Classify SMS in Mobile System Using IBK and K-Star

SMS Analysis for Future Security and User Convenience

[1]Neetu Gupta, [2]Ms. Sonal Arora
[1]Student, [2]Assistant Professor,
[1] Department of Computer Science and Engineering,
[1]DPGITM (MDU University), Gurgaon, India.

Abstract: The intense growth of smart mobile phones and users has contributed to the expansion of online or offline Instant Messaging and SMS usage as an alternate way of Transaction and communication. Along with the faith they instinctively have in their devices, makes this kind of messages a congenial environment for spammers. In fact, reports distinctly shows that volume of spam over Instant Messaging and SMS is rapidly increasing year by year. This represents a challenging problem for classical filtering methods these days. Smishing this term represents a phishing in SMS/Messages called as SMS-phishing is a cyber-security attack, which utilizes Short Message Service (SMS) to steal personal data/credentials of mobile users. The faith level of mobile users on their smartphones has attracted attackers to perform various mobile security attacks like SMS-Phishing. In this paper, we implement the SMS-Case-based data mining classification approach to classify them subpart of SMS category by detecting of legitimate, Illegitimate/Smishing messages and these classified messages will further categorize in three parts Primary, Other, Fake. In this research paper IBL and K-Start is used to classify the SMS and moreover we will analyses the classified data model in detail using Weka tool. During the lockdown period due to Pandemic affected by COVID19 SMS- Phishing becomes more active and the attackers send fraud messages to the mobile users intensively

Keywords:
SMS, Message Analysis, Smishing, Illegitimate, SMS Classification, Data Mining, Cases, Phishing, Short Messages,

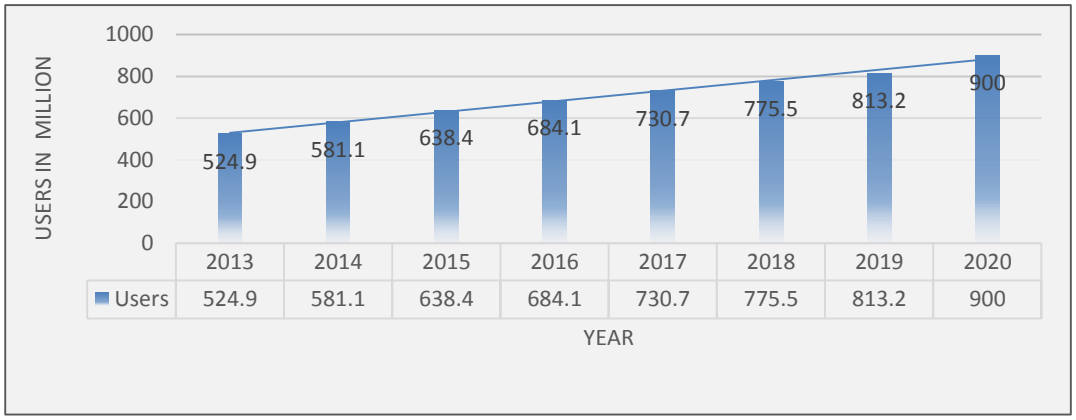
Benefits:
Using this proposed solution giant organization can enhance their existing mobile Message/SMS application to the user security and convenience.

Why this approach/ need of this solution:
As SMS/Instant messages has become alternative way of transaction and communication, so there is a need to make SMS services more secure and classified.

I. INTRODUCTION

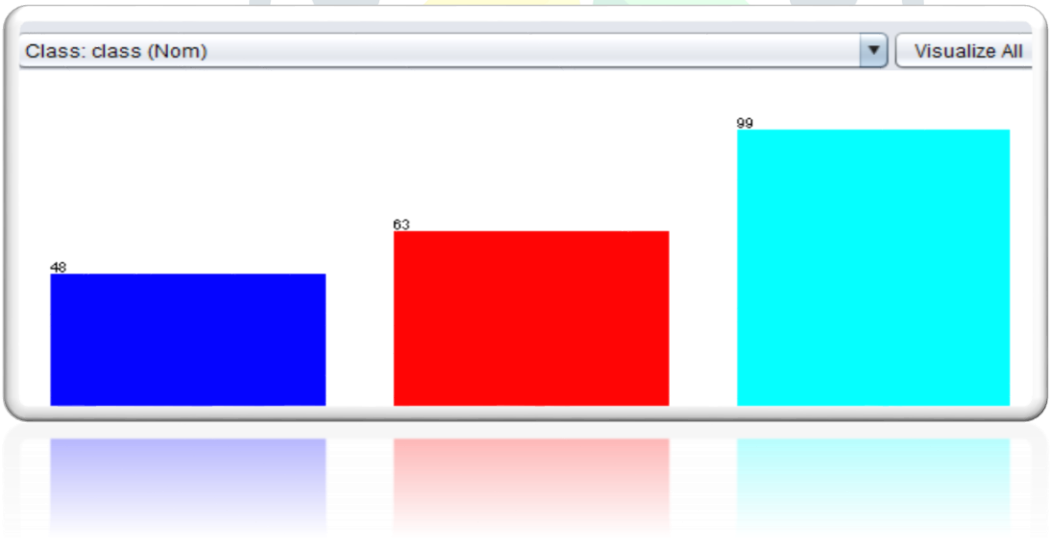
During the lockdown period due to Pandemic affected by COVID19 SMS- Phishing becomes more active and the attackers send fraud messages to the mobile users intensively. Not only attackers in fact the money frauds active a lot. As we all know Short text messaging has become a mean of communication these days. IM/SMS are clearly the leading way of communication. In fact, estimation says that near about 1.5 billion messages are sent a day by considering just SMS. The popularity of (SMS) has been growing during the last decade. The intense growth of smart mobile phones has contributed to the growth of online or offline SMS usage as an alternative way of Transaction and communication. For businesses text messages are simpler than even emails this is often because while 96% of mobile users read their SMS by the top of the day about 85% of the emails remain unopened SMS comparison 2018 [3-4]. Example of a fraud/fake spam text message is like ‘‘Hurray!! CONGRATULATIONS!!: YOUR MOBILE NO HAVE WON 1000,000 IN YOUR ACCOUNT— MOBILE DRAW USA, TO CLAIM PRIZE SEND BANK DETAILS, NAME, AGE, etc. Many more messages patterns we can see on our mobile system. So, this paper is further extension of my review paper, basically some research work using some algorithms and their comparative analysis will be discussed. The approach will be data mining technique is classification and algorithm used Instance Based Learning and a comparative analysis with K-Star algo a LAZY approach.

FIG1: MOBILE PHONE USER ANALYSIS IN INDIA [2]



Now a day's mobile security may be a major concern because attackers have diverted their mind from Computers to smartphones due to technology growth. Moreover, people are more attracted towards smartphones because it may be a portable and multi-functioning device, Smartphones are more popular now a days as compared to laptops due to their small screen size, lower cost, and portability. consistent with Dimensional Enterprise Mobile Security Survey report and it shows that Smishing attack stands at the second position altogether quite mobile devices attacks [14]. There's two sort of security methods are wont to identify Illegitimate/fake mobile SMS. The primary method is that the Blacklist based method that forestalls the incoming SMS from the fake sources [17]. However, blacklist-based techniques don't cover all the fake sources, as an attacker can buy any mobile number to send the Illegitimate/fake/bogus SMS. The second sort of solution is predicated on the machine learning algorithm during which various features are extracted and compute from the SMS to require appropriate decision. The advantage of the machine learning based technique is that it can detect the fake message coming from any source. Data processing methods help within the feature extraction and finding the relation between them [16]. These approaches identifying hidden knowledge from datasets in terms of Cases and make the choice supported extracted Cases. Human easily understands these Cases. In this paper, some fraud terms data processing classification approach within the prediction of useful/illegitimate/promotional/offers SMS. We've used WEKA tool to classify SMS/Messages for data processing. We study the varied characteristics of text messages thorough then found more than fifteen terms/Cases which may efficiently classify SMS to the subcategory. We then use classification algorithm namely **Instance Based Learning and K-star**. In this, we've also identified the illegitimate/Smishing messages. *The performance of the proposed approach is evaluated, and it achieved quite 99% of true negative rate and 99.9% true positive rate.*

Fig 2: SMS frequency analysis



(Primary/Useful, Other/ Promotional/Offers, Illegitimate/Smishing)

II. REVIEW OF LITERATURE

Over the years, data scientists have proposed several ML models to spot Spam and not Spam. These aren't only for mobile text messages but also email spam and on social network platforms like Facebook Twitter delany et al [23] provided a survey of existing works for filtering spam SMS. They mostly covered articles that relied on conventional machine learning approaches but not deep learning for instance, [24] compared Bayesian classifier with other classification algorithms and located that the previous was better to classify Spam text Messages. Androulidakis et al. [25] proposed another version to clear out Spam messages. Their version became supported the Android OS during which the user's cell manage was wont to filter out the Spam.. The model checked the knowledge of message senders against a previously defined spammer list so when a message came from the users present within the list of spammers it had been treated as spam else not spam zainal et al.

Related Work

This section discusses the various existing mobile Text SMS classification detection techniques. The existing mobile classification and detection techniques divide into following section.

a) User Knowledge/Education Based Scheme

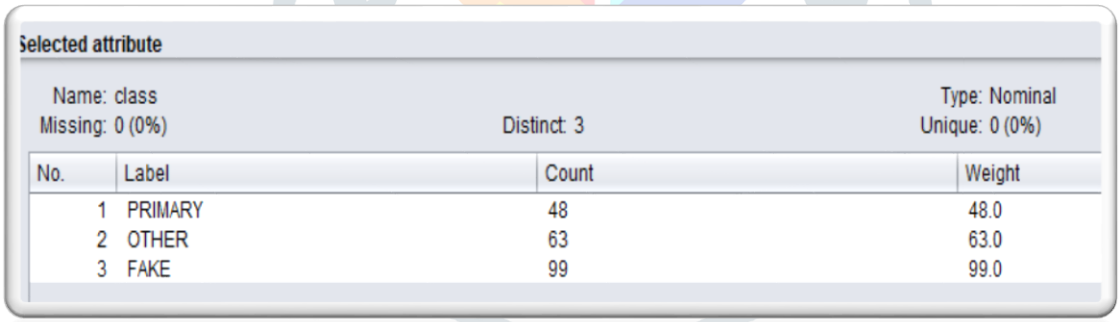
The educational based solutions focus on educating the mobile users about the characteristics of phishing message through training, workshop and awareness programs so that they correctly identify the phishing attack [8]. However, the phishing attack becomes successful due to human flaws and ignorance. This conceptual knowledge may help the users in avoiding phishing attacks.

b) Technical solutions to mitigate mobile phishing attack

The technical solutions are cost-effective and simple to enforce as compare to educational based answers. In this, Amrutkar et al. [9] cautioned mechanism named KAYO, which differentiates among the malicious and genuine cell webpages. It detects mobile malicious pages by measuring 44 mobile features from webpages. Among 44 features, 11 are newly identified mobile specific features. KAYO's 44 feature set is split into four classes namely HTML, mobile specific, URL and JavaScript features. Joo et al. [6] proposed a model 'S-Detector' for detecting Smishing attack. They used Naïve Bayesian Classifier in their system to filter Smishing messages by finding the words used more often in these messages. S-Detector consists of SMS monitor, SMS analyzer, SMS determinant, and Database. Foozy et al. [7] proposed a Rule-based methodology to filter Illegitimate/Smishing messages from spam messages. Authors applied two Rule namely 'winner announcement' and 'marketing advertisement'. They need applied the Bayesian technique in WEKA tool to see the accuracy of Smishing, spam and ham messages. Alfy et al. [15] proposed a spam filtering model for both email and SMS. The proposed technique used 11 features namely presence of URLs, likely spam words, emotion symbols, special characters, gappy words, message metadata, JavaScript code, function words, recipient address, discipline and spam domain. they need evaluated their proposed model with five email and SMS datasets. Within the literature, we will conclude that no single technique exists which will detect illegitimate/Smishing attacks efficiently. Therefore, we'd like a way which will protect the user against Illegitimate/fake/Smishing attacks.

Research Methodology

In this we have discuss our proposed methodology of classifying messages into different categories by using some terms/cases and accordingly we will classify them using WEKA tool by classification algorithms like a lazy approach i.e IBK and K-Star algorithms and will identify the accuracy, further will also use some detection cases to detect illegitimate/Smishing SMS/ Messages.


Selected attribute

Name: class Type: Nominal
Missing: 0 (0%) Distinct: 3 Unique: 0 (0%)

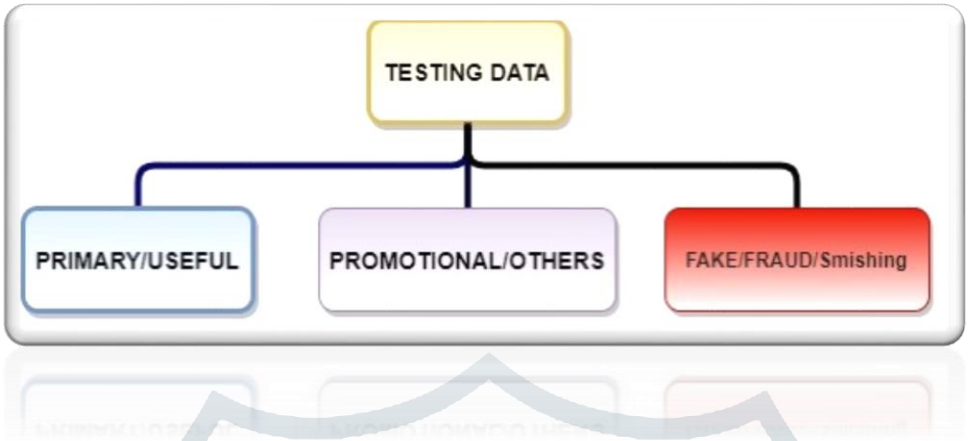
No.	Label	Count	Weight
1	PRIMARY	48	48.0
2	OTHER	63	63.0
3	FAKE	99	99.0

The proposed approach is a model to filter SMS, protects the user from the phishing SMSs by blocking these messages and also implemented system can classify them into different category and delivering only Normal ones to the mobile user instead of making all into a single category it will further filtered into different category like- Primary/Useful, illegitimate/fake, Other/Promotional. The SMS detection is a type of ternary classification problem where a message can be the divide in three categories (i.e., Primary, Illegitimate/fake, Other/Promotional). Illegitimate message is a dangerous spam message that steals personal data/credentials. As per our research and observation, we find the followings characteristics of fraud message:

- *It can have .exe message content in the link form.*
- *SMS having URLs.*
- *Now a days a different format seen like SMS includes.txt files.*
- *It can have any honey coated audio/video content that can trap the user.*
- *Advertising for offers/ Promotions.*
- *It can have the bogus fake links. Advertising something like providing free minutes, etc.*
- *It can have email address or a phone number.*
- *Links can have harmful viruses.*
- *Machine Recorded voice/Self-answering SMS asking the user to subscribe or unsubscribe any service.*
- *Announcing to users as a winner for fake contest and attract him using the prize money.*

- *Intended to spread some fake news.*
- *Message can have link and link have steganography.*
- *Java script contents.*
- *Long SMS can have fake. Etc.*

Fig 4: Testing Data Classification Hierarchy



III. TOOL AND TECHNIQUES

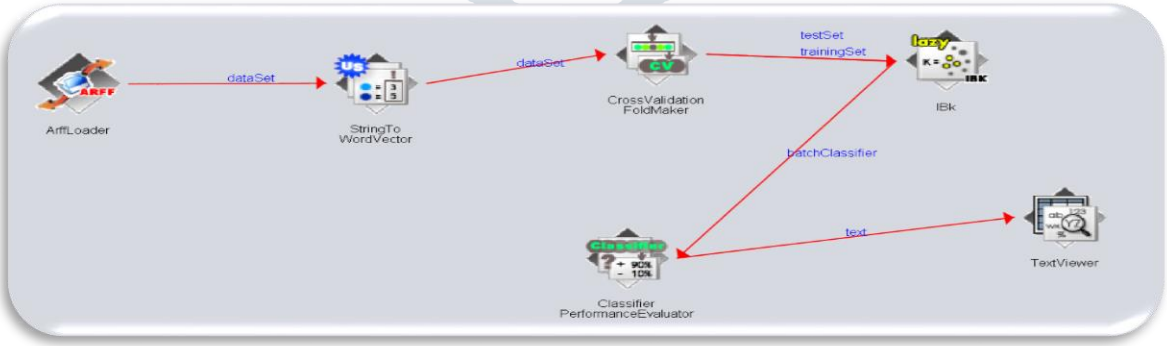
In our research work we used data mining classification techniques to classify the data and accordingly we found accuracy, and result set data, to do so we used WEKA tool, PC configuration having windows 10 core i5 processor 8 GB RAM etc.

Basic Steps will follow the different stages of data mining-

- Data collecting.
- Data Pre-processing.
- Fraud Terms/Case- Extraction.
- Classifier Training, Testing and Validation data sets.

In Weka data mining tool, there is a different way we can analyse data like: Explorer, Experimenter, Knowledge Flow Environment Workbench, simple CLI. Below is the snapshot using Weka Knowledge Flow. Weka Knowledge Flow is the way to analyse data by graphical design approach [30]. Designs are as ArffLoader, String to word vector, Cross Validation fold maker, IBK and K-Star algorithms, Classifier Performance Evaluator TextViewer etc. In ArffLoader we will load the data set which we need to analyse, here in my case my data set is in Text format so we need to convert this to the Word vector format, then here in fig we used Cross Validation Fold maker and then we pass the train set and test set data to the used algorithm and then a Classifier is used and finally Text Viewer is used to display the result.

Fig 5: Knowledge Flow Environment link using IBK:



IV. ALGORITHM ANALYSIS AND DETAILS:

IBL (Instance Based Learning) *[31]-*

The primary output of IBL algorithms is a idea description (or idea). This is a feature that maps times to categories: given an example drawn from the example space, it yields a class, that's the predicted fee for this example's class attribute. An example-based idea description includes a set of stored instances and, possibly, some information regarding their beyond performances for the duration of classification (e.G., their wide variety of accurate and incorrect category predictions). This set of instances can trade after each training instance is

processed. However, IBL algorithms do not assemble extensional idea descriptions. Instead, idea descriptions are decided through how the IBL algorithm's decided on similarity and category functions use the modern-day set of saved times. These capabilities are of the three functions in the following framework that depicture all IBL algorithms:

1. Similarity Function
2. Classification Function
3. Concept Description Updater

Performance Dimensions –

- Generality: This is the elegance of concepts which might be describable through the illustration and learnable by way of the algorithm. We will display that IBL algorithms can pac-learn (Valiant, 1984) any idea whose boundary is a union of a finite variety of closed hyper-curves of finite size.
- Accuracy: This is the concept descriptions' classification accuracy.
- Learning Rate: This is the speed at which classification accuracy increases during training. It is a more useful indicator of the performance of the learning algorithm than is accuracy for finite-sized training sets.
- Incorporation Costs: These are incurred while updating the concept descriptions with a single training instance. They include classification costs.

Storage Requirement: This is the size of the concept description which, for IBL algorithms, is defined as the number of saved instances used for classification decisions

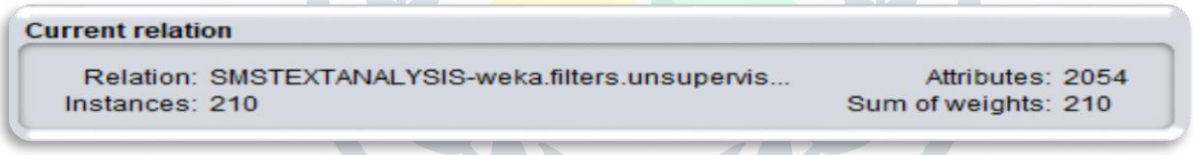
$$\text{Similarity}(x, y) = -\sqrt{\sum_{i=1}^{n} f(x_i, y_i)},$$

Table 1. The IBl algorithm (*CD* = Concept Description).

```
CD ← ∅
for each x ∈ Training Set do
  1. for each y ∈ CD do
       Sim[y] ← Similarity(x, y)
  2. y_max ← some y ∈ CD with maximal Sim[y]
  3. if class(x) = class(y_max)
       then classification ← correct
       else classification ← incorrect
  4. CD ← CD ∪ {x}
```

Summary :

Correctly Classified Instances	***200***	*95.2381 %*
Incorrectly Classified Instances	***10***	*4.7619 %*



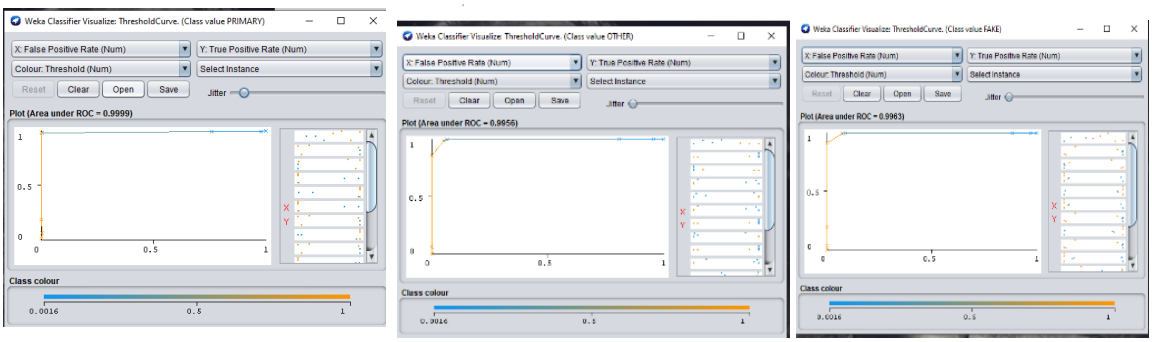
Detailed Accuracy by Class:

TP Rate	*FP Rate*	*Precision*	*Recall*	*F- Measure*	*MCC*	*ROC Area*	*PRC Area*	*Class*	
1.000	0.012	0.960	1.000	0.980	0.974	1.000	0.999	Primary	
0.984	0.054	0.886	0.984	0.932	0.904	0.996	0.983	Other	
0.909	0.000	1.000	0.909	0.952	0.917	0.996	0.993	Fake	
0.952	0.019	0.957	0.952	0.953	0.926	0.997	0.991		***Weighted Avg***

Advantage of IBL:

Instance-primarily based algorithms additionally have several advantages. One advantage of this approach is its simplicity, which allowed us to apply a rigorous analysis to guide our intuition and studies goals. Our evaluation of IB1 led to the improvement of the storage discount algorithm. The IBL paradigm supports quite robust gaining knowledge of algorithms. They can tolerate noise and irrelevant attributes and might represent each probabilistic and overlapping concepts (Aha, 1989b).

Other Measures and Graph Evaluations:



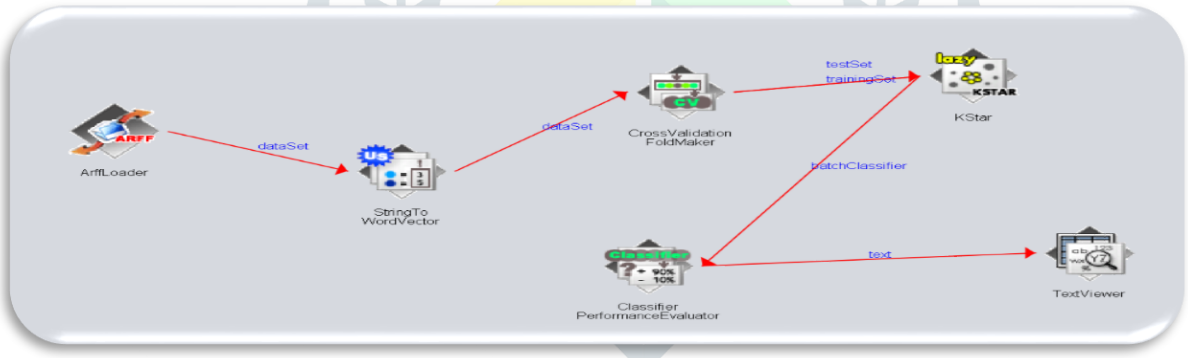
Time Taken :

Time taken to build model: 0.01 seconds
Time taken to test model on training data: 0.04 seconds

***K-Star(K*)*:**

K*[29,33] is an instance-based classifier, that's the category of a test instance is predicated upon the category of these training instances almost like it, as determined by some similarity function. K Star is an instance-based learner that tries to enhance its performance for handling missing values, smoothness problems and both real and symbolic valued attributes. Instance-based (IB) learners, also called Memory-based. It differs from other instance-based learners therein it uses an entropy-based distance function. IB learners are able to learn quickly from a very small dataset; while others like rule induction methods require a reasonable representation of each rule before they can be induced. An instance-based learner can begin to make useful predictions from as little as one example per class. Classification performance often exceeds 75% of the utmost possible after accepting only 25% of an entire data set.

Fig 6: Knowledge Flow Environment link using K-Star:



Specification of K-Star:

Let I be a (possibly infinite) set of instances and T a finite set of transformations on I. Each t ∈ T maps instances to instances: t:I→I. T contains a distinguished member σ (the stop symbol) which for completeness maps instances to themselves (σ(a)=a).

Let P be the set of all prefix codes from T* which are terminated by σ. Members of T* (and so of P) uniquely define a transformation on I:

$$t(a) = t_n(t_{n-1}(...t_1(a)...)) \text{ where } t = t_1,...t_n$$

A probability function p is defined on T*. It satisfies the following properties:

$$0 \le \frac{p(\bar{t}u)}{p(\bar{t})} \le 1$$

$$\sum_{u} p(\bar{t}u) = p(\bar{t})$$

$$p(\Lambda) = 1$$

As a consequence it satisfies the following:

$$\sum_{\bar{t} \in P} p(\bar{t}) = 1$$

The probability function P* is defined as the probability of all paths from instance a to instance b:

$$P^*(b|a) = \sum_{\bar{t} \in P : \bar{t}(a) = b} p(\bar{t})$$

K*: An Instance-based Learner Using an Entropic Distance Measure

$$P^*(b|a) = P^*(i) = \frac{s}{\sqrt{2s - s^2}} \left(\frac{1 - \sqrt{2s - s^2}}{1 - s} \right)^i \text{ where } i = |a - b|$$

and

$$K^*(b|a) = K^*(i) =$$

$$\tfrac{1}{2} \log_2(2s - s^2) - \log_2(s) + i[\log_2(1 - s) - \log_2(1 - \sqrt{2s - s^2})]$$

That is, the distance is proportional to the absolute difference between two instances.

Symbolic Probabilities:

$$P^*(j|i) = \begin{cases} (1-s)p_j & \text{if } i \neq j \\ s + (1-s)p_i & \text{if } i = j \end{cases}$$

***Combining Attributes*:**

To compute a distance between instances with more than one attribute is straightforward.

$D = \sqrt{a^2 + b^2}$ (where a and b are the individual distances on the two co-ordinates)

Missing Values:

$$P^*(?|a) = \sum_{b} \frac{P^*(b|a)}{N}$$

Summary :

Correctly Classified Instances	***200***	*95.2381 %*
Incorrectly Classified Instances	***10***	*4.7619 %*

Detailed Accuracy by Class:

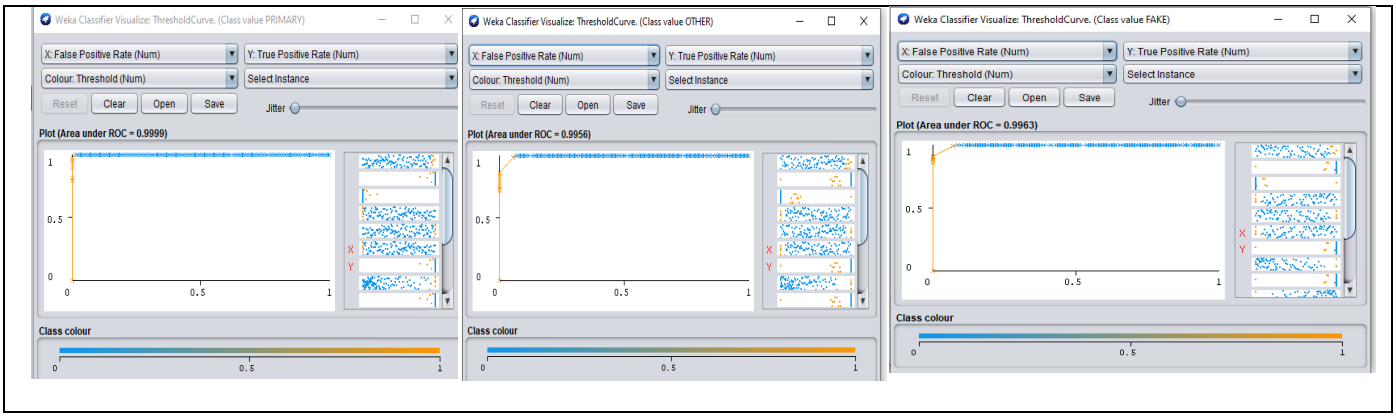
Same as IBL but the time taken is different to build model and test model on training data.

TP Rate	*FP Rate*	*Precision*	*Recall*	*F- Measure*	*MCC*	*ROC Area*	*PRC Area*	*Class*
1.000	0.012	0.960	1.000	0.980	0.974	1.000	0.999	Primary
0.984	0.054	0.886	0.984	0.932	0.904	0.996	0.983	Other
0.909	0.000	1.000	0.909	0.952	0.917	0.996	0.993	Fake
0.952	0.019	0.957	0.952	0.953	0.926	0.997	0.991	***Weighted Avg***

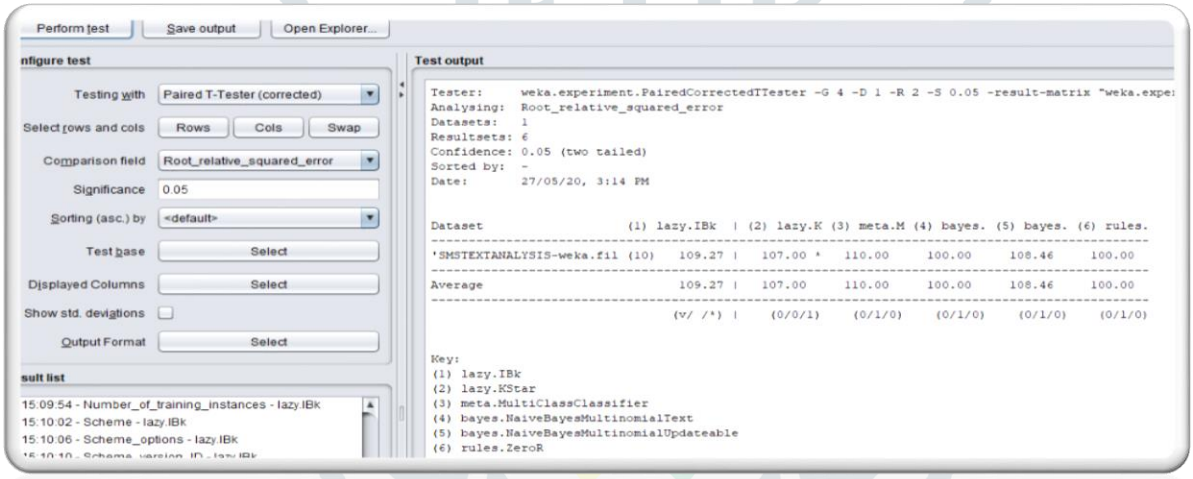
Time Taken:

Time taken to build model	0.00
Time taken to test model on training data: 0.13 seconds	0.13

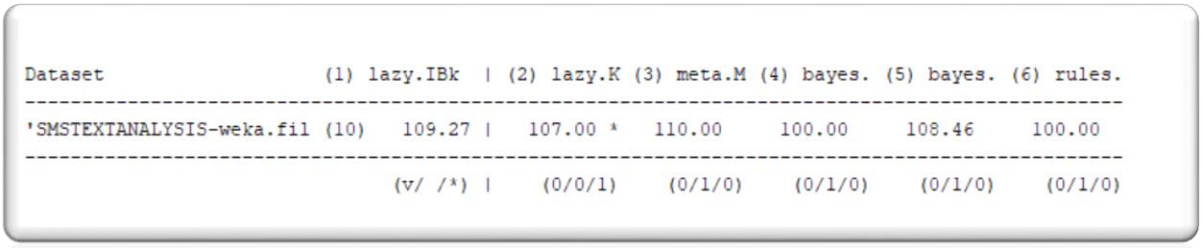
Other Measures and Graph Evaluation: Threshold Curve Visualization for all the classified categories of K- Star.



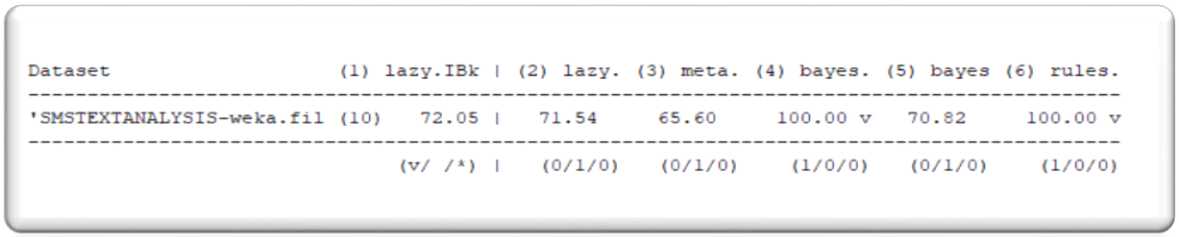
Furthermore, a comparative different algorithms analysis using Experimenter for the same data set:



Root Relative Square Errors of multiple algorithms together:



Relative Absolute Errors of multiple algorithms together:



Dataset	(1) lazy.IBk	(2) lazy.	(3) meta.	(4) bayes.	(5) bayes	(6) rules.
'SMSTEXTANALYSIS-weka.fil (10)	72.05	71.54	65.60	100.00 v	70.82	100.00 v
	(v/ /*)	(0/1/0)	(0/1/0)	(1/0/0)	(0/1/0)	(1/0/0)

V. CONCLUSION AND FUTURE WORK

In our paper we have given an approach to secure SMS for now as well as in future. We had studied a lot with collected message data and made some cases to classify the messages into different category, so that it can be implemented into the existing application by the giant organizations like Google, Microsoft, and Apple not with different application but with the same existing application. Different inbuilt algorithms used by WEKA tool. True positive rate is 99.99%, and FPR is 0.01%.

As there is another application exist but giving the permission to the other application it's also a privacy breach so instead of securing system by other application it will be good to have a functionality in existing one application. Future research can have more data sets collected from different users; we will make more terms/cases through it so that we can have a better security to the users. Also, our planning to make more and more cases accordingly. We are exploring more algorithm and data set as well as cases and characteristics so that we can get better classification accuracy. Also, we will extend this research work using data mining tools like Rapid Miner and for sample and testing purpose we will develop a software, an android based application for the same.

VI. ACKNOWLEDGMENTS:

The whole project of Weka tool was done as part of the WEKA project, at the University of Waikato. We would like to thank the members of the WEKA team, who made such a great Weka tool to analyze data sets. Really a great feature ahead to use for machine learning projects. Throughout my research work the support documentation by Weka team helped us a lot.

VII. REFERENCES

[1] A. K. Jain and B. B. Gupta, A novel approach to protect against phishing attacks at client side using auto-updated white-list. EURASIP Journal on Information Security, 2016(9), 2016.
[2] N. Gupta, S. Arora Review paper SMS Categorization for Future Security and User Convenience http://www.jetir.org/papers/JETIR1908A97.pdf
[3] SMS, C, The real value of sms to businesses, 2018, https://www. smscomparison.co.uk/sms-gateway-uk/2018-statistics/. (Accessed March 2019).
[4] T.A. Almeida, J.M.G. Hidalgo, A. Yamakami, Contributions to the study of sms spam filtering: new collection and results, in: Proceedings of the 11th ACM Symposium on Document Engineering, ACM, 2011, pp. 259–262.
[5] C. Wang, Y. Zhang, X. Chen, Z. Liu, L. Shi, G. Chen, F. Qiu, C. Ying, W. Lu, A behavior-based sms antispam system, IBM J. Res. Dev. 54 (2010) 3–1.
[6] T. Yamakami, Impact from mobile spam mail on mobile internet services, in: International Symposium on Parallel and Distributed Processing and Applications, Springer, 2003, pp. 179–184.
[7] V. Gupta, A. Mehta, A. Goel, U. Dixit, A.C. Pandey, Spam detection using ensemble learning, in: Harmony Search and Nature Inspired Optimization Algorithms, Springer, 2019, pp. 661–668.
[8] Z. Chen, Q. Yan, H. Han, S. Wang, L. Peng, L. Wang, B. Yang, Machine learning based mobile malware detection using highly imbalanced network traffic, Inform. Sci. 433 (2018) 346–364.
[9] C. Amrutkar, Y.S. Kim and P. Traynor, Detecting Mobile Malicious WebPages in Real Time, IEEE Transactions on Mobile Computing (2016)
[10] J.W. Joo, S.Y. Moon, S. Singh and J.H. Park, S-Detector: an enhanced security model for detecting Smishing attack for mobile computing, Telecommunication Systems vol. 66(1), 29–38 (2017).
[11] M. Foozy, C. Feresa, R. Ahmad and M.F. Abdollah, A practical Case based technique by splitting SMS phishing from SMS spam for better accuracy in mobile device, International Review on Computers and Software, vol. 9(10), pp. 1776-1782 (2014).
[12] E. M. El-Alfy and Ali A. AlHasan, Spam filtering framework for multimodal mobile communication based on dendritic cell algorithm, Future Generation Computer Systems, vol. 64, pp. 98-107, (2016).
[13] Symantec Internet Security Threat Report, Available at: http://www.symantec.com/content/en/us/enterprise/other_resources/bistr_main_report_v19_21291018.en-us.pdf. Accessed August 2017
[14]Mobile messaging fraud report, Available at: https://mobileecosystemforum.com/mobile-messaging-fraud-report-2016/.
[15] Smishing Report, Available at : http://resources.infosecinstitute.com/category/enterprise/phishing/phishing-variations/phishing-variationssmishing/, last accessed 2017/07/15.
[16] The Social Engineering Framework, Available at: https://www.social-engineer.org/framework/attack-vectors/smishing/.
[17] J.W. Joo, S.Y. Moon, S. Singh and J.H. Park, S-Detector: an enhanced security model for detecting Smishing attack for mobile computing, Telecommunication Systems vol. 66(1), 29–38 (2017).
[18] M. Foozy, C. Feresa, R. Ahmad and M.F. Abdollah, A practical rule based technique by splitting SMS phishing from SMS spam for better accuracy in mobile device, International Review on Computers and Software, vol. 9(10), pp. 1776-1782 (2014).
[19] A. Tewari, A. K. Jain and B. B. Gupta, Recent survey of various defense mechanisms against phishing attacks. Journal of Information
[20] Dimensional Enterprise Mobile Security Survey, Available at: http://blog.checkpoint.com/wpcontent/uploads/2017/04/Dimensional_Enterprise-Mobile-Security-Sury.pdf.
[21] N. Choudhary and A.K. Jain, Towards Filtering of SMS Spam Messages Using Machine Learning Based Technique. Advanced Informatics for Computing Research, 18-30, 2017.
[22] A. K. Jain and B. B. Gupta, A novel approach to protect against phishing attacks at client side using auto-updated white-list. EURASIP Journal on Information Security, 2016(9), 2016.
[23] Mobile User Statistics referred from here https://www.statista.com/statistics/274658/forecast-of-mobile-phone-users-in-india/.
[24] Walter Daelemans; Antal van den Bosch (2005). Memory-Based Language Processing. Cambridge University Press.

[25] Stuart Russell and Peter Norvig (2003). Artificial Intelligence: A Modern Approach, second edition, p. 733. Prentice Hall. ISBN 0-13-080302-2
[26] Tom Mitchell (1997). Machine Learning. McGraw-Hill.
[27] D. Randall Wilson; Tony R. Martinez (2000). "Reduction techniques for instance-based learning algorithms". Machine Learning.
[28] Gagliardi, F (2011). "Instance-based classifiers applied to medical databases: Diagnosis and knowledge extraction". Artificial Intelligence in Medicine. 52 (3): 123–139.
[29] John G. Cleary, Leonard E. Trigg: K*: An Instance-based Learner Using an Entropic Distance Measure. In: 12th International Conference on Machine Learning, 108-114, 1995.
[30] https://www.cs.waikato.ac.nz/ml/weka/
[31] D. Aha, D. Kibler (1991). Instance-based learning algorithms. Machine Learning. 6:37-66.
[32] John G. Cleary, Leonard E. Trigg: K*: An Instance-based Learner Using an Entropic Distance Measure. In: 12th International Conference on Machine Learning, 108-114, 1995.
[33] Dayana C. Tejera Hernández University of the Informatics Sciences/Department of Software Engineering and Management, La Habana, 10800, Cuba
[34] S.J. Delany, M. Buckley, D. Greene, Sms spam filtering: methods and data, Expert Syst. Appl. 39 (2012) 9899–9908.
[35] K. Mathew, B. Issac, Intelligent spam classification for mobile text message, in: Computer Science and Network Technology (ICCSNT), 2011 International Conference on, vol. 1, IEEE, 2011, pp. 101–105.
[36] I. Androulidakis, V. Vlachos, A. Papanikolaou, Fimess: filtering mobile external sms spam, in: Proceedings of the 6th Balkan Conference in Informatics, ACM, 2013, pp. 221–227.