A Fuzzy-based approach to Enhance Cyber Defence Security for Next-generation IoT

Article *in* IEEE Internet of Things Journal · January 2021

DOI: 10.1109/JIOT.2021.3053326

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3053326, IEEE Internet of Things Journal

A Fuzzy-based approach to Enhance Cyber Defence Security for Next-generation IoT

Aaisha Makkar*, Uttam Ghosh †, Pradip Kumar Sharma ‡, Amir Javed §,
*Computer Science and Engineering Department, Chandigarh University,Punjab, India, aaisha.makkar@thapar.edu
†Department of EECS, Vanderbilt University, Nashville, TN, USA, ghosh.uttam@ieee.org
‡Department of Computing Science, University of Aberdeen, Aberdeen, UK, pradip.sharma@abdn.ac.uk
§School of Computer Science and Informatics, Cardiff University, Cardiff, UK, a.javed@cs.cardiff.ac.uk

***Abstract*—In modern era, Cognitive Internet of Things (CIoT) in conjunction with IoT evolves which provides the intelligence power of sensing and computation for next-generation IoT (Nx-IoT) networks. The data scientists have discovered a large amount of techniques for knowledge discovery from processed data in CIoT. This task is accomplished successfully and data proceeds for further processing. The major cause for the failure of IoT devices is due to the attacks, in which web spam is more prominent. There seems a requirement of a technique which can detect the web spam before it enters into a device. Motivated from these issues, in this paper, Cognitive spammer framework (CSF) for web spam detection is proposed. CSF detects the web spam by fuzzy rule based classifiers along with machine learning classifiers. Each classifier produces the quality score of the webpage. These quality scores are then ensembled to generate a single score, which predicts the spamicity of the web page. For ensembling, fuzzy voting approach is used in CSF. The experiments were performed using standard dataset WEBSPAM-UK 2007 with respect to accuracy and overhead generated. From the results obtained, it has been demonstrated that CSF improves the accuracy by 97.3%, which is comparatively high in comparison to the other existing approaches in literature.**

***Index Terms*—Ensemble, fuzzy, cognitive, web spam**

I. Introduction

The *Internet of Things (IoT)* plays a key role in providing services to the end users in wide range of applications starting from transportation to medical to mission critical applications to name a few in the next-generation IoT (Nx-IoT) networks. IoT devices sense the environment with the help of network of things, commonly known as *objects*. These objects use the service of Internet to perform the tasks of computation and data transfer in Nx-IoT. Different architectures of IoT are built, depending upon the requirements of different applications. However, none of these architectures meet various International standards and protocols used in such environment for different applications. From the literature survey, it has been found that the intelligent decisions need to be taken by IoT objects in the environment where these are deployed. So, there is a need of new advancement known as *Cognitive Internet of Things (CIoT)* for Nx-IoT.

There are various techniques used for retrieving the useful data/information in CIoT as shown in Fig. 1. Machine learning (ML) is one of those techniques [1]. It plays a key role for automatic decision making, knowledge based information collection, and decision making [2]. It also acts as an abstraction tool in various CIoT applications as discussed below:

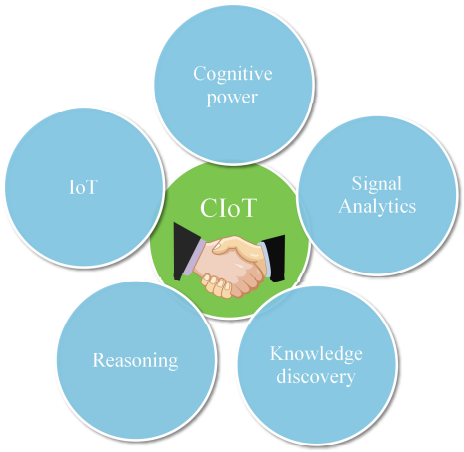

Fig. 1: CIoT trends

- It is one of the optimization techniques, mainly used for decision making with prediction. It was adopted for the future decisions to be taken in the three layer cognitive rings in CIoT [6].
- The reasoning in CIoT technology is done using ML techniques like support vector machine and neural networks. The reasoning with ML is used as the initial step for ontological reasoning. Although ontologies are the semantic technologies which are used to infer the high level, but still prior to their use, ML plays a major role [7].
- Machine learning technique along with CIoT technology has given a new invention in intelligent medical services. It provided the support to the people with wearable sensors to predict the uncertainties that the human body may face. Social interactions with other wearable sensors, daily living atmosphere, monitoring chronic diseases, all of these facilities were developed [8].
- Cognitive radio technology followed machine learning and signal processing for its implementation. Machine learning is used for the proposed theory in the same manner as it was implemented in 'no-regret' theorem of game-theoretic learning [9].
- For enhancing autonomous applications, cognitive management framework is proposed. This framework adopts

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3053326, IEEE Internet of Things Journal

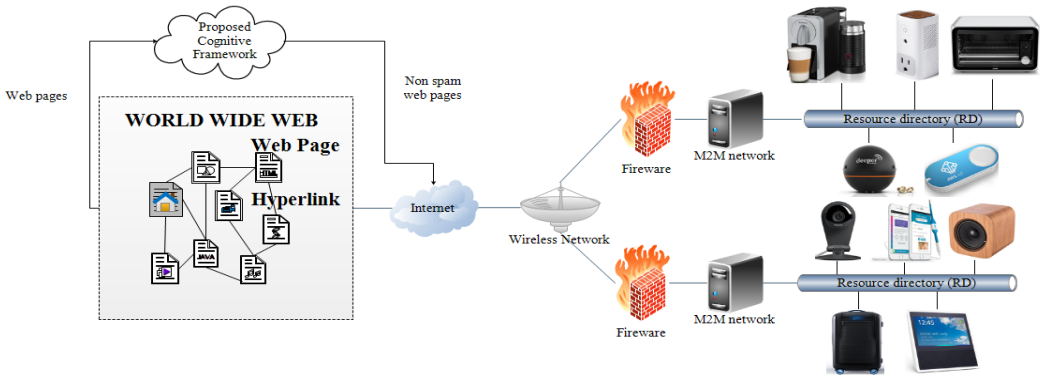

Fig. 2: Role of Internet in CIoT objects

TABLE I: Relative comparison of different existing spam detection algorithms

Author	Method	Description	Dataset	Exe. time	Results	**Advantages**	**Disadvantages**	**Future work**
Wang *et al.*, 2019 [3]	Updated PageRank	Updated PageRank by detecting bias supporters	WEBSPAM-UK2007	Approx 3 hours	99.6% Accuracy	It does not allow the websites to enter into the module of ranking	Complexity increases with increase in IoT devices	Exploring content based features
Becchetti *et al.*, 2008 [4]	Truncated PageRank	Estimation of spam supporters for the detection of link spam	UK-2002, UK-2006	60 days	F-measure: 0.870	Revised PageRank score	Considered only link features	Spam detected effectively at page level
Wang *et al.*, 2008 [3]	DirichletRank algorithm	It solves the problem of zero-one gap in PageRank	.GOV and .UK Dataset	dependency over number of iterations		Solves the zero-one gap problem of PageRank		
Hayati *et al.*, 2010 [5]	Web spam detection	Detects the spam pages by analyzing user behavior attributes	MSLR-WEB10K	21.88 ms	RMSE= 0.47	Detected spam	Visiting patterns not considered in experiments	Spam demotion
Becchetti *et al.*, 2008 [4]	Link analysis	Feature Selection Scheme using Recurrent Neural Networks	WEBSPAM-UK2007		Accuracy 85.6%	Deep learning for web spam detection	Preprocessing data increases the computational time	

machine learning for gathering knowledge and information. It also various machine learning techniques for knowledge based decision making and reasoning [10].

II. Background

From the above discussion, the usage of ML in cognitive applications is illustrated. It seems to be the requirement of present era, to detect the web spam. Motivated from this stipulation, we aimed to detect the malicious web pages by ML with cognition power. Web spam is an unfair practice to change the search engine ranking methodology to get on top of the results pages of the search. Content writers and website developers are helping to implement spamming techniques successfully. Spammers are known to such practitioners. The alternative definition of a web spam is "*the unethical steps taken in support of a substantial website to get boost in the search engine results*". Lots of researchers explored the web spam detection [11], [12].

The existing techniques, mainly focused to detect the malicious webpages after getting indexed by the search engine. Numerous authors attempted to discover various web spam detection techniques as listed below:

1) Ranking algorithm: Google follows the ranking algorithm, PageRank for computing the rank score of the web page. But still, the spammers attempt to manipulate the search results. For example, authors [13], updated the PageRank algorithm in such a manner so as to detect malicious web pages before their appearance in the search results.
2) User behavior: The user behavior analysis is a good parameter for detecting spam web pages. The relevancy of a web page is predicted by the time spent by a user and the number of clicks. The two factors considered by the proposed scheme of web spam detection are dwell time and click count [14].
3) Web page quality: Analyzing the quality of the web page can help in computing its importance. The quality score of the web page was computed and the model developed is known as content trust model [15].
4) Machine learning: It is the paradigm which plays different roles in different fields. It acts as the core for performing the experiments and validating the spam detection algorithms. It predicts the merger of web page features which are used to form a spam web page. Numerous merged web page features are used to train ML models so as to successfully detect the spam with the standard datasets [4].

The Table I summarizes the aforementioned techniques. The web spam detection technique with the cognition power as seen can be effective in different applications. This power can detect the spam automatically by making the system more intelligent.

A. Motivation

Different techniques of machine learning work differently. By combining the results of more techniques, can improve the results. This ensemble phenomenon if given the cognition power can help in the construction of successful web spam technique. Web spam detection is attempted using machine learning by authors [**?**], [16]. But, to the best of our knowledge, the literature includes no cognitive technique for the identification of web spam with fuzzification which can be adopted in IoT devices as well. So, there is a need of an intelligent model to rectify the search engine result pages (SERPs).

On the other hand, fuzzy logic with two class labels is found to be effective as illustrated in [15]. The role of Internet in CIoT objects is essential as shown in Fig. 2. Even, the search engine *Google*, uses the fuzzy logic for detecting the email spam, [17]. So, the proposed approach develops a cognitive framework to detect web spam using fuzzy rule based classifier and fuzzy ensemble approach.

B. Contributions

Following are the major contributions of the paper:

1) The dataset WEBSPAM-UK 2007, is preprocessed with feature extraction technique, i.e., PCA. Then this collected data is cleaned using the (SOTU) approach.
2) Three machine learning classifiers and one fuzzy rule based classifier, are designed for the detection of web spam.
3) The fuzzy voting approach is used for the computation of the aggregated results.
4) The tests were carried out on the standard dataset, i.e. WEBSPAM-UK 2007, whereby the proposed scheme is found to be superior compared to the other current schemes in its literature.

C. Organization

The rest of the paper is structured as follows. Section III discussed the System model. The proposed scheme is illustrated in Section IV. Section V presents the results and discussion. Finally, the paper is concluded in Section VI.

III. System model

CIoT enabled the systems to work more intelligently with the power of environment sensing. One of the powerful tools, i.e., machine learning, helps in prediction and data extraction in CIoT. Ensemble of such machine learning models with fuzzification can be the good for web spam detection. Detecting the spam can help the search engine to give better results. This proposal aims to improve the accuracy of machine learning models with the help of new fuzzy classifier and fuzzy ensemble technique defined as below.

$$max(\beta) = \alpha + c, \tag{1}$$

α refers to the accuracy achieved by the execution of machine learning model. c is the constant value which is the accuracy obtained after the ensemble of multiple models. β is the summation of α and c, i.e., the improved accuracy. In Eq. 1, β is to be maximized, which means the better detection of web spam. It automatically improves the search results by providing this CIoT power of automated detection to the search engine results.

IV. Cognitive Fuzzification Spammer: The Proposed Scheme

The proposed framework detects the web spam with the fuzzy classifier and machine learning (ML) models. These models are then ensembled to build an intelligent model. Ensemble approach targets to improve the accuracy of ML models. Fuzzy system methodology started with the fuzzy logic and expanded with assigning the quality score to each webpage. The predicted quality score by each classifier is used to detect the spam pages. The quality score is computed with the QAIR methodology. The fuzzy classifier along with other ML classifiers are ensembled using fuzzy voting approach. The complete architecture of CFS is presented in Fig. 3. Table II represents the different fuzzy system using different methodologies for spam detection which exist in the literature.

TABLE II: Different fuzzy methods

Purpose	Method	Data
Email spam detection [18]	Fuzzy clustering algorithm	ASSP search engine
Removing web spam links [19]	Fuzzy lattice reasoning classifier (FLR)	Google result pages
Detecting phishing activities[20]	Fuzzy data mining techniques	e-banking website
Email spam detection [21]	Fuzzy voting method	Google data

Many ensemble methods exists in literature. Few of these are listed below:

- Unweighted voting: Each classifier not only produces the classification decision but also the class probability estimation. The estimator produced by all the classifiers are combined in Eq. 2. In this equation, h_l is the classifier, which results in true prediction for k at a data point x.

$$P(f(x) = k) = \frac{1}{L}\sum_{l=1}^{L} P(f(x) = k|h_l) \tag{2}$$

- Least squares: This method is used for regression problems. This method works to target the maximum weights which improves the accuracy of the ensemble model. The principle applied states that the variance of estimate by h_l is inversely proportional to the weight of h_l.
- Likelihood combination: This method is suitable for classification problems. In this method, the accuracy of each classifier is computed for the independent classifier weight. The methodology of this method uses the prior distribution P(h_l) which is multiplied with estimated likelihood P($S|h_l$).
- Gating networks: It is the method of combining classifiers which accepts the input x and produces the output w_l. Eq. 3 uses the input x and parameter V for computation of Z. After computing Z, weights W are computed in Eq. 4. The output of Eq.4 W_l is known as the *soft-max* of output of Eq.3 Z_l.

$$Z_l = V_{l^T} * x, \tag{3}$$

$$W_l = \frac{e^{Z_l}}{e\sum_u^{Z_u}} \tag{4}$$

- Stacking: It is implemented with leave-one-out cross validation. For each classifier, and for each training set, a combinational hypothesis is produced. In the next iteration, the same procedure is used except the last-one.

The proposed framework uses the approach as shown in Fig. 3. The experimental data needs to be clean. So, the preprocessing of data is required. It requires various steps like finding the missing values. The clean data not only fits properly in the model but also improves the performance of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3053326, IEEE Internet of Things Journal

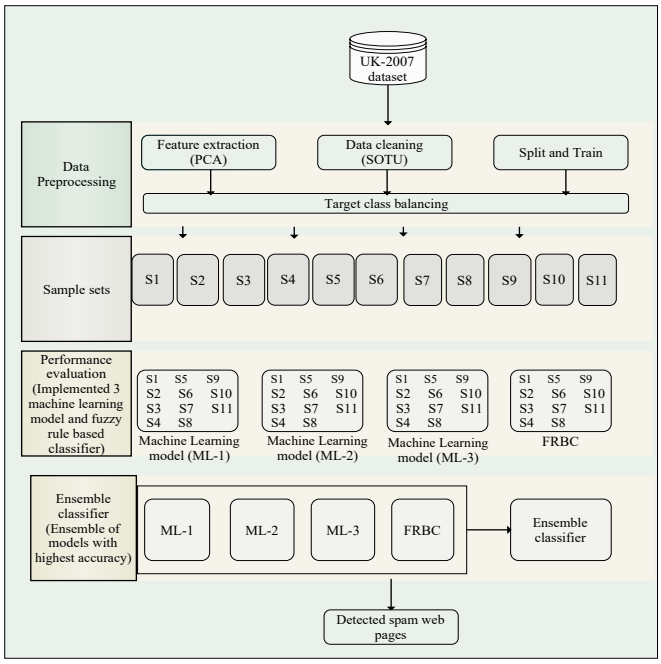

Fig. 3: Architecture of cognitive fuzzification spammer framework

model. Various steps used in the proposed framework are as follows.

- Feature extraction: Feature extraction reduces the dimension of the dataset. It is the procedure of transforming the correlated variables into uncorrelated variables. It is done by extracting the variance among the variables [17]. It is also known as principal axis method or data compression technique. The standard value of each row is multiplied with the standard value of each column, which results in formation of Principal component (PC). The variability in data is presented in the form of PCs. The first 7 PCs are represented in Fig. 5 as used in the proposed scheme.
- Feature selection: The feature importance is calculated using the feature ranking method, i.e., *random forest*. This method works on the principal of decision trees. The score computed for ranking the features is the 'Gini index'. The different features form the different nodes of tree. The Gini value is calculated for each leaf either it is the parent leaf or the child leaf. Then, these Gini values are used for the computation of mean decrease Gini values. The results of this round are presented in Table III. The optimal features selected are: hostid, Label, eq_hp_mp, indegree_hp, indegree_mp, outdegree_hp, outdegree_mp, pagerank_hp, pagerank_mp, trustrank_hp, trustrank_mp, truncatedpagerank_1_hp ,truncatedpagerank_1_mp.
- Data cleaning: The important module for good accuracy of the scheme is the balanced classes. As observed in the data collection used for experiments of this proposal, the spam labels are relatively less than the non spam labels. So, an over sampling method, *SOTU* is used. The data is distributed in sample sets by Eq. 5 and each classifier is trained with each set.

TABLE III: Feature importance score

Variable Description	Gini value	DVariable Description	Gini value
Is the home page the page with the maximum PageRank in the host?	3.66	Fraction of out-links that are also in-links of hp.	2.15
Assortativity coefficient of the home page	7.18	Fraction of out-links that are also in-links of mp	1.92
Assortativity coefficient of the page with the maximum PageRank	7.87	Number of different hosts pointing to hp	5.57
Average in-degree of out-neighbors of home page	4.28	Number of different hosts pointing to mp	5.75
Average in-degree of out-neighbors of page with maximum PageRank	5.02	Number of different hosts (approx.) supporting at distance 2 the hp	9.03
Average out-degree of in-neighbors of hp	7.32	Number of different hosts (approx.) supporting at distance 2 the mp	7.79
Average out-degree of in-neighbors of mp	7.24	Number of different hosts (approx.) supporting at distance 3 the hp	9.89
Indegree of hp	11.43	Number of different hosts (approx.) supporting at distance 3 the mp	10.52
Indegree of mp	10.36	Number of different hosts (approx.) supporting at distance 4 the hp	12.58
Neighbors at distance 2 of hp	13.43	Number of different hosts (approx.) supporting at distance 4 the mp	12.53
Neighbors at distance 2 of mp	13.41	TruncatedPageRank using truncation distance 1, hp	22.87
Neighbors at distance 3 of hp	12.76	TruncatedPageRank using truncation distance 1, mp	22.60
Neighbors at distance 3 of mp	14.21	TruncatedPageRank using truncation distance 2, hp	21.43
Neighbors at distance 4 of hp	10.46	TruncatedPageRank using truncation distance 2, mp	23.35
Neighbors at distance 4 of mp	11.40	TruncatedPageRank using truncation distance 3, hp	20.54
Out-degree of hp	3.65	TruncatedPageRank using truncation distance 3, mp	22.41
Out-degree of mp	5.13	TruncatedPageRank using truncation distance 4, hp	21.91
PageRank of hp	20.97	TruncatedPageRank using truncation distance 4, mp	22.87
PageRank of mp	21.85	TrustRank of hp	18.85
Standard deviation of the PageRank of in-neighbors of hp	7.15	TrustRank of hp	19.07

$$S = \sum_{i=1}^{d} \frac{x_m + x_n}{d} \tag{5}$$

In Eq. 5, S refers to the training file of dataset. x_m refers to the instances of minor class and x_n refers to the instances of major class. The value of d depends on the problem type and selected features. It has to be an odd number for classification problem. It must be an even number for regression problem. The approximate number should be near the number of features (suggested variation of 1 if required). In our case, it is 13, an odd number because of the classification problem and 13 features were selected in the previous phase.

- Machine learning models: The data is already selected from the dataset in the previous rounds of feature extraction and feature selection. The filtered data is ready for the experiments and fed into the ML models. The three different ML models along with fuzzy rule based classifier are used for experiments. To determine their applicability to our function, we evaluated a total of eight distinct classification models from the Weka toolkit. Every model has a unique set of properties that we shall address briefly here. The next section presents

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3053326, IEEE Internet of Things Journal

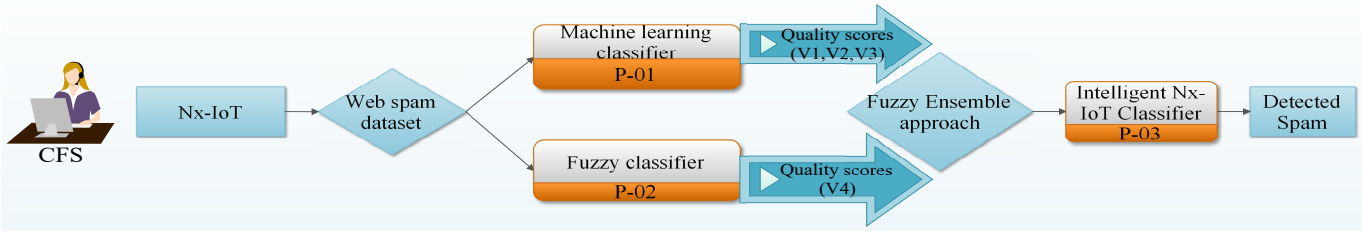

Fig. 4: Steps followed in the formulation of CFS

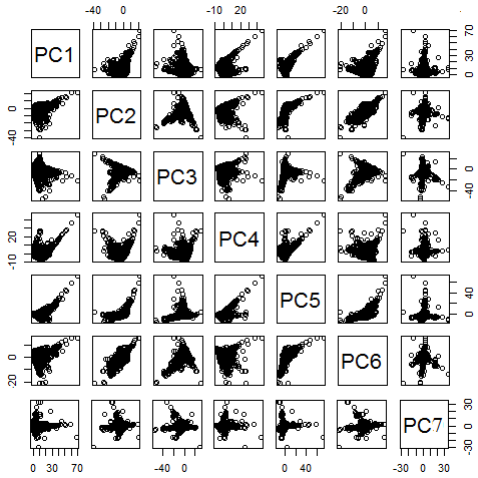

Fig. 5: Transformations of PCs

a comparative examination of the current classification models.

- Ensemble approach: Ensemble is the technique which is used to improve the performance of classifiers. In the proposed framework, the three ML models, i.e., Bagged Mars, Bayesian Generalized Linear Model, Boosted linear model, and one fuzzy rule based classifier (FRBC) are ensembled as shown in Fig. 4. The ensemble approach has been used, which is built in such a manner that it improves the performance of each classifiers. It uses the fundamental fuzzy logic. The following steps are used for ensemble the results of all the classifiers:
  1) Collect the output of each classifier, i.e., the quality score of the webpage. (v1= vector from first classifier)
     (v2= vector from second classifier)
     (v3= vector from third classifier)
     (v4= vector from fourth classifier)
  2) Normalize each vector within the range from 0 to 1.
  3) More the score is close to 0, more is the probability of spam. More the score is close to 1, good is the quality of web page.
  4) Use Eq. 6 for the generation of combined all the votes.
  5) Evaluate the performance.

$$S = \sum_{i=1}^{n} \frac{P|h_i}{n} \tag{6}$$

In Eq.6, n refers to number of classifiers, h_i is the hypothesis generated by each classifier.

V. Results and Discussion

The proposed framework enhances the ability of search engine by detecting the spam pages. The results of the proposed scheme are presented as follows.

TABLE IV: Distribution of the number of pages reviewed by judges

SET	Spam	Nonspam	Undecided
SET1	222	3776	277
SET2	122	1933	149

A. Data Collection

The proposed framework is validated by performing the experiments on the publicly available dataset, i.e., WEBSPAM-UK2007. This data collection is launched by the *Universita degli Studi di Milano*, the Laboratory of Web Algorithms [22]. It is the collection of 114529 hosts with 41 features. It contains the information of webpages like host_id, pagerank score, incoming links, outgoing links, and many more. The labels were released in two different sets, SET1 containing 2/3 of the label hosts are used for training and SET2 containing 1/3 of the label hosts are used for testing. The labels provided by judges, are named as 'spam', 'nonspam', 'borderline' and 'cannot classify'. The scores are also marked to take the final decision for classification, spam-1, non-spam-0, borderline-0.5, count of each category is represented in Table IV.

B. Impact of machine learning models

The data is pre-processed with the help of feature extraction and feature selection techniques. The resulted data is clean and noisy free. ML models are then trained with the pre-processed data. Four different classifiers are trained as discussed below:

1) Bagged MARS: Multivariate adaptive regression splines (MARS) offer a simple method for capturing the non-linearity dimension of polynomial regression by evaluating cut-points (knots), comparable to phase features. For each predictor, the approach evaluates each data point as a knot, and produces a linear regression model with the candidate's features. This process will continue until several knots are found, resulting in an extremely non-linear pattern. Although we can add a very nice link to our training data with the addition of several knots, new, invisible knowledge can not generalize very well. The ROC formed is as shown in Fig. 6.
2) Bayesian Generalized Linear Model: Randomness is a result of the incomplete random variable information. Upon examination of the results, we use the Bayes rule

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3053326, IEEE Internet of Things Journal

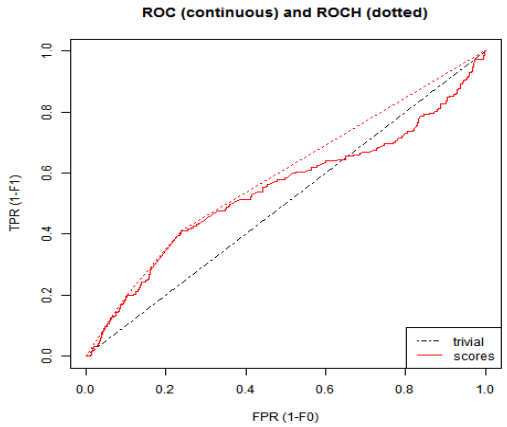

Fig. 6: ROC plot of Bagged MARS

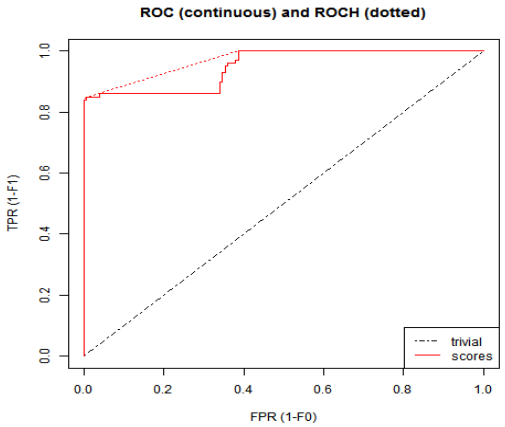

Fig. 8: ROC plot of Boosted Linear Model

to change the distribution beforehand and obtain the distribution afterwards. The below equations are used. The ROC formed is as shown in Fig. 7.

- Logistic model:

$$p(Y_i \mid X_i, \beta) \sim \text{Binomial}(n_i, p_i) \tag{7}$$

where, $p_i = \text{logit}^{-1}(\beta^T X)$

- Poisson model:

$$p(Y_i \mid X_i, \beta) \sim \text{Poisson}(\lambda_i) \tag{8}$$

where, $\lambda_i = e^{\beta^T X}$

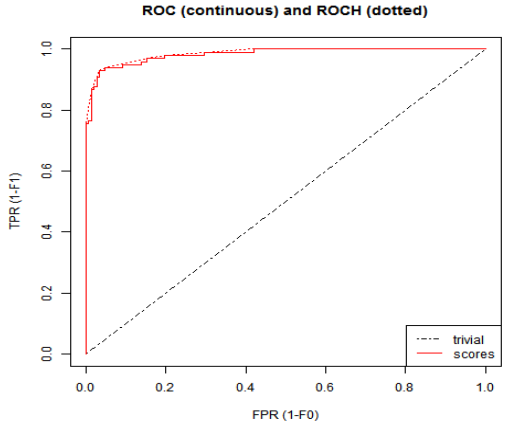

Fig. 7: ROC plot of Bayesian Generalized Linear Model

3) Boosted Linear Model: This method used 'BstLm' as the argument. 'bst, plyr' packages are required for its execution. The ROC formed is as shown in Fig. 8.
4) Fuzzy rule based classifier: Fuzzy rule base classifier is implemented in R. It consists of IF-THEN rules. The class labels are fixed to two. A single layer function is build. The *'MAMDANI'* FRBS model is implemented. The method used by de-fuzzification for rule evaluation is *'Weighted average method'* method. The name of the model is *'sim-0'*. It treated the 13 features as a combination of three variables, in which var.1 and var.2 are taken as input variables and var.3 as output variable. The interval for the training variables are fixed, having range of -2.716 to 2.834 for *Var1*, 45.6 to 60.5 for *Var2* and *Var2* The rules produced are in the form of:

$\langle Ruleid = "1" \rangle$
$\langle If \rangle$
$\langle CompoundPredicatebooleanOperator = "and" \rangle$
$\langle SimplePredicatefield = "var.1" value = "v.1\_a.7" / \rangle$
$\langle SimplePredicatefield = "var.2" value = "v.2\_a.12" / \rangle$
$\langle / CompoundPredicate \rangle$
$\langle / If \rangle$

The proposed fuzzy voting ensemble approach improves the performance of models as discussed in Table VII.

TABLE VI: Performance of Accuracy and Execution time with respect to models and samples

Set	ML-1		ML-2		ML-3		FRBC	
	ET	PA	ET	PA	ET	PA	ET	PA
S1	65.2	96.79	6.9	98	64.3	92.22	78.6	92.2
S2	50.2	96.8	7.4	98.8	92.3	96.8	62.1	88.8
S3	285.14	96	8.6	98.4	13.2	98	79.1	86.8
S4	137.4	96.81	12.4	98.01	24.5	97.21	98.5	90.8
S5	268.68	95.6	4.3	98.8	116.7	94	213.9	89.8
S6	149.5	96.81	0.88	95.6	198.5	96	66.4	87.99
S7	168	97.6	3.4	100	47.99	97.2	265.2	88.8
S8	79.32	96.53	2.5	77.8	14.8	97.59	234.2	67.47
S9	114.5	99.2	13.14	99.67	17.9	98.8	136.3	85.4
S10	75.88	98.41	13.14	99	26.51	98.01	52.2	84.46
S11	65.58	96.8	13.14	97	24	96.8	72	88.4
Avg.	132.64	96.79	7.18	96.46	58.2	89.2	123.5	95.5
ET- Execution time , PA- Performance Accuracy								

TABLE VII: Results of ensemble model

Web Page	ML-1	MI-2	ML-3	FRBC	Ensemble
1	0.5	0.4	0.45	0.43	0.52
2	0.85	0.43	0.56	0.58	0.59
3	0.11	0.32	0.34	0.35	0.45
4	0.67	0.41	0.45	0.43	0.56
5	0.76	0.65	0.72	0.75	0.78
6	0.12	0.55	0.52	0.32	0.57
7	0.23	0.54	0.53	0.48	0.55
8	0.22	0.31	0.34	0.37	0.39
9	0.34	0.54	0.44	0.44	0.49
10	0.97	0.98	0.87	0.86	0.95
11	0.54	0.43	0.5	0.45	0.46
12	0.12	0.11	0.13	0.15	0.16
13	0.89	0.87	0.89	0.76	0.88
14	0.87	0.43	0.6	0.67	0.63
15	0.98	0.88	0.99	0.95	0.96
16	0.78	0.87	0.8	0.85	0.86
17	0.13	0.76	0.56	0.65	0.6
18	0.25	0.24	0.21	0.29	0.26
19	0.64	0.65	0.66	0.54	0.68
20	0.65	0.78	0.75	0.76	0.71

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3053326, IEEE Internet of Things Journal

TABLE V: Summary of performance of the experimental models

Model	H	Gini	AUC	AUCH	KS	MER	MWL	ER	Sens	Spec	P	R	Accuracy
ML-1	0.97	0.97	0.94	0.97	0.97	0.01	0.01	0.03	0.92	1	1	0.92	96.79
ML-2	0.78	0.89	0.94	0.97	0.83	0.06	0.07	0.08	0.79	1	1	0.79	96.46
ML-3	0.82	0.96	0.93	0.99	0.86	0.05	0.05	0.10	0.75	1	1	0.75	89.20
FRBC	0.85	0.92	0.959	0.96	0.89	0.04	0.05	0.65	0.91	0.01	1	0.89	95.50
Ensemble	0.86	0.93	0.96	0.97	0.89	0.04	0.05	0.04	0.89	1	1	0.89	97.30

C. Comparison of machine learning models

A summary of relative comparison of various parameters is presented in Table V. Each model is trained with 11 sets of data, formed by the data cleaning approach. So, the Table VI compares the execution time and accuracy of different classifier with different sample set.

D. Training and testing criteria

The results evaluated in the experiments by ML models are highly depends upon the system learning. If the system learns good during training, like identifies all the patterns in the data, able to identify the outliers. Then, the system performs best during testing. The data used for training and testing can be stored collectively. The ratio differentiates both the sets. In our experiments, we used the ratio as 40:60. But, for the validation, we have evaluated different metrics as shown in Table. VIII.

E. Limitation of the proposed scheme

As the proposed scheme deals with the detection and prevention of spam images, so handling the large volume of images is difficult in IoT devices. The proposed scheme does not perform well in the IoT environment where edge storage is preferably recommended. The data produced by the IoT devices are processed at edge server rather than cloud. At such environment, dealing with the images with the proposed scheme suffers from the limitation of complexity. If the number of devices increases, the computational time and searching time also increases. This increased cost in terms of computational time and searching time is presented in Fig. 9.

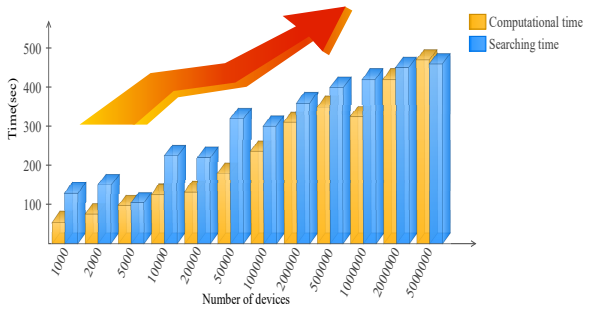

Fig. 9: Increase in computational and searching time with increase in IoT devices

F. Comparison with existing approaches

The various existing benchmark techniques for web spam detection are compared with the proposed scheme. The evaluation parameter, i.e., accuracy is used for comparison. It can be concluded that the proposed scheme performed superior in comparison to the existing schemes. The analysis of different algorithms is presented in Fig. 10.

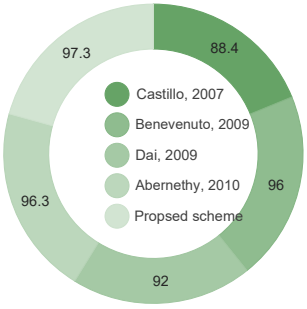

Fig. 10: Accuracy achieved by different spam detection algorithms

TABLE VIII: Precondition set up for experiments

Models	Acc. (40:60)	Exe. time (40:60)	Acc. (60:40)	Exe. time (60:40)	Acc. (70:30)	Exe. time (70:30)
ML-1	96.79	132.64	75.6	145.2	89.5	98.4
ML-2	96.46	7.18	86.7	67.5	88.4	55.6
ML-3	89.2	58.2	45.1	32.6	85.1	112.8
FRBC	95.5	123.5	89.2	155.6	91.6	178.9
Ensemble	97.3	5.31	92.5	6.1	95.8	5.6

The experiments to test the efficiency of the proposed scheme were carried out on various data sets. The three separate datasets which are used are:

- WEBSPAM-UK 2006
- WEBSPAM-UK 2007
- Microsoft Learning to Rank(MLR)

It is observed from the Fig. 11 that the efficiency of the proposed scheme depends on the size of the experimental but also increases the cost incurred when balancing the class with the increase in the data size. We may infer that in order to enforce using the proposed method, if the data analysis needs to be performed in terms of scale.

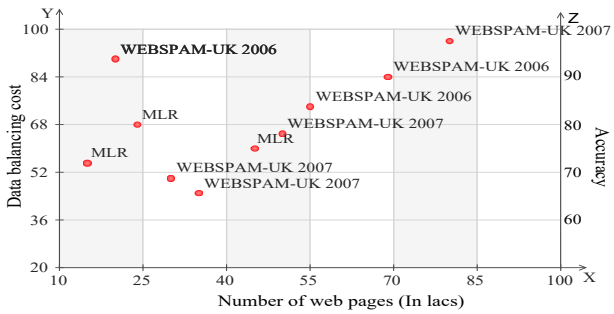

Fig. 11: Effect of datasets at data balancing cost and accuracy

VI. Conclusion

Cognitive framework, machine learning technique is used to built with fuzzy technique for web spam detection. This framework detects the spam successfully. The proposed ensemble approach improves the performance of individual classifier. The objective of the proposed framework is to enhance the

accuracy, which has been accomplished. It reduces the load of ranking module of search engine. The experiments are conducted on web spam dataset. The dataset is preprocessed before performing the experiments.

In future, we would like to explore more CIoT applications that would help in web spam detection.

REFERENCES

[1] N. Kumar and A. Makkar, *Machine Learning in Cognitive IoT*. CRC Press, 2020.

[2] A. Makkar and S. Goel, "Spammer classification using ensemble methods over content-based features," in *Proceedings of Sixth International Conference on Soft Computing for Problem Solving*. Springer, 2017, pp. 1–9.

[3] X. Wang, T. Tao, J.-T. Sun, A. Shakery, and C. Zhai, "Dirichletrank: Solving the zero-one gap problem of pagerank," *ACM Transactions on Information Systems (TOIS)*, vol. 26, no. 2, p. 10, 2008.

[4] L. Becchetti, C. Castillo, D. Donato, R. Baeza-Yates, and S. Leonardi, "Link analysis for web spam detection," *ACM Transactions on the Web (TWEB)*, vol. 2, no. 1, p. 2, 2008.

[5] P. Hayati, K. Chai, V. Potdar, and A. Talevski, "Behaviour-based web spambot detection by utilising action time and action frequency," in *International Conference on Computational Science and Its Applications*. Springer, 2010, pp. 351–360.

[6] M. Zhang, H. Zhao, R. Zheng, Q. Wu, and W. Wei, "Cognitive internet of things: concepts and application example," *International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 6, p. 151, 2012.

[7] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE communications surveys & tutorials*, vol. 16, no. 1, pp. 414–454, 2014.

[8] D. Bandyopadhyay and J. Sen, "Internet of things: Applications and challenges in technology and standardization," *Wireless Personal Communications*, vol. 58, no. 1, pp. 49–69, 2011.

[9] S. Haykin *et al.*, "Cognitive radio: brain-empowered wireless communications," *IEEE journal on selected areas in communications*, vol. 23, no. 2, pp. 201–220, 2005.

[10] V. Foteinos, D. Kelaidonis, G. Poulios, P. Vlacheas, V. Stavroulaki, and P. Demestichas, "Cognitive management for the internet of things: A framework for enabling autonomous applications," *IEEE vehicular technology magazine*, vol. 8, no. 4, pp. 90–99, 2013.

[11] A. Makkar, S. Garg, N. Kumar, M. S. Hossain, A. Ghoneim, and M. Alrashoud, "An efficient spam detection technique for iot devices using machine learning," *IEEE Transactions on Industrial Informatics*, 2020.

[12] A. Makkar and N. Kumar, "An efficient deep learning-based scheme for web spam detection in iot environment," *Future Generation Computer Systems*, 2020.

[13] ——, "Cognitive spammer: a framework for pagerank analysis with split by over-sampling and train by under-fitting," *Future Generation Computer Systems*, vol. 90, pp. 381–404, 2019.

[14] ——, "User behavior analysis-based smart energy management for webpage ranking: Learning automata-based solution," *Sustainable Computing: Informatics and Systems*, vol. 20, pp. 174–191, 2018.

[15] A. Makkar, N. Kumar, M. S. Obaidat, and K.-F. Hsiao, "Qair: Quality assessment scheme for information retrieval in iot infrastructures," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[16] C. Wang, A. Kalra, C. Borcea, and Y. Chen, "Webpage depth-level dwell time prediction," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, 2016, pp. 1937–1940.

[17] A. Makkar, M. S. Obaidat, and N. Kumar, "Fs2rnn: Feature selection scheme for web spam detection using recurrent neural networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[18] R. Vijayan, S. Viknesh, and S. Subhashini, "An anti-spam engine using fuzzy logic with enhanced performance tuning," *International Journal of Computer Applications*, vol. 975, p. 8887, 2011.

[19] M. Egele, C. Kolbitsch, and C. Platzer, "Removing web spam links from search engine results," *Journal in Computer Virology*, vol. 7, no. 1, pp. 51–62, 2011.

[20] M. Aburrous, M. A. Hossain, K. Dahal, and F. Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert systems with applications*, vol. 37, no. 12, pp. 7913–7921, 2010.

[21] W. Lin, "Fuzzy logic voting method and system for classifying e-mail using inputs from multiple spam classifiers," Mar. 16 2010, uS Patent 7,680,890.

[22] L. B. P. B. M. S. Carlos Castillo, Debora Donato and S. Vigna, "Web spam collections," http://chato.cl/webspam/datasets/, 2007 (accessed November 10, 2016).

Aaisha Makkar received her Ph.D. degree from the Computer Science and Engineering Department in Thapar Institute of Engineering Technology, Patiala(Punjab), India. Her research interests includes cyber security, web mining, algorithms, machine learning, deep learning and Internet of things.

Uttam Ghosh is working as an Assistant Professor of the Practice in the Department of Electrical Engineering and Computer Science, Vanderbilt University, USA. Dr. Ghosh obtained his Post-doctoral experience at the University of Illinois in Urbana-Champaign, Fordham University, and Tennessee State University.

Dr. Pradip Kumar Sharma (M'18) is an Assistant Professor in Cybersecurity in the Department of Computing Science at the University of Aberdeen, UK. He received his Ph.D. in CSE (August, 2019) from the Seoul National University of Science and Technology, South Korea. He also worked as a Postdoctoral Research Fellow in the Department of Multimedia Engineering at the Dongguk University, South Korea.

Amir Javed Recently got his PhD from Cardiff University where he also received his MSc Degree in Information security and privacy. He is currently a Lecture at Cardiff University and has worked on a number research projects, as a research associate in Cyber Security Analytics in Cardiff.