

Artificial Intelligence Techniques for Enhancing Smartphone Application Development on Mobile Computing

<https://doi.org/10.3991/ijim.v14i17.16569>

Dr. S. V. Manikanthan ^(✉), Dr. T. Padmapriya
Melange Academic Research Associates, Puducherry, India
s.vmanikanth@gmail.com

Azham Hussain
Universiti Utara Malaysia, Sintok, Malaysia

Thamizharasi. E
Melange Academic Research Associates, Puducherry, India

Abstract—Nowadays, Artificial Intelligence is being integrated into the modern innovations, including mobile, Electronic gadgets and as well as our daily lives. The smartphones are becoming a crucial and indistinguishable part of modern life. Whether that be in terms of speech, prototype, efficiency, features, quality and so forth, together all system requirements are provided in one machine. Researchers and innovations analysts are making advances in mobile computing with the excellent technologies. While Artificial Intelligence as a commercial product has been directly accessible. In this, both corporations and violent offenders take benefit of emerging technologies and advances. Cyber-security specialists and authorities have predicted there have been high possibilities of cyber-attacks. There's really, besides that, a need to improve quite advanced and powerful data security processes and software to protect all fraudulent activities and threats. The objective of this study is to introduce latest developments of implementing Methodologies to mobile computing, to prove how such techniques could become an efficient resource for data security and protocols, and to provide scope for future research.

Keywords—Artificial Intelligence, Smartphones, Mobile Computing, Data Security.

1 Introduction

Mobile computing is the cloud-based services that are available in a mobile environment. Mobile computing is a human – machine interface in which a device is supposed to be moved throughout daily operation, enabling data, voice and video to be transmitted [1,27]. Mobility relates to usability or probability of switching to various sites, utilizing specific forms of handheld devices over several periods. In

contemporary society, the Internet and new technology have increasing the importance of smartphone devices. As well as Mobile security or security of smart phones is now becoming particularly significant in mobile computing.

Mobile protection is an environment which has not been provided much consideration by smartphone users to date [2]. That's evolving now, though, because of a number of reasons that exist in the smartphone industry. Traditionally, smartphone consumers utilized their mobile apps mainly for voice messaging, without any kind of wireless data operation at all. Today, smartphones are commonly used and have plenty of features including such personal computers (PCs) and, in turn, have more than enough connectivity choices like 3G, 4G, Wi-Fi, GPS, LTE, NFC, and Bluetooth [3]. This wide variety of good features has given rise to highly utilization of Smartphones which become ideal targets for malicious as a consequence still.

Security is an essential concern and must be properly considered in all fields of applications, particularly in mobile computing, as the mobile user can sometimes recognize several security issues that are not really known to the typical internet user. Such attacks include losing or robbing the important and confidential data of smartphone users that were collected on their apps or that cybercriminals could misuse it and some problem detection [4].

Similarly, the current security measures are not sufficient to prevent cyber threats due to the emergence of malware cyber-attacks mobile devices. When the environment is becoming more digital, it is essential that in most of the defensive and offensive security initiatives, AI does have a significant effect on enhancing software security and cyber security requirements [5]. DARPA 's Cyber Dataset, which would have been trained to support build systems for identifying, testing and patching vulnerabilities in software before they've been attacked, has already seen the benefits of using AI to enhance security. As the mobile security and computing have improved efficiency, versatility, and availability, AI has the ability to lead on a modern generation of benefits and risks [26,28].

2 Literature Review

The authors Mitchell et al. [6] emphasizes machine learning as an in-computer science field of existing approaches that function better with practice. This means that software applications use their previous knowledge with problem-solving tasks to enhance their skills. But for the mentioned software applications, definition does not provide the idea of acquiring information.

The authors Wang et al. (2008) [7] suggested that Heuristic Technology applies the future of anti-virus identification technology, meaning "the training and expertise with using certain technique to measure and smartly evaluate protocols to identify the unknown malware by certain guidelines while detecting". Chen (2008) developed NeuroNet-a neural network framework that helps to collect data more efficiently, manages network control device operations, scans for anomalies, warnings, and

establishes threats. Results revealed that NeuroNet is impactful toward decentralized cyber - attacks focused at low-rate TCP [8].

In the authors of Jongsuebsuk et al. (2013) suggested a Fuzzy Genetic Algorithm based IDS network. Fuzzy rules have been used to distinguish data from network attacks, while genetic algorithms automate making new fuzzy logic to get the accurate solutions. The outcomes of the study demonstrate that the suggested IDS can identify network threats and risks, when the data arrives at the sensor module with a detection accuracy of over 97.5 per cent [9].

Benaicha et al. (2014) provided a framework for detecting intrusion on the system, Genetic algorithm approach with enhanced performance and had something user used to automate scenario searches in reporting entities and also provide sufficient computation time a selection of future threats [10]. They used genetic algorithm methodology since efficiency is boosted and brings down the false alarms.

For risk control in a particular situation, the authors Padmadas et al. (2014) proposed a structured genetic algorithm-based malware detection method to assess if they are really genuine or fraudulent based on available data sources, data privacy and security [11]. The simulation results demonstrate which R2L threats are effectively detected with 90 percent accuracy by the suggested technique.

Machine learning methods are reinforcement learning and case-based learning in smart phones function better. Such methods are often important for various types of language expressions in the smart phones' application network [12]. An m - learning framework was created using existing web tools, which participated in active learning according to customer needs in the paper [13].

Mobile devices are operators who move from one host to another in a network for efficient performance of specific tasks. A mobile agent is an informed decision-maker about its schedule and improves the decision maker(s) according to the data obtained as it moves from one host to another [14]. Machine learning classification strategies including Naïve Bayesian and ANN are effective for classification tasks and advanced software entity extraction. Mobile apps provide a wide variety of services over other communication protocols for cellular networks [15]. Smart phones are likely to be vulnerable to security risks and machine learning techniques such as K-Nearest Neighbour, Bayesian Networks and Random Forests are important for smart phones intrusion detection. Machine-learning methods are cost effective in many aspects for smart phones.

3 Artificial Intelligence in Mobile Computing

The implementation of Artificial Intelligence into security devices can be used to reduce the ever-increasing security vulnerabilities that major corporations' experience. Mobile Computing and Artificial Intelligence were originally regarded as two different components [16]. AI researchers were ready to develop software to decreased human intelligence, although security analysts were trying to repair the data security and privacy breaches.

Smartphone apps across the enterprise using both machine learning and artificial intelligence (AI) are even more commonly included as data collection, storage features are rising and computing resources is boosting. The massive quantity of data is hard for some people to maintain in actual environments. The extensive knowledge can likely be reduced in fractions of a second with the advances in machine learning as well as Artificial Intelligence, as a consequence of something that the organization can quickly access also rebuild from the security risk [17]. Mobile technology's emergence has driven the growth of artificial intelligence.

The study of Machine Learning has emerged from the area of artificial intelligence, which seeks to use computers to imitate intelligent human skills. Therefore, machine learning models perform perfectly with smart software in enhancing the performance and reliability of smart decision-making processes. For mobile devices such as smart cards, smart phones, sensors, handheld and automotive computing systems, AI technologies have also proved to be successful. A few other significant mobile device machine learning techniques involve Sensor-based behavior recognition, mobile character recognition, smart phone vulnerability scanning, language comprehension etc.

3.1 Genetic Algorithm (GA)

A Genetic Algorithm (GA) is a search tool that imitates evolutionary theory. The algorithm develops until the issue is properly solved by fitting solutions in a continuing sample and transferring their characteristics to descendants that substitute weaker responses [18, 19]. For instance, every other possible solution is encrypted as a given sequence, termed a chromosome. Subsequent groups are referred to as generations.

Unknown sample $U(0)$ is automatically chosen. Then $U(n)$ generates $U(n+1)$ by selecting and reproducing. For reproducing and generating new chromosomes a number of individuals are chosen. Selection depends on the fitness of feature selection, e.g. frequency to an ideal solution, including by availability of slot machines and probabilistic measuring. Slot machine evaluation consists of selecting a parent with a determined confidence interval from the behavioral fitness (f) by,

$$F = \frac{f}{\sum_n f} \quad (1)$$

Probabilistic analysis corresponds $P_a = GA(zf_n)$ to the evolutionary process 'e' of the population's P_i individuals, while GA helps to balance its claim to the neighbouring unit. Also, every individual is chosen as a parent P_a -times.

The initial phase in the Genetic programming is to describe the issue as genetic labeled. In other words, it is essential to identify genetic frameworks, like genes, chromosomes and population. Similarly, features for fusion, mutation, and fitness are being established. GA is in existence for several decades. Evaluation of a chromosome requires experienced classifiers about its precision on every decade.

3.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) is the emerging supervised technique in research for machine learning. SVM contains the keyword margin which is either side of hyper plane separating 2 categories of information [19]. A decision plane is an axis separating a collection of attributes that have various groups in the class. SVM utilizes a set of supervised learning processes. Generating the greatest distance between the hyper plane splitting and the occurrences on either side of it by optimizing the margin decrease the absolute limit on the error rate assumed [20]. The basic structure of Support Vector Machine (SVM) is shown in Figure [1].

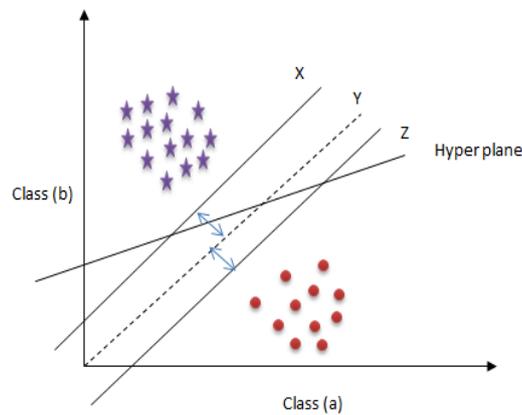


Fig. 1. SVM Classifier

The basic SVM model is the category that is implemented to decision boundary selections. After that, there are 2 types and one classifier class is described by superior quality and another classifier class is defined by 1. If the m -dimensional x transactions can be excluded uniformly for a two-class problem, the whole class is distinguished with the interpretation in following equation.

$$S(N) = h^i n + y \quad (2)$$

From the Eq. (2), where n is a hyper plane value, v is a vector in the SVM function space and y is a bias. The first for maximizing the support vectors between the 2 classifications term in Equation (3) should also be reduced and the second inequality for effective analysis of all extracts shall be given to any other learning algorithm.

$$\frac{1}{2} |h|^2 \min \quad (3)$$

For both types, a single hyper plane could be decoded as $hn_i + y \geq 1$, Class $C_i = 1$ and $hn_i + y \leq 1$ for class $C_i = -1$. Some might integrate equations to provide the below equations.

$$C_i (h^i n + y) \geq 1, \text{ for training samples} \quad (4)$$

3.3 Artificial Neural Network (ANN)

An artificial neural network is a system made up of components and weights implemented in order to model human neurons. Developers created a neural network and correspondingly wrote the algorithms utilizing 2 hidden layers to categorize an interaction for both inputs and outputs that seemed to be in figure [2]. The network is fully linked to weights w_{Ljk} beginning with component j and ending in component k in which L is the number of the layer. The basic sigmoid formula is used in the activation function at (4)

$$S(f) = \frac{1}{1 + e^{-h(n)}} \tag{5}$$

The node outputs are determined by modifying $h(n)$ with threshold and conditional function.

$$T_b^n = \frac{1}{1 + e^{-(NN \frac{n}{b} - \delta \frac{n}{b})}} \tag{6}$$

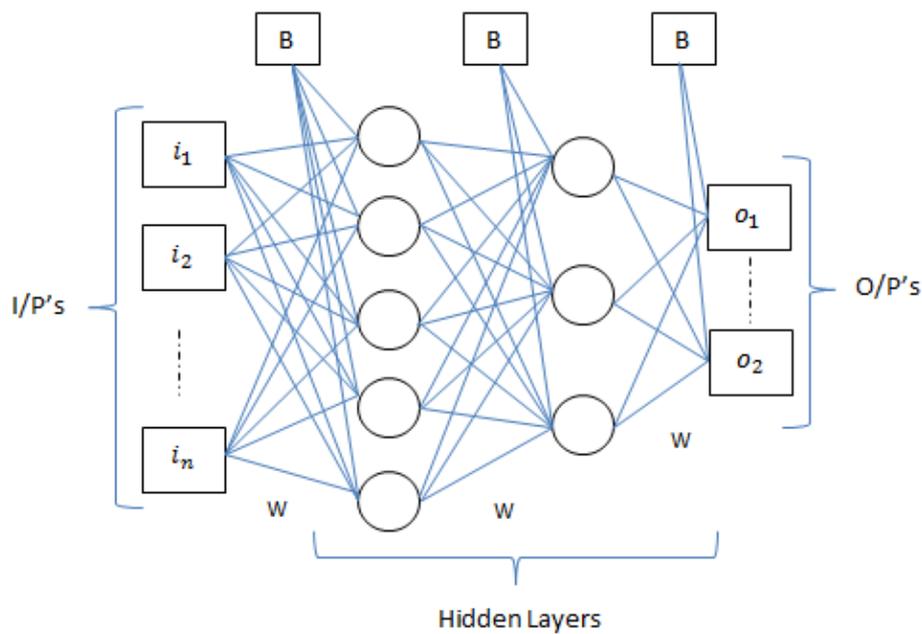


Fig. 2. Artificial Neural Network

3.4 Neuro- Fuzzy Classifier

Neuro-fuzzy Classifier system is essentially a neural network being used in a Takagi – Sugeno input vector to construct objective function, rules and output

features rather than modifying weights [21]. Furthermore, NFC is typically used for estimating features and predicting a set of fuzzy rules with a suitable Objective Functions (OF). Just like ANN, an NFC begins with training and for these simulations the network is equipped with enhancing the accessibility with objective function and with single output. For matrix [X] variables, objective functions are developed for nodes. After testing, the developed fuzzy logic - based framework can be used for upcoming login credentials in testing stage.

NFC combines a neural network with fuzzy logic to enhance accurate predictions in order to separate from the benefits of all of the above [22]. The fuzzy inference used by NFC consists of 3 layers: a fuzzy set of laws, a logic system and a repository dataset. The basis of the fuzzy law contains a collection of if – then fuzzy laws. The reasoning approach incorporates the inference process and the dataset describes the participation roles in the fuzzy laws [23]. In contrast, NFC uses the neural network to fine-tune the objective function and relevant features that manage chosen datasets [24].

Two inputs of the FIS, a and b, and one output, c. Then the fuzzy if – then law focused on Takagi and Sugeno.

$$\text{if } a \text{ is } U_n \ \& \ V_n, \text{ then } Nf_n = p_n a + q_n b + s_n \quad (7)$$

In which vector p, q and s are the variables defined as the rule.

Framework of NFC consists of several layers, as seen in the following below.

- a) The nodes in that kind of layer use the objective function to generate input dependent class labels. The rule's loading effectiveness is the output of every node in that layer.
- b) The nodes in this interface are Takagi – Sugeno form Fuzzy rules.
- c) Every node receives feedback from the corresponding node in layer 1 and measures its variance to decide the loading power of the Fuzzy law s_n .

$$Z_n = f(\delta_n(a_{n1}), \delta_n(b_{n2})) = \delta_n(a_{n1}) \cdot \delta_n(b_{n2}) \quad (8)$$

Where, $\delta_n(a_{n1})$ and $\delta_n(b_{n2})$ corresponds to the ambiguous feature values of the a_{n1} and b_{n2} inputs separately.

- d) In this neuron the nodes calculate from the destination nodes in layer 2, the regularization of the loading force of the rule. The single node conducts layer 3 measurement for all outputs.

4 Sherlock Dataset Description

A long-term samples of smartphone sensors with a large time resolution. The dataset also provides standard labels that identify malicious behaviour working on smartphones. The registry actually comprises 10 billion existing data from 30 users compiled throughout a 2-year span and a further 20 users during a 10-month intervals. The main objective of the dataset is to support protection experts and scientific

researchers establish new techniques of indirectly identifying mobile security vulnerabilities. In general, from data that can be accessed without the privileges of the developer. These datasets can also be used for analysis in databases that are not directly relevant to protection [25]. For example, context-conscious recommender systems, predictor of incidents, user personalization and knowledge, predictor of location, etc., the dataset often provides incentives not present in other datasets. The dataset, for example, includes the Ethernet and signal intensity of the attached Wi-Fi Access Point (AP) that is collected once per second for several months.

The dataset SherLock is divided into raw data, from each SherLock probe. Data archives from such sensors cannot be preserved in their parent probes' data tables. The data from these sensors were also contained in different tables. Each table of data has the Uuid, Userid and Version fields. Uuid is the Unix Millisecond period point of the record being stored. Userid is a unique volunteer identifier to which the record belongs. Finally, is the product development file for the agent. Table 1 lists the 15 datasets of the SherLock, with their number of documents from August 2016.

	Data Table	Number of Records
PUSH	<i>Call Log</i>	443,175
	<i>SMS Log</i>	245,693
	<i>Screen Status</i>	2,608,766
	<i>User Presence</i>	685,910
	<i>Broadcast Intents</i>	95,471,166
	<i>App Packages</i>	108,612
	<i>Moriarty</i>	650,625
PULL	<i>T0</i>	242,762
	<i>T1</i>	14,050,156
	<i>WiFi</i>	54,654,980
	<i>Bluetooth</i>	2,945,238
	<i>T2</i>	43,383,170
	<i>T3</i>	85,861,126
	<i>T4</i>	180,012,794
	<i>Application</i>	9,271,351,994
Total:		9,752,716,167

Fig. 3. The data tables and data record numbers in the SherLock dataset

5 Experimental Results

Machine learning algorithms, such as Support Vector Machine, Genetic Algorithm, Neuro-Fuzzy Classifier and Artificial Neural Network are used for data sets comparison and evaluation. Create an uncertainty matrix to measure the precision of the specified experiments to test the results effectiveness. The parameters used for the uncertainty matrix followed. Figure 3 as shows the confusion matrix of hybrid classifier algorithm. The experimental analysis was carried out on a dataset of around millions of APKs based on five classes: malware or harmful programs. The APKs are modified to obtain characteristics. A CSV is created due to the presence of 99 features with training samples as Malware (0) and Goodware (1). The Error rates of Malware detection on smartphone using classifier Algorithm are shown in figure 4. The main aim of this analysis is to find the optimal subset of features for which Genetic Algorithm could be used. The regressive features selected by Genetic Algorithm are fed as input to train classifiers for Support Vector Machine, Artificial Neural Network and Neuro-Fuzzy Classifiers. The algorithms are tested on Intel(R) Core(TM) i3-3110M CPU@ 2.40GHz 2.19GHz, 4GB RAM, 64-bit operating system. The Statistical Analysis of classifier Algorithm is shown in figure 5. The implementation are done by RStudio Version 1.1.456 – © 2009-2018 RStudio, Inc.



Fig. 4. Confusion Matrix

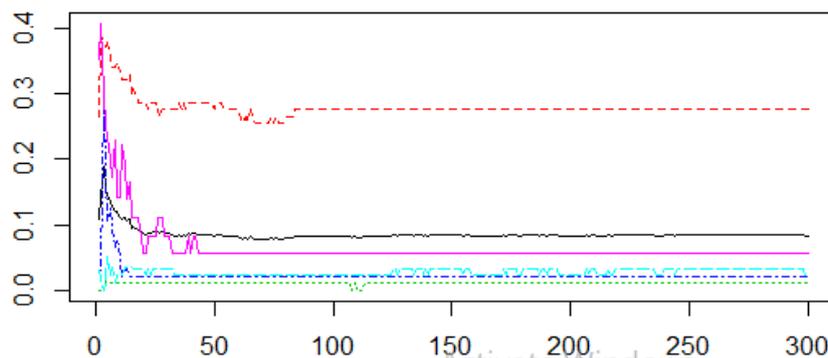


Fig. 5. Error Rate of Malware Detection on Smartphones

	Class: 1	Class: 2	Class: 3	Class: 4	Class: 5
Sensitivity	0.7766	1.0000	0.9783	1.0000	1.00
Specificity	1.0000	0.9936	1.0000	0.9325	1.00
Pos Pred Value	1.0000	0.9780	1.0000	0.8091	1.00
Neg Pred Value	0.9358	1.0000	0.9935	1.0000	1.00
Prevalence	0.2350	0.2225	0.2300	0.2225	0.09
Detection Rate	0.1825	0.2225	0.2250	0.2225	0.09
Detection Prevalence	0.1825	0.2275	0.2250	0.2750	0.09
Balanced Accuracy	0.8883	0.9968	0.9891	0.9662	1.00

Fig. 6. Statistical Result of Classifier Algorithm

The visual representations of Classifier Algorithm graph are shown in figure 6. Dendrogram are helps to view the group of clusters in a visual representation, where the below graph shows the 5 classes of attacks are separated from the large data of Sherlock Datasets. The x axis shows each classes of attacks on Smartphone Malware Detection. The individual classes of attacks are arranged along the bottom of the dendrogram and referred to as leaf nodes. The y axis shows the distance of Clustering tree of the attacks.

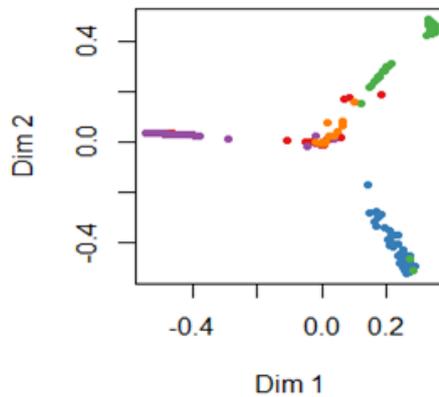


Fig. 7. Visual Representation of Classifier Algorithm

5.1 Performance metrics

Machine learning classifier efficiency is measured based on precision, accuracy, and recall determined by calculating true positive, false positive, true negative, and false negative metrics.

5.2 Accuracy

Accuracy is the most important basic measure of the performance of a learning method. It gives the possibility that the algorithms can correctly predict positive and negative instances and is computed as:

$$\text{Accuracy} = \frac{TP+TN}{P+N} \quad (9)$$

5.3 Precision

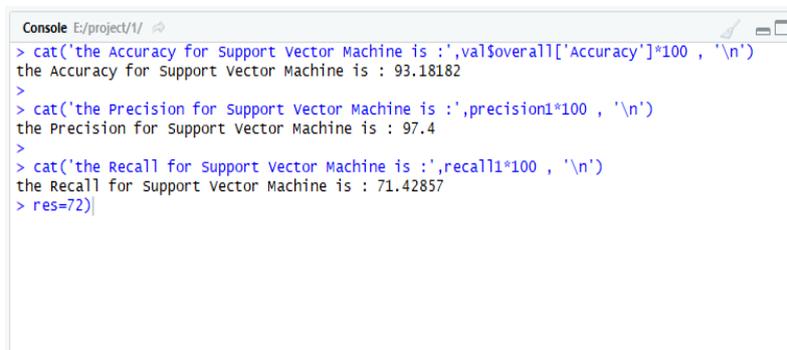
Precision is defined as the proportion of positive predictions which made by the classifier that are true. The precision rate directly affects the performance of the system. The precision is calculated by

$$\text{Precision} = \frac{TP}{TP+FP} \quad (10)$$

5.4 Recall

The Recall rate is also an important value for measuring the performance of the detection system and to indicate the proportion of instances belonging to the positive rate that are correctly predicted as positive. This is also known as sensitivity or true positive rate.

$$\text{Recall} = \frac{TP}{P} \quad (11)$$



```
Console E:/project/1/
> cat('the Accuracy for Support Vector Machine is :',val$overall['Accuracy']*100 , '\n')
the Accuracy for Support Vector Machine is : 93.18182
>
> cat('the Precision for Support Vector Machine is :',precision1*100 , '\n')
the Precision for Support Vector Machine is : 97.4
>
> cat('the Recall for Support Vector Machine is :',recall1*100 , '\n')
the Recall for support Vector Machine is : 71.42857
> res=72)
```

Fig. 8. Evaluation Metrics of Support Vector Machine

```

Console E:/project/1/
> # Accuracy for 'ANN'
> #-----
>
> cat('the Accuracy for Artificial Neural Network is :', as.numeric(as.character(unlist(accuracy)))*100, '\n')
the Accuracy for Artificial Neural Network is : 96
>
> cat('the Precision for Artificial Neural Network is :', as.numeric(as.character(unlist(Precision)))*100, '\n')
the Precision for Artificial Neural Network is : 98.56
>
> cat('the Recall for Artificial Neural Network is :', as.numeric(as.character(unlist(recall)))*100, '\n')
the Recall for Artificial Neural Network is : 88.27
> |
    
```

Fig. 9. Evaluation Metrics of Artificial Neural Network (ANN)

```

Console E:/project/1/
> # Accuracy for 'NFC'
> #-----
>
> cat('the Accuracy for Neuro-Fuzzy is :', as.numeric(as.character(unlist(accuracy)))*100, '\n')
the Accuracy for Neuro-Fuzzy is : 97.58
>
> cat('the Precision for Neuro-Fuzzy is :', as.numeric(as.character(unlist(Precision)))*100, '\n')
the Precision for Neuro-Fuzzy is : 99.35
>
> cat('the Recall for Neuro-Fuzzy is :', as.numeric(as.character(unlist(recall)))*100, '\n')
the Recall for Neuro-Fuzzy is : 91.27
> |
    
```

Fig. 10. Evaluation Metrics of Neuro-Fuzzy Classifier (NFC)

Table 1. Performance Factors for Classifier Algorithm

S. No	Performance Metrics	Neuro-Fuzzy Classifier (NFC)	Support Vector Machine (SVM)	Artificial Neural Network (ANN)
1	Accuracy	97.58%	93.18%	96%
2	Precision	99.35%	97.4%	98.56%
3	Recall	91.27%	71.42%	88.27%

In table 2, shows the comparison results of classifier algorithm of Neuro-Fuzzy Classifier (NFC) with Support Vector Machine (SVM) and Artificial Neural Network (ANN) using the evaluation metrics of Accuracy, Precision and Recall.

Thus, the above table represents the performance metrics of Accuracy, Precision and Recall for Classifier Algorithm such as NFC, SVM and ANN where the NFC have outperformed in characterizing the intrusions with 97.58% of accuracy, 99.35% of precision and has got 91.27% of recall followed by SVM and ANN are shown in figure 9, because the NFC has reduced the training time. The Support Vector Machine (SVM) has got the 93.18% of accuracy, 97.4% of precision and 71.42% of recall are

shown in figure 7 and the Artificial Neural Network (ANN) has got the 96% of accuracy, 98.56% of precision and 88.27% of recall are shown in figure 8.

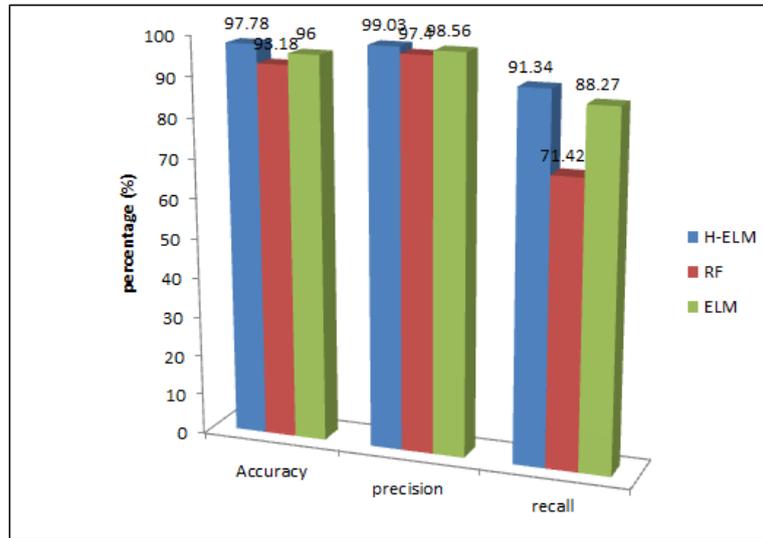


Fig. 11. Comparison Graphs of Performance Metrics for Classifier Algorithm

Thus, the Figure 10 shows the Comparison Graphs of Performance metrics for Classifier Algorithm. The proposed algorithm of NFC has got the high accuracy, where the accuracy also termed as false alarm (false positives) of Intrusion Detection System when compared to other Classification Algorithms of SVM and ANN.

6 Conclusion

In this study, Artificial Intelligence based Mobile Computing was evaluated for a performance comparison of unknown Android malware detection on mobiles for enhancing the smartphone Application development, using different set of features. Increasing efficiency of the classifier is measured by its accuracy, precision and Recall. The findings show better output of Neuro-Fuzzy classifier. The study found enhanced performance in terms of accuracy and false positive rate, with additional error rate. This experiment showed that hybrid classifier of Neuro-Fuzzy can be great choices for effective malware detector execution. Feature extraction for datasets and performance evaluation of android malware mobile applications shall be determined in future study.

7 References

- [1] Mollah, M. B., Azad, M. A. K., & Vasilakos, A. (2017). Security and privacy challenges in mobile cloud computing: Survey and way ahead. *Journal of Network and Computer Applications*, 84, 38-54. <https://doi.org/10.1016/j.jnca.2017.02.001>
- [2] Gangula, A., Ansari, S., & Gondhalekar, M. (2013, November). Survey on mobile computing security. In *2013 European Modelling Symposium* (pp. 536-542). IEEE. <https://doi.org/10.1109/ems.2013.89>
- [3] Hur, J. B., & Shamsi, J. A. (2017, December). A survey on security issues, vulnerabilities and attacks in Android based smartphone. In *2017 International Conference on Information and Communication Technologies (ICICT)* (pp. 40-46). IEEE. <https://doi.org/10.1109/icict.2017.8320163>
- [4] Alotaibi, E. F., AlBar, A. M., & Hoque, M. R. (2016). Mobile computing security: issues and requirements. *Journal of Advances in Information Technology* Vol, 7(1).
- [5] AI: Increasing the Intelligence on Smartphones. <https://www.aithority.com/guest-authors/ai-increasing-the-intelligence-on-smartphones/>
- [6] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260. <https://doi.org/10.1126/science.aaa8415>
- [7] Wang, X. B., Yang, G. Y., Li, Y. C., & Liu, D. (2008, September). Review on the application of artificial intelligence in antivirus detection system i. In *2008 IEEE Conference on Cybernetics and Intelligent Systems* (pp. 506-509). IEEE. <https://doi.org/10.1109/iccis.2008.4670733>
- [8] Chen, Y. (2008, January). NeuroNet: Towards an Intelligent Internet Infrastructure. In *2008 5th IEEE Consumer Communications and Networking Conference* (pp. 543-547). IEEE. <https://doi.org/10.1109/ccnc08.2007.126>
- [9] Jongsuebsuk, P., Wattanapongsakorn, N., & Charnsripinyo, C. (2013, May). Real-time intrusion detection with fuzzy genetic algorithm. In *2013 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology* (pp. 1-6). IEEE. <https://doi.org/10.1109/ecticon.2013.6559603>
- [10] Benaicha, S. E., Saoudi, L., Guermeche, S. E. B., & Lounis, O. (2014, August). Intrusion detection system using genetic algorithm. In *2014 Science and Information Conference* (pp. 564-568). IEEE. <https://doi.org/10.1109/sai.2014.6918242>
- [11] Padmadas, M., Krishnan, N., Kanchana, J., & Karthikeyan, M. (2013, December). Layered approach for intrusion detection systems based genetic algorithm. In *2013 IEEE International Conference on Computational Intelligence and Computing Research* (pp. 1-4). IEEE. <https://doi.org/10.1109/iccic.2013.6724120>
- [12] Nate Derbinsky, Georg Essl, "Cognitive Architecture in Mobile Music Interactions", NIME'11, Oslo, Norway May-June 2011.
- [13] Alzaabi, M., Berri, J., & Zemerly, M. J. (2010). Web-based Architecture for Mobile learning. *International Journal for Infonomics (IJI)*, 3(1), 207-216. <https://doi.org/10.20533/iji.1742.4712.2010.0022>
- [14] Yang, J., Pai, P., Honavar, V., & Miller, L. (1998, April). Mobile intelligent agents for document classification and retrieval: A machine learning approach. In *14th European Meeting on Cybernetics and Systems Research. Symposium on Agent Theory to Agent Implementation, Vienna, Austria* (pp. 707-712).
- [15] Damopoulos, D., Menesidou, S. A., Kambourakis, G., Papadaki, M., Clarke, N., & Gritzalis, S. (2012). Evaluation of anomaly-based IDS for mobile devices using machine learning classifiers. *Security and Communication Networks*, 5(1), 3-14. <https://doi.org/10.1002/sec.341>

- [16] Bhutada, S., & Bhutada, P. (2018). Applications of Artificial Intelligence in Cyber Security. *International Journal of Engineering Research in Computer Science and Engineering*, 5(4), 214-219.
- [17] Chaudhar, A., & Kolhe, S. (2013). Machine Learning Techniques for Mobile devices-A Review.
- [18] Chen, S. H., Jakeman, A. J., & Norton, J. P. (2008). Artificial intelligence techniques: an introduction to their use for modelling environmental systems. *Mathematics and computers in simulation*, 78(2-3), 379-400. <https://doi.org/10.1016/j.matcom.2008.01.028>
- [19] Yildiz, O., & Doğru, I. A. (2019). Permission-based android malware detection system using feature selection with genetic algorithm. *International Journal of Software Engineering and Knowledge Engineering*, 29(02), 245-262. <https://doi.org/10.1142/s0218194019500116>
- [20] Chen, Y., Zhao, J., & Li, F. (2018, December). An SVM-Based Recognition Method for Safety Monitoring Signals of Oil and Gas Pipeline. In *IOP Conference Series: Materials Science and Engineering* (Vol. 452, No. 3, p. 032008). IOP Publishing. <https://doi.org/10.1088/1757-899x/452/3/032008>
- [21] Alpar, O. (2015). Intelligent biometric pattern password authentication systems for touchscreens. *Expert Systems with Applications*, 42(17-18), 6286-6294. <https://doi.org/10.1016/j.eswa.2015.04.052>
- [22] Lee, A. H., Kang, H. Y., Hsu, C. F., & Hung, H. C. (2009). A green supplier selection model for high-tech industry. *Expert systems with applications*, 36(4), 7917-7927. <https://doi.org/10.1016/j.eswa.2008.11.052>
- [23] Enck, W. PG-G. (2010). Taintdroid: An information-flow tracking system for real-time privacy monitoring on Smartphones. In *9th USENIX conference on Operating systems design and implementation*.
- [24] Elish, K. O., Shu, X., Yao, D. D., Ryder, B. G., & Jiang, X. (2015). Profiling user-trigger dependence for Android malware detection. *Computers & Security*, 49, 255-273. <https://doi.org/10.1016/j.cose.2014.11.001>
- [25] Sherlock Dataset: <http://bigdata.ise.bgu.ac.il/sherlock/#/>
- [26] Thipsuda Wongkhamdi, Nagul Cooharajanone, Jintavee Khlaisang. (2020). E-Commerce Competence Assessment Mobile Application Development for SMEs in Thailand. *International Journal of Interactive Mobile Technologies*, 14(11), 48-75. <https://doi.org/10.3991/ijim.v14i11.11358>
- [27] Muralidhar Kurni, Madhavi K (2020). Approaches to Address the Operational Limitations of MANETs through Ad Hoc Mobile Cloud Computing Paradigm. *International Journal of Interactive Mobile Technologies*. 14(9), 153-165. <https://doi.org/10.3991/ijim.v14i09.14103>
- [28] Sherine Akkara, Mallikarjuna Sastry Mallampalli, Venkata Surya Seshagiri Anumula. (2020). Improving Second Language Speaking and Pronunciation through Smartphones. *International Journal of Interactive Mobile Technologies*. 14(11), 280-287. <https://doi.org/10.3991/ijim.v14i11.13891>

8 Authors

Dr. S. V. Manikanthan is a Director of Melange Academic Research Associates, Puducherry, India. His area of Research Interest is Wireless Sensor Networks. He has 21 years of experience in Anna University Affiliated Colleges and in Industrial Research Projects. Email: s.vmanikanth@gmail.com

Dr. T. Padmapriya is a Managing Director of Melange Academic Research Associates, Puducherry, India. She obtained her PhD in Pondicherry Engineering College, Puducherry, India. Her area of Research Interest is LTE, Wireless Networks. Email: dr.padmapriyaat@gmail.com

Azham Hussain is the Associate Professor of Software Engineering at School of Computing, Universiti Utara Malaysia, Kedah, Malaysia. He is the founder of Human-Centered Computing Research Group, which is affiliated with the Software Technology Research Platform Center at School of Computing, Universiti Utara Malaysia. Azham Hussain is a member of the US-based Institute of Electrical and Electronic Engineers (IEEE), and actively involved in both IEEE Communications and IEEE Computer societies. Email: azham.h@uum.edu.my.

Thamizharasi E currently is a Junior Research Fellow at the Melange Academic Research Associates, Puducherry, India. She received B.Tech in Computer Science and Engineering from Christ College Engineering & Technology, Puducherry and M.Tech in Distributed Computing Systems (DCS) from Pondicherry Engineering College, Puducherry. Her area of Research includes Machine Learning, Cyber Security, Data Science and Security. Email: thamizhmelange@gmail.com

Article submitted 2020-06-25. Resubmitted 2020-07-24. Final acceptance 2020-07-25. Final version published as submitted by the authors.