F$ is calculated as follows

$$F = \frac{2 * P^r * R^c}{P^r + R^c} \quad (8)$$

where $P^r$ depicts precision, $R^c$ recall. Then, the Detection Rate $DR$ is computed as follows

$$DR = \frac{T^P}{T^P + F^N} \quad (9)$$

| Experiment Id (Dataset) | Sampling Frequency | Number of tweets with spam | Number of tweets without spam | Accuracy | F-Measure | Detection rate |
|---|---|---|---|---|---|---|
| **1** | Continuous | 5000 | 5000 | 0.904 | 0.91 | 0.93 |
| **2** | Continuous | 5000 | 95000 | 0.978 | 0.93 | 0.91 |
| **3** | Noncontinuous | 5000 | 5000 | 0.819 | 0.83 | 0.91 |
| **4** | Noncontinuous | 5000 | 95000 | 0.964 | 0.92 | 0.88 |
| **Average** | | | | **0.916** | **0.897** | **0.905** |

TABLE 1: DATASET CONSIDERED FOR EXPERIMENT

**Dataset description:** In Table I, we see that the non-spam and spam ratio is 1:1 in dataset of experiment number 1 and 3, whereas the ratio is 1:19 in experiment 2 and 4. In state-of-the-art model [2], [10], [11], [12], and [13] the dataset is almost uniformly distributed, i.e., the non-spam to spam ratio is nearly equal to 1:1. However, in real-world scenario [4] Twitter Social Network platform has around five percent of spam tweets with overall tweet available. As a result, the evenly distributed tweet set cannot depict or show the actual Twitter Social Network in real-world scenario. To represent real-world Twitter scenario, this work considers the dataset used by researcher in [14] which has a spam ratio of 1:19 as shown experiment number 2 and 4 in Table I. All four-experiment dataset are obtained arbitrarily from the whole 600 million tweets. However, the dataset is segmented into two classes based on the sampling frequency. Experiment 1 and 2 are arbitrarily chosen from the entire twitter dataset, but the twitter data was transmitted in certain continuous instance. Similarly, Experiment 3 and 4, the tweets were transmitted completely independent of each other.

*b)*

*Twitter Spam detection performance evaluation considering varied experiments:*

This section present performance evaluation of *AKNCN* over exiting model [14] in terms of Accuracy (*A*), F-Measure (*F*) and Detection Rate (*DR*) considering varied experiments. The outcomes attained by proposed AKNCN for twitter spam detection is shown in Table II. The outcome shows the proposed *AKNCN* model attains an average twitter spam detection accuracy of 91.95% considering varied experiments. The exiting model attains an average accuracy of 80% considering varied experiments. This shows the *AKNCN* attain significant performance than the exiting model. Accuracy is improved by 11.95% with the proposed *AKNCN* over existing model. Similarly, the proposed *AKNCN* model attain an average twitter spam detection F-measure of 89.7% considering varied experiments. The exiting model attain an average F-Measure of 82% considering varied experiments. This shows the *AKNCN* attain significant performance than exiting model. F-Measure is improved by 5.7% with the proposed *AKNCN* over existing model. Further, the proposed *AKNCN* model attains an average twitter spam Detection Rate of 90.05% considering varied experiments. The exiting model attains an average detection rate of 88% considering varied experiments. This shows the *AKNCN* attain significant performance over exiting model. Detection Rate is improved by 2.05% with the proposed *AKNCN* over the existing model.

*Twitter Spam detection performance evaluation considering varied labeled training sample:*

This section presents evaluation of *AKNCN* over exiting model [14] in terms of Accuracy $(A)$, Detection Rate $(DR)$ and F-measure $(F)$ considering varied labeled training samples.

For experiment analysis we had considered continuous twitter training sample data. The Accuracy performance attained by proposed *AKNCN* and existing model for twitter spam detection is shown in Fig. 2. The outcome shows that the *AKNCN* model attains significant accuracy performance improvement over exiting model considering all experiment. An average Accuracy performance improvement of 8.803% is attained by *AKNCN* over exiting model. Similarly, The F-Measure performance attained by proposed *AKNCN* and existing model for twitter spam detection is shown in Fig. 3. The outcome shows that the *AKNCN* model attain significant F-Measure performance improvement over exiting model considering all experiment. An average F-Measure performance improvement of 8.76% is attained by *AKNCN* over exiting model. Further, the Detection Rate performance attained by proposed *AKNCN* and existing model for twitter spam detection is shown in Fig. 4. The outcome shows that the *AKNCN* model attain significant detection rate performance improvement over exiting model considering all experiment. An average detection rate performance improvement of 5.632% is attained by *AKNCN* over exiting model. The overall result attained by proposed *AKNCN* shows it can detect spam efficiently and accurately with respect to time.

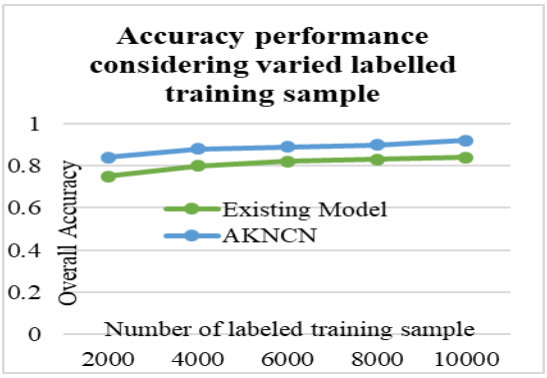


Fig. 2. Accuracy performance evaluation considering varied labeled training sample

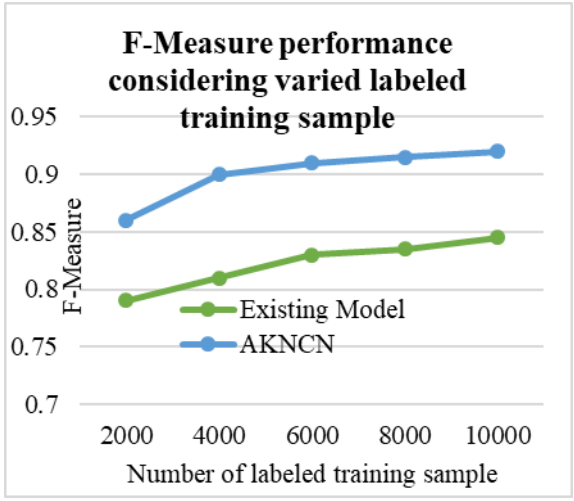


Fig. 3. F-Measure performance evaluation considering varied labeled training sample

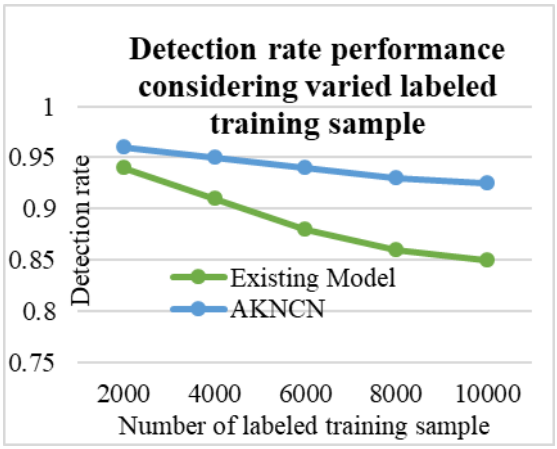


Fig. 4. Detection rate performance evaluation considering varied labeled training sample

## V. RESULT ANALYSIS

In this work, we first identified the "Spam Drift" issues for detecting spam in Twitter social network using in statistical features. Secondly, we identified that the accuracy of classification model need to be modeled. To solve we presented a non-parametric Adaptive K-Nearest Centroid Neighbor (*AKNCN*) classifier model which can learn efficiently from most recent tweet labeled and unlabeled tweet. Further, used one million spam and non-spam tweets for training *AKNCN* for meeting real-world requirement. Experiment are conducted considering different experiment sets and different labeled training sets. The result shows *AKNCN* attain an average performance improvement of 11.95%, 5.7% and 2.05% in term of Accuracy $(A)$, F-Measure $(F)$ and Detection Rate $(DR)$ respectively, over exiting model considering varied experiment sets. Similarly, *AKNCN* attain an average performance improvement of 8.803%, 8.76%, and 5.632% in term of Accuracy $(A)$, F-Measure $(F)$ and Detection Rate $(DR)$ respectively, over exiting model by considering varied labeled training sets. The overall result attained shows that *AKNCN* attains significant performance in terms of Detection Rate $(DR)$, Accuracy $(A)$, and F-Measure $(F)$ in real-world scenarios.

## References

1. A. Greig. (2013). Twitter Overtakes Facebook as the Most Popular Social Network for Teens,According to Study, DailyMail, accessed on Aug. 1, 2015. [Online]. Available: http://www.dailymail.co.uk/news/article-2475591/Twitter-overtakes-Facebook-popular-socialnetwork-teens-according-study.html.
2. F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammer on twitter," in Proc. 7th Annu. Collaboration, Electron. Messaging, Anti-Abuse Spam Conf., p. 12, 2010.
3. C. Pash. (2014). The lure of Naked Hollywood Star Photos Sent the Internet into Meltdown in New Zealand, Bus. Insider, accessed on Aug. 1, 2015 [Online]. Available: http://www.businessinsider.com.au/the-lure-of-naked-hollywood-star photos-sent-the-internet-into-meltdown-in-new-zealand-2014-9.
4. J. Oliver, P. Pajares, C. Ke, C. Chen, and Y. Xiang, "An in-depth analysis of abuse on twitter," Trend Micro, Irving, TX, USA, Tech. Rep., 2014.
5. R. Jeyaraman. (2014). Fighting Spam With Botmaker, Twitter, accessed on Aug. 1, 2015. [Online]. Available: https://blog.twitter.com/2014/fighting-spam-with-botmaker.

6. C. Grier, K. Thomas, V. Paxson, and M. Zhang, "@spam: The underground on 140 characters or less," in Proc. 17th ACM Conf. Comput. Commun. Security, pp. 27–37, 2010.
7. K. Thomas, C. Grier, D. Song, and V. Paxson, "Suspended accounts in retrospect: An analysis of twitter spam," in Proc. ACM SIGCOMM Conf. Internet Meas. Cof., pp. 243–258, 2011.
8. C. Yang, R. Harkreader, J. Zhang, S. Shin, and G. Gu, "Analyzing spammers' social networks for fun and profit: A case study of cyber criminal ecosystem on twitter," in Proc. 21st Int. Conf. World Wide Web, pp. 71–80, 2012.
9. C. Yang, R. Harkreader, and G. Gu, "Empirical evaluation and new design for fighting evolving twitter spammers," IEEE Trans. Inf.Forensics Security, vol. 8, no. 8, pp. 1280–1293, 2013.
10. M. Fazil and M. Abulaish, "A Hybrid Approach for Detecting Automated Spammers in Twitter," in IEEE Transactions on Information Forensics and Security, vol. 13, no. 11, pp. 2707-2719, 2018.
11. M. Jiang, P. Cui and C. Faloutsos, "Suspicious Behavior Detection: Current Trends and Future Directions," in IEEE Intelligent Systems, vol. 31, no. 1, pp. 31-39, 2016.
12. G. Lin, N. Sun, S. Nepal, J. Zhang, Y. Xiang and H. Hassan, "Statistical Twitter Spam Detection Demystified: Performance, Stability and Scalability," in IEEE Access, vol. 5, pp. 11142-11154, 2017.
13. S. Sedhai and A. Sun, "Semi-Supervised Spam Detection in Twitter Stream," in IEEE Transactions on Computational Social Systems, vol. 5, no. 1, pp. 169-175, 2018.
14. C. Chen, Y. Wang, J. Zhang, Y. Xiang, W. Zhou and G. Min, "Statistical Features-Based Real-Time Detection of Drifted Twitter Spam," in IEEE Transactions on Information Forensics and Security, vol. 12, no. 4, pp. 914-925, 2017.
15. J. Zhang, R. Zhang, Y. Zhang and G. Yan, "The Rise of Social Botnets: Attacks and Countermeasures," in IEEE Transactions on Dependable and Secure Computing. doi: 10.1109/TDSC.2016.2641441, 2016.
16. J. Song, S. Lee, and J. Kim, "Spam filtering in twitter using senderreceiver relationship," in Proc. 14th Int. Conf. Recent Adv. Intrusion Detection, pp. 301–317, 2011.
17. T. Dasu, S. Krishnan, S. Venkatasubramanian, and K. Yi, "An information-theoretic approach to detecting changes in multidimensional data streams," in Proc. Symp. Interface Statist., Comput. Sci., Appl., pp. 1–24, 2006.
18. J. Gama, I. Žliobait˙e, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," J. ACM Comput. Surv., vol. 46, no. 4, p. 44, 2014.
19. J. M. Romo and L. Araujo, "Detecting malicious tweets in trending topics using a statistical analysis of language," Expert Syst. Appl., vol. 40, no. 8, p. 2992–3000, 2013.
20. M. Bhaduri, J. Zhan, C. Chiu and F. Zhan, "A Novel Online and Non-Parametric Approach for Drift Detection in Big Data," in IEEE Access, vol. 5, pp. 15883-15892, 2017.
21. S. Yardi, D. Romero, G. Schoenebeck, and D. Boyd, "Detecting spam in a twitter network," First Monday, vol. 15, nos. 1–4, pp. 1–13, Jan. 2010.
22. H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in Proc. 19th Int. Conf. World Wide Web, pp. 591–600, 2010.
23. G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in Proc. 26th Annu. Comput. Security Appl. Conf., pp. 1–9, 2010.
24. K. Lee, J. Caverlee, and S. Webb, "Uncovering social spammers: Social honeypots + machine learning," in Proc. 33rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., pp. 435–442, 2010.
25. S. Lee and J. Kim, "Warningbird: A near real-time detection system for suspicious URLs in twitter stream," IEEE Trans. Depend. Sec. Comput., vol. 10, no. 3, pp. 183–195, May 2013.
26. K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, "Design and evaluation of a real-time URL spam filtering service," in Proc. IEEE Symp. Security Privacy, 2011, pp. 447–462, 2011.
27. M. Egele, G. Stringhini, C. Kruegel, and G. Vigna, "Compa: Detecting compromised accounts on social networks," in Proc. Annu. Netw. Distrib. Syst. Security Symp., 2013.
28. Shirkhorshidi AS, Aghabozorgi S, Wah TY (2015) A Comparison Study on Similarity and Dissimilarity Measures in Clustering Continuous Data. PLoS ONE 10(12): e0144059. https://doi.org/10.1371/journal.pone.0144059.
29. J. Gou, Y. Zhang, L. Du, and T. Xiong, ``A local mean-based k-nearest centroid neighbor classifier," Comput. J., vol. 55, no. 9, pp. 1058-1071, 2012.
30. C. Norrenbrock, "Percolation threshold on planar Euclidean Gabriel graphs," C. Eur. Phys. J. B (2016) 89: 111. https://doi.org/10.1140/epjb/e2016-60728-0, 2016.
31. Jean Cardinal, Sebastien Collette, Stefan Langerman "EmptyRegionGraphs," http://www.ulb.ac.be/di/algo/secollet/papers/ccl06b.pdf, last accessed in July 10, 2018.