A Swarm Negative Selection Algorithm for Email Spam Detection

Article · January 2015

DOI: 10.4172/2324-9307.1000122

1 author:

Ali Selamat
Universiti Teknologi Malaysia
326 PUBLICATIONS **1,476** CITATIONS

SEE PROFILE

Idris and Selamat, J Comput Eng Inf Technol 2015, 4:1
http://dx.doi.org/10.4172/2324-9307.1000122

Journal of Computer Engineering & Information Technology

Research Article A SCITECHNOL JOURNAL

A Swarm Negative Selection Algorithm for Email Spam Detection

Ismaila Idris[1*] and Ali Selamat[2]

[1]*Department of Cyber Security Science, School of Information Communication Technology, Federal University of Technology, P.M.B 65, Minna, Niger State, Nigeria*

[2]*Software Engineering Research Group (SERG), Knowledge Economy Research Alliance and Faculty of Computing, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia*

[*]**Corresponding author**: Ismaila Idris, Department of Cyber Security Science, School of Information Communication Technology, Federal University of Technology, P.M.B 65, Minna, Niger State, Nigeria; E-mail: ismi.idris@futminna.edu.ng, idris.ismaila95@gmail.com

Rec Date: July 18, 2014 **Acc Date:** March 10, 2015 **Pub Date:** March 17, 2015

Abstract

The increased nature of email spam with the use of urge mailing tools prompt the need for detector generation to counter the menace of unsolocited email. Detector generation inspired by the human immune system implements particle swarm optimization (PSO) to generate detector in negative selection algorithm (NSA). Outlier detectors are unique features generated by local outlier factor (LOF). The local outlier factor is implemented as fitness function to determine the local best (Pbest) of each candidate detector. Velocity and position of particle swarm optimization is employed to support the movement and new particle position of each outlier detector. The particle swarm optimization (PSO) is implemented to improve detector generation in negative selection algorithm rather than the random generation of detectors. The model is called swarm negative selection algorithm (SNSA). The experimental result show that the proposed SNSA model performs better than the standard NSA.

Keywords: Detectors; Negative selection algorithm; Differential evolution; Email; Spam; Non-spam

Introduction

A battle against spam is a very difficult one; therefore, it makes a lot of sense to fight an adaptive pathogen with an adaptive system. This brought about the study of negative selection algorithm which is an adaptive algorithm in the fight of email spam One significant and growing task that resulted from unsolicited email is the classification of email. This pose a problem among cooperate organizations and individuals trying to solve this menace call email spam. The task of email classification is shared into sub-tasks [**1**]. The initial task is the collection of data and email message representation. Secondly, the selection of email feature and dimensional reduction of features [**2**], finally is the mapping of both training and testing set for classification of email. The essence of classification is to distinguish between spam and non-spam email. The problem of email spam is a global issue and often encountered by all email users. It is defined as an unwanted junk email delivered to services on internet mail. The amount of email spam has sky rocket due to bulk mailing tools, this annoyed the receivers the more and the internet service providers (ISP) are constantly under great pressure and complain on the problem of unsolicited email messages. The paper was organized in to six sections, Section 1 is the introduction, Section 2 discusses the related work in negative selection algorithm, the proposed improved model and its constituent framework was discussed in Section 3. Empirical studies, results and discussion was in Section 4, Section 5 discuss the experimental results while conclusion and recommendation was in Section 6.

Related work

Major work implemented in the combination of two different algorithm in email spam was an hybrid model of artificial immune system (AIS) based on module whose extracted features was designed by [**3**] and further used for logistic regression model. A set of detector was initially generated with the use of terms that are extracted from the training message, and also data on matched detector use in regression model. Spam-assassin was used for the experimental work. Rough set theory which is a mathematical approach for approximate reasoning in other to group messages in three class was proposed by [**4**], targeting low false positive. The selection of feature; spam, non-spam or suspicious was first implemented on the training set after which genetic algorithm was implemented. The universe of message is divided in to three region base on some induced set of rules. The experiment used only 11 features of the UCI corpus. It was concluded that the techniques is very efficient in reducing number of non-spam message that are blocked and superior to naïve Bayes classifier. A genetic optimized spam detection using AIS algorithm was proposed by [**5**]. The genetic algorithm optimized AIS to cull old lymphocytes (Replacing the old lymphocyte with new ones) and also check for new interest for users in a way that is similar. In updating intervals such as the number of receive messages, the interval is updated with respect to time, user request and so on, many choices were used in selecting the update intervals which was the aim of using the genetic algorithm. The implementation of different pattern recognition scheme inspired by biological immune system in order to identify uncommon situations like the email spam [**6**,**7**], unfortunately has not been able to produce outstanding results due to the scalability of the generated detectors.

The proposed model

Implementation of negative selection algorithm

The real value negative selection algorithm is encoded in real valued for classifying non-spam and spam. The dataset used in this research is implemented in real value, there is need to define the non-spam and the spam space. The non-spam space is the normal state of a system while the spam space is the abnormal state of a system. The candidate detector is randomly generated and then compared to the non-spam samples. Candidate detectors that do not match any sample of the non-spam set are accepted as viable detectors. Candidate detectors that matches sample of the non-spam set are discarded as unwanted detectors. The non-spam sample in a real value negative selection algorithm is represented in N-dimensional points and a non-spam radius R_s, as training dataset. In clearer terms, let equation (2) represents the non-spam space.

$$S = \{X_i \mid i=1,2,\ldots\ldots m;\ Rs=r\} \qquad (1)$$

X_i are some point in the normalized N-dimensional space.

$X_i = \{x_{i1}, x_{i2}, x_{i3}\ldots\ldots\ldots X_{iN}\}$, I=1, 2, 3.....m (2)

The entire normalized sample$space^{I\subset}[0,1]^N$, the spam space can then be represented as S = I − NS where S is spam and NS is non-spam.

$$dj = (Cj, Rdj) \qquad (3)$$

Equation (3) denote one detector where $C_j = \{C_{j1}, C_{j2}, C_{j3} \ldots C_{Jn}\}$ is the detector center respectively, R_j is the detector radius. The Euclidean distance is used as the matching measurement. The distance between non-spam sample X_i and the detector d_j can be defined as:

$$L(Xi,dj) = \sqrt{(xi1 - cj1)^2 + \cdots + (xiN - CjN)^2} \qquad (4)$$

L (X_i, d_j) is compared with the non-spam space threshold R_s, obtaining the matching value of

idjRs (5)

The detector d_j fails to match the non-spam sample X_i if > 0, therefore if d_j does not match any non-spam sample, it will be retained in the detector set. The detector threshold R^d, j of detector d_j can be defined as:

Rd, j= min (),if (6)

If detector d_j match the non-spam sample, it will be discarded. This will not stop the generation of detector until the required detector set is reached and the required spam space covered. The generated detector set can then be used to monitor the entire system.

The proposed improved model

Definition of non-spam space: In the case of real value negative selection algorithm, there is need to define the non-spam and the spam space. The non-spam space is the normal state of a system while the spam space is the abnormal state of a system.

Let's assume the non-spam space to be *S* defined as:

$$s = (s1\ldots sn) = \begin{bmatrix} S11 & \cdots & S1m \\ \vdots & \ddots & \vdots \\ Sn1 & \cdots & Snm \end{bmatrix} \qquad (7)$$

$S_{ij} \in K^m$, i = 1,……, n; j = 1,……,m

S is normalized as follows:

$$Si = \frac{Si}{\|Si\|} \qquad (8)$$

Therefore, s_i is the i^{th} non-spam unit; and s_{ij} is the J^{th} vector of the i^{th} non-spam unit.

Detector generation Parameters and implementation: Particles are made up of 57 features {f57} while the accelerated constant C value is 0.5.

The position and velocity of particle swarm optimization are represented in the N-dimensional space as:

p_i (p_i^1,p_i^2,...p_i^n) (9)

V_i (v_i^1, v_i^2,...v_i^n) (10)

Where p_{id} the binary bits, i=1, 2 … m (m is set to be the total number of particles), d = 1,2 ... n (n is the dimensionality of the data). Each particle in the generation updates its own position and velocity base on equation (12) and (13).

The initialization of the real valued particle swarm optimization is established by population of particles (non-spam and spam). All particle moves in the problem space in other to find the optimal solution in individual iteration. Given n-dimensional space, the particles exhibit potential solution while each particle possess direction and position vector for its movement and direction.

In generating a random initial velocity matrix for random candidate detector we have v(0)

$$v(0) = \begin{bmatrix} v_1{}^1(0)\ldots v_1{}^m(0) \\ \vdots \quad \ddots \quad \vdots \\ v_j{}^1(0)\ldots v_j{}^m(0) \end{bmatrix} \qquad (11)$$

Equation (9) and (10) calculate the new velocity and particle position as above

$$v_{id}(t+1) = v_{id}(t) + c[Pbest_{id}(t) - x_{id}(t)] \quad (12)$$

$$x_{id}(t+1) = x_{id}(t) = v_{id}(t+1) \qquad (13)$$

The process of the methodology can be explained in the following steps (Figure 1):

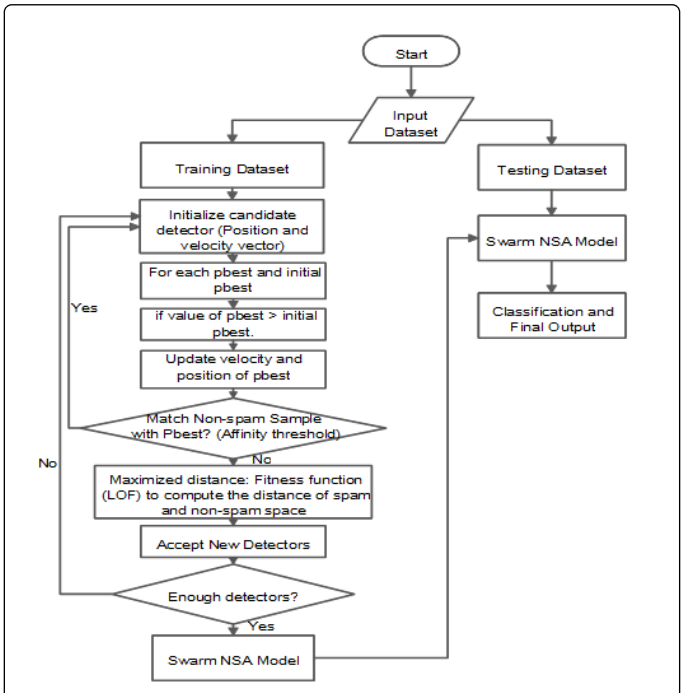

Figure 1: Framework of the Proposed Swarm Negative Selection Algorithm Model

Step 1: Define a stable behaviour and activities of a system as non-spam space (normal pattern) as shown in equation (8).

Step 2: Equation (9) and (10) initializes both position and velocity of particle swarm optimization.

Step 3: Calculate both reachability distance and the local outlier factor for each candidate detector as shown in equation (14) and (15)

Step 4: Update each candidate detector position and velocity with equation (12) and (13)

Step 5: Implement the distance measure in equation (4) and threshold value in equation (5) to determine the pbest similarity in the non-spam space region S. If pbest did not match S, it is a valid detector.

Step 6: Continuous generation and matching of pbest against S is observed for changes, deviation of the system may occur if pbest matchesS. Pbest is not meant to match S.

Step 7: After maximum coverage in the spam space, the testing set is employed for evaluation.

Computation of fitness function

The local outlier factor (LOF) was employed to calculate the fitness function in quest for a purely normal data that will efficiently train our model. An outlier can be defined as a data point that is not the same as the remaining data with respect to some measure. It is employed as a fitness function for the generation of unique features in the spam space. The technique will model the data point with the use of a stochastic distribution [**8**] and the point is determined to be an outlier base on its relationship with the model. The outlier detection algorithm proposed as fitness function in this study of spam detection generation is very unique in computing the full dimensional distance from one point to another while computing the density of local neighbourhood.

Let's assume *k* distance (i) be the distance of the candidate detector or particle (*i*) to the nearest neighborhood (non-spam).

Set of nearest neighbor (non-spam element) includes all particles at this distance.

Set S of nearest neighbor is denoted as N_k (i)

This distance defines the reachability distance.

Reachability-distance (i, s) = max{k – distance (s) ,d(i, s)}

The local reachability distance is then defined as:

$$lrd(i) = 1 / \left(\frac{\sum s \in Nk(i) reachability - distance_k(i,s)}{|NK(i)|} \right) \quad (14)$$

Equation (14) is the quotient of the average reachability distance of the candidate detector *i* from the non-spam element. It is not the average reachability of the neighbor from but the distance from which it can be reached from its neighbor. We then compare the local reachability density with those of its neighbor using the equation below:

$$LOF_K(i) = \frac{\sum s \in Nk(i) \frac{lrd(s)}{lrd(i)}}{|Nk(i)|} = \frac{\sum s \in Nk(i) lrd(s)}{|Nk(i)|} / lrd(i) \quad (15)$$

Equation (15) shows the average local reachability density of the neighbour divided by the particle own local reachability density. In this scenario, values of particle approximately 1 indicates that the particle is comparable to its neighbour (not an outlier), value below 1 indicates a dense region (which will be an inlier) while value larger than 1 indicates an outlier. The major idea of this technique is to assign to each particle degree of being an outlier. The degree is called the local outlier factor (LOF) of the particle. The methodology for the computation of LOF's for all particles is explained in the following steps:

The process of calculating the new velocity and particle position with fitness function is required once a new particle is installed until specific termination criteria are reached. From equation (8) of the normalized non-spam space, the non-spam space is represented in equation (1) with radius Rs in section 3.2.1. Computing the generation of candidate detector of particle swarm optimization in the spam space is as shown in section 3.1. If detector dj match the non-spam sample, it will be discarded. This will not stop the generation of detector until the required detector set is reached and the required spam space covered. After the generation of detectors in the spam space, the generated detectors can then monitor the status of the system. If some other new email (test) samples matches at least one of the detectors in the system, it is assume to be spam which is abnormal to the system but if the new email (test) sample does not match any of the generated detectors in the spam space, it is assume to be a non-spam email.

Empirical study and dataset analysis

Spam base dataset was required for the research. The entire dataset was divided using stratified sampling approach into training and testing set. 70% of the entire dataset was used for training and 30% of the remaining dataset was used for testing the model. The corpus bench mark is obtained from spam base dataset which is an acquisition from email spam messages. In acquiring this email spam message, it is made up of 4601 messages and 1813 (39%) of the message are marked to be spam messages and 2788 (61%) are identified as non-spam and was acquired by [**9**]. The Instances or features are represented as 58-dimensional vectors. In the corpus of 58 features, 48 of the features of the corpus is represented by words generated from the original messages with the absence of stop-list or stemming and they are considered and enlisted as most unbalanced words for the class spam. The remaining 6 features are the percentage of manifestation of the special characters ";", "(", "[", "!", "$" and "#". Some other 3 features is a representation of various measure of manifestation of capital letters that exist in the text of the messages. Lastly, is the class label in the corpus; it gives the condition of an instance to be spam or non-spam by 1 and 0 representation. Spam base dataset is among one of the best test bed that performs good during learning and evaluation techniques.

Performance Evaluation Measure, Experimental results and discussion

Different measures can be use to evaluate the accuracy and performance of NSA and NSA-PSO models. To evaluate and compare the performance and accuracy of these models, statistical quality measures used in machine learning and data mining journals were employed. They are sensitivity (SN), specificity (SP), positive prediction value (PPV), accuracy (ACC), negative prediction value (NPV), correlation coefficient (MC) and f-measure (F1).

(i) Sensitivity (*SN*): The *SN* measures the proportion of positive pattern that are correctly recognized as positive.

$$SN = \frac{TP}{TP + FN} . \quad (16)$$

(ii) Specificity (*SP*): The *SP* measures the proportion of negative pattern that are correctly recognized as negative

$$SP = \frac{TN}{TN + FP} \quad (17)$$

(iii) Positive prediction value *(PPV)*: *PPV* of a test gives a measurement of the percentage of true positives to the overall number

of patterns that are recognized to be positive. It measures the probability of a positively predicted pattern as positive

$$PPV = \frac{TP}{TP + FP} \qquad (18)$$

(iv) Negative prediction value (NPV): NPV of a test also gives the measurement of percentage of true negative to the overall number of patterns recognized to be negative. It measures the probability of a negatively predicted pattern as negative.

$$NPV = \frac{TN}{FN + TN} \qquad (19)$$

(v) Accuracy (Acc): Acc measures the percentage of samples correctly classified

$$Acc = \frac{TP + TN}{TP + TN + FN + FP} \qquad (20)$$

(vi) Correlation Coefficient (CC): CC is use as a measure of the quality of binary (two class) classification in machine learning.

$$CC = \frac{[(TP)(TN) - (FP)(FN)]}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \qquad (21)$$

(vii) F-measure (F1): It is a measure that combines both positive predictive value and sensitivity. The positive predictive value and sensitivity are evenly weighted.

$$F1 = 2.\frac{(Positive\ prdictive\ value)(Sensitivity)}{Positive\ prdictive\ value + Sensitivity} \qquad (22)$$

In the evaluation equation above, TP is the number of true positive, TN is the number of true negative, FN is the number of false negative and FP is the number of false positive.

At 100, 200, 300, 400 and 500 generated detectors with threshold value of 0.4, Figure 2 gives summary and comparison of results in percentage for NSA and SNSA model.

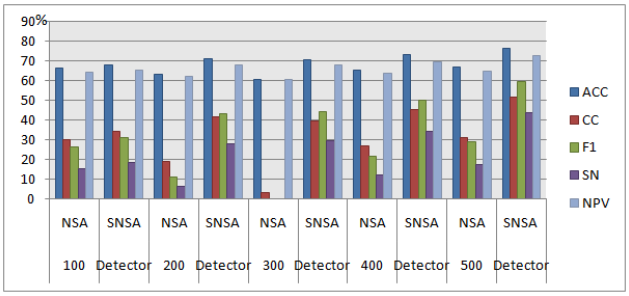

Figure 2: Result of negative selection algorithm (NSA) and swarm negative selection algorithm model (SNSA)

**Note: ACC=Accuracy, CC=Correlation coefficient, F1=F measure, SN=sensitivity, PPV=Positive prediction value, SP=Specificity and NPV= Negative prediction value.

Accuracy measures the percentage of sample that is correctly classified. It can be observed that the proposed swarm negative selection algorithm (SNSA) model performs better than negative selection algorithm (NSA) model. The Figure 2 shows best accuracy at 500 generated detectors with threshold value of 0.4. Accuracy for negative selection algorithm is at 66.98% while the swarm negative selection algorithm is at 76.25%. Other measuring standard are as represented in the figure above.

The Average accuracy of the standard negative selection algorithm is at 65.147%, the improved swarm negative selection algorithm model is at 70.48%. At 8000 generated detectors with threshold value of 0.4, accuracy for negative selection algorithm is 68.863% while improved swarm negative selection algorithm is at 82.69%.

Conclusion and Recommendation

In this research, a new improved model that combines negative selection algorithm (NSA) with particle swarm optimization (PSO) has been proposed and implemented. The uniqueness of this model is that PSO was implemented at the random generation phase of NSA. The detector generation phase of NSA determines how robust and effective the algorithm will perform. The new model is called swarm negative selection algorithm (SNSA). The implementation of PSO with its fitness function no doubt improved the detector generation phase of NSA. In totality, the empirical report as shown the superiority of the proposed SNSA improved model over the NSA model. The proposed improved systems will be useful in other applications since negative selection algorithm solves a vast number of complex problems. This research should be viewed as an improvement in the field of computational intelligence. Future work will further improve the performance accuracy of the proposed swarm negative selection algorithm model by applying independent radius to each candidate detectors.

References

1. Xuesong Yan, Wei Chen, Qinghua Wu, Hanmin Liu (2013) Data Classification Algorithm Based on Differential Evolution Algorithm. Int J Dig Cont Tech App 7: 406-413.
2. Awad WA, ELseuofi SM (2011) Machine Learning Methods for Spam E-mail Classification. Int J Com Sci InfTech 3: 1.
3. Sirisanyalak B, Sornil O (2007) An artificial immunity-based spam detection system. In: IEEE Congress on evolutionary computation, CEC 3392-3398.
4. Wenqing Z, Zili Z (2005) An email classification model based on rough set theory. In: Proceedings of the International conference on active media technology 403-408.
5. Mohammad Adel Hamdan, Abu. ZR (2011) Application of genetic optimized artificial immune system and neural networks in spam detection. J App Soft Computing 11: 3827-3845.
6. Duolin Liu (2013) Research on Sentiment Classification of Chinese Micro Blog Based on Machine Learning. Int J Dig Cont Tech App 7: 395- 402.
7. Tan Li (2012) Research on Fault Detection Algorithm for Frequency Conversion Hydraulic System based on the PSO-BP. Int J Dig Cont Tech App 6: 634-641.
8. Sajesh TA, Srinivasan MR (2012) Outlier detection for high dimensional data using the Comedian approach. J STAT COMPUT SIM 82: 745-757.
9. Mark Hopkins, Erik Reeber, George Forman, Jaap Suermondt (1999) "Spam Base Dataset," Hewlett-Packard Labs, USA.