[1] **Kalaiselvi. R**

[2] **Jebin Bose S**

MH-ASO Based Deep NNET Classifier Scheme for Effective Android Malware Recognition & Classification Strategy

Abstract: With the emergence of smartphone technology and mobile applications, mobile phones become a most vibrant tool for accessing internet to get several services in just a single click. At the same time, susceptibilities of application are considered as a hazard for the Android device security. Because of this weakness, an attacker could hack the privacy of mobile phone data more easily. The malware application thus performs fraud actions automatically on mobile devices without the knowledge of user. Hence, these attacks are regarded as a major threat to the mobile device security. For detecting the malicious applications that are installed on Android smartphones, a work is proposed to detect Android malware using deep learning-based classification approach. At first, the input android malware dataset is considered as input and the redundant data is removed. The features are extracted and optimal features are selected using Meta-Heuristic Aquila Swarm Optimization strategy (MH-ASO). An Opti Deep Nnet classifier is employed so as to classify the malwares effectively. The proposed classifier is responsible for detecting and classifying android malwares as Adware, Scareware, SMS Malware, and Ransomware. The blowfish-based encoder-decoder is employed so as to protect the data from attackers. By this, the privacy of android device is maintained effectively. Finally, the performance analysis is carried, tested and verified over CICInvesAndMal2019 dataset and the outcomes are compared with traditional methods in terms of accuracy, F-score, precision, recall, ROC plot, and TPR (True positive rate). The analysis shows that the proposed model is effective than others.

Keywords: Android malware, Detection and classification, MH-ASO, Opti Deep Nnet classifier, Blowfish based encoder decoder.

I. INTRODUCTION

At the present time, dependency on smartphone devices is increasing drastically. By the year 2021, number of smartphone device has grasped about 3.8 billion worldwide [1]. Over 72% of these devices runs on Android operating system. In addition, the android users who install antivirus software at their mobile devices are rare. On the other hand, even the users who install them might not effectively utilize it for the detection of viruses. All these factors lead cyber attackers to seek attention on android system because of its huge valuable information and usage worldwide [2]. The malicious actions like malicious detection, solitude theft, remote control, and traffic utilization threatens the privacy and security of infected devices more seriously.

A new investigation characterization and detection by the smartphone devices network behavior is considered so as to guard the mobile device users and the cellular department set-up from the malicious activities [3]. Several kinds of machine learning (ML) techniques and artificial intelligence techniques have been developed so far. Different classifiers such as Decision tree (DT), K-Nearest Neighbor (KNN), Random Forest (RF), and Regression and random tree (RRT) are there in the machine learning based system [4]. Varied range of general malwares and adware are being provided with a mobile malware traffic dataset. In case of huge volume of data availability, then deep learning (DL) models will be effective than the machine learning schemes. however, it needs a choice of DL architecture, processing of feature, extraction, and data hugely. Also, the methods of DL have gained implausible intrudes in the data identification pastures [5], classification and recognition process because of its significant ability. By means of reproducing those success in the detection of malware, several techniques on malware detection needs to enhance the demand of malware forecasters [6].

The rate of android malware is rapidly growing; specifically, use of ever more malicious software technology complication. The existing or outmoded methods of detection expose some issues like low accuracy, and slow detection rate [7]. For this reason, several researchers have solved the issues of android malware detection more recently with the use of machine learning approaches by gaining huge research results. With the advent of deep learning methods, and enhancement of computer computing power, several researchers start to

[1] Associate Professor, Department of Computer Science and Engineering, R.M.K. College of Engineering and Technology, Puduvoyal, Gummidipoondi. Email: kalaiselvir32@gmail.com.

[2] Research Scholar, Department of Computer Science and Engineering, Noorul Islam Centre for Higher Education, Kumaracoil. Email: jebinboses@gmail.com

utilize deep learning methods for the detection of android malware. This work proposes an android detection methodology with the use of artificial intelligence and deep learning schemes [8]. With the rise of deep learning and the improvement of computer computing power, more and more researchers began to use deep learning models to detect Android malware. The residual section of this paper is structured as: an overview on existing works related to deep learning and machine learning based android malware detection is narrated in section II. The proposed work is depicted in section III. The assessment of performance is given in section IV. The conclusion is provided in section V.

II. RELATED WORKS

The related work uncovers the idea which others have found such that the researchers could be beneficial from the efforts made by others. An endeavor has been made in this section for inspecting the traditional works presented before related to the android malware detection using machine or deep learning models.

An effectual machine learning based technique was proposed in [9] for the detection of android malware thus making use of evolutionary GA (genetic algorithm) for the discrimination of feature selection. The features selected by this method are employed for training the ML classifiers and its capability in the identification of malwares is compared before and after the process of feature selection. The outcomes of experiment validates that the GA offers an optimized feature subset selection range thus aiding in decreasing the feature dimension to more than half of feature set that were original.

In the work [10], a De-LADY (deep learning-based detection of android malware with the use of Dynamic features) De-LADY (Deep Learning based Android malware detection using Dynamic features) a robust obfuscation approach was proposed. This in turn utilizes the behavioral characteristics from the dynamic analysis of executed application in the emulated framework. The approach proposed was estimated over 13533 applications from the categories like utilities, gaming and banking. This method was effectual offering 98.84% F-measure and 98.08% detection rate.

A deep learning-based system was presented in [11] for the detection of malicious Android applications over a dynamic analysis with the use of stateful generation of input. The performed experiments were done in 30000 benign and malware applications on the real-time devices. Moreover, experimentations were similarly conducted for comparing the performance of detection rate and the code coverage method of stateful input generation by usually employed stateless method employing DL system. This approach exposes that the DL-Droid could attain detection rate of 97.8% (dynamic features) and detection rate of 99.6% (dynamic +static features) correspondingly that outperforms the existing machine learning approaches.

In [12] MLDroid—web-based framework was proposed that aids in the detection of malware from the android devices. Because of growing popularity of android devices, malware developers thus develop malwares regularly for threating the user's privacy and system's integrity. The framework proposed thus identifies malware from Android apps by means of performing dynamic analysis. So as to detect real-world apps malwares, proposed method was trained on selecting features that were attained on implementing the approach of feature selection. Additionally, these features selected aid to figure a model on considering various algorithms of machine learning.

A novel method was suggested in [13] so as to detect malwares in the android application with the use of gated recurrent unit (GRU) which was regarded as a kind of recurrent neural network (RNN). Two static features were extracted termed API (Application Programming Interface Calls and Permissions from android application). The presented approach was trained and tested by the dataset termed CICAndMal2017. The outcomes reveals that the deep learning technique performs well than others by offering an accuracy of 98.2%.

The work presented in [14] employed an approach based on machine learning algorithm with the use of Support Vector Machines (SVM) for detecting Android applications that are malicious; a novel technique offers outcome that were competitively high on comparing traditional methods. The employed dataset was from the environment of Android at which the benign and malicious applications access frequently a system resources over Android API calls. Meanwhile an Android application might appeal a small number of APIs comparatively in the normal circumstances, data in the dataset was high dimensional and sparse fundamentally.

In the work[15],a static approach of base classification was presented for the detection of malware depending on the API calls and android permissions. It was based on 3 renowned machine leaning algorithms like support vector machines (SVM), Naïve Bayes (NB), and KNN in contrast to the wide-ranging novel dataset of Android malware (CIC InvesAndMal2019), after accomplishing higher malware detection rates, effort of contribution & studies in defending the mobile information development.

A new technology was introduced in [16] for the Ransomware detection which was based on evolutionary-dependent ML method. An algorithm of particle swarm optimization was used for tuning hyperparameters classification approach, along with performing feature selection approach. The SVM approach was utilized along with the SMOTE (synthetic minority oversampling technique) for the purpose of classification. The proposed technique SMOTE-tBPSO-SVM performance attains merits over existing ML approaches by offering maximum scores for the metrics g-mean, sensitivity, and specificity.

The work [17] suggested a new framework at which the hybrid feature selection approach with the use of wrapping feature selection (WFS) along with the integration of greedy stepwise and random forest (RF-GreedySW) was devised for the optimization of malware features. The framework proposed was competent of decreasing huge attributes number to optimal range of features for enhancing ML model performance. In this, most popular ML models like RF, decision tree (C5.0) and SVM radial basis function (SVM RBF).

An automated framework termed TC-Droid was suggested in [18] for the detection of android malwares depending on the method of classification. A TC-Droid core idea was derived from text field classification. The technique TC-Droid feeds on the APPs text sequence on analyzing information produced by Andro PyTool, smears a CNN (convolutional neural network) for exploring important knowledge (or information) in unique text report, in spite of manual feature engineering.

A machine learning based approach was proposed in [19] for the detection of Android malware depending on the feature applications. Furthermore, the extracted proposed approach with aper mission features and new static API Calls dataset using huge benign and malicious dataset samples of Android APK.

A deep learning-based approach was suggested in [20] for the detection of malware. The method of deep learning employed for the detection of malware which covers auto encoders, CNN, LSTM and RNN. LSTM was identified to have cells memory for having better probabilities. The auto encoders were found to offer better unsupervised approach having decoding and encoding scheme for anomalies (malware) detection. There were several contributions identified with the use of ML and DL approach for the detection of android malwares.

III. PROPOSED WORK

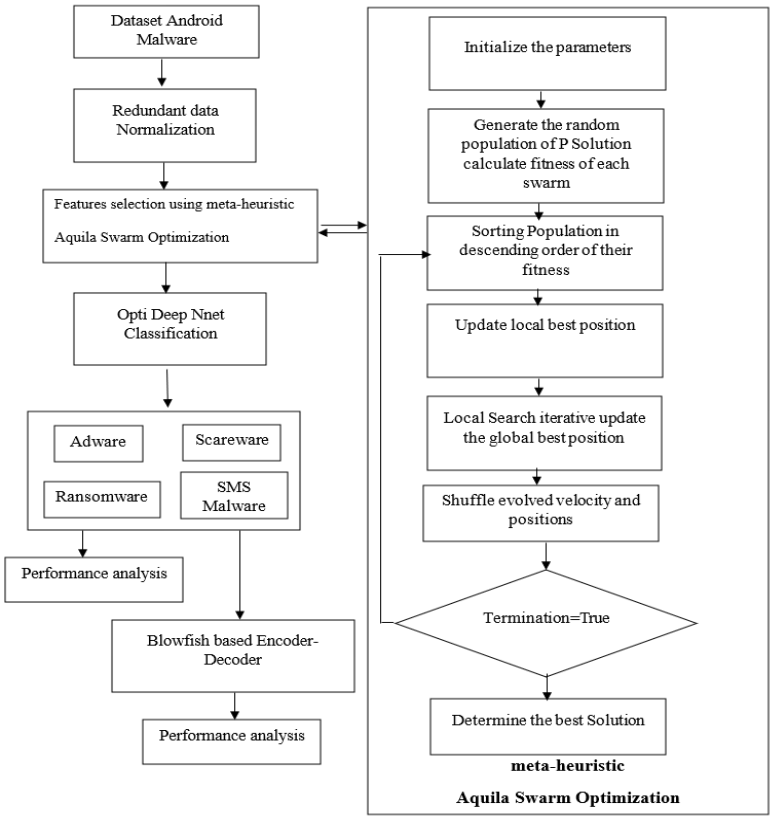

Fig. 1 Flow depiction of proposed system

A new framework for the detection of Android Malware is proposed in this section using deep learning approach. The input android malware dataset is retrieved and the redundant data is normalized to remove redundant information. The features are selected with the use of meta-heuristic Aquila swarm optimization

algorithm. From the selected features, optimization based Deep Neural Network (Opti Deep Nnet) classifier is employed for classifying and detecting android malwares as benign or malwares. The detected malwares are classified as Adware, Scareware, Ransomware, and SMS malware. The attained result is stored in mobile platform using Blowfish based Encoder-Decoder algorithm. At last, the performance is estimated.

A. Input data preprocessing

A data is preprocessed initially to remove the redundant data using normalization approach. Since the android apps contains high dimensional features, data preprocessing is needed for identifying the significant and most appropriate features. The intention of feature filtration approach that is applied before the classification process is to enhance the accuracy of malware detection thereby reducing the computational complexity so as to train the deep learning method. In this work, the top-ranked features are attained for the output of CV Attribute, Chi Square, information Gain Attribute evaluation, Gain Ratio Attribute evaluation, Relief attribute, Symmetrical Un cert Attribute Eval, and One R Attribute Eval. The experiment of feature filtration is carried finally from the input dataset considered. After filtering approach, a Meta-heuristic Aquila Swarm Optimization algorithm is carried for the selection of features and is described briefly in subsequent sections.

B. Meta-heuristic Aquila Swarm Optimization for feature selection

The features from the normalized data is extracted and the most significant ones are selected with the use of meta-heuristic Aquila Swarm optimization technique. From this, an optimal range of features are selected which enhances the accuracy of detection process. In the typical AO algorithm, the individual of swarm would carry four stages for capturing prey.

Stages 1: the expanded exploration

Initially, the Aquila may recognize the prey location thereby widely fly in search of prey and it is expressed as:

$$X\_i\ (t+1) = X\ \_best\ (t)\times(1-t/T)+X\_M\ (t)-X\_\ best\ (t)*rand \quad (1)$$

Here, X_ i (t+1) signifies ith individual position at t+1 iteration. X_ best (t) signifies better location at present iteration. The mean position of entire individual at current iteration is signified by X_M (t).

$$X\_M\ (t)=1/N\ \sum\_(i=1)\ \hat{}N\ 〚X\_\ i\ (t)\ 〛 \quad (2)$$

Here, X_ i (t) signifies the ith position individuals at t iteration and N is the swarms population size. rand denotes the Gaussian distribution random number among the interval 0 and 1. T is the maximum number of iterations allowed.

Stage 2: A narrowed exploration

It is still at the stage of exploration, once Aquila identifies a prey, it might fly above the prey on circling and thus prepares the land which is expressed by:

$$X\_i\ (t+1) =X\_\ best\ (t)\times Levy(D)+X\_R\ (t)+(y-x)*rand \quad (3)$$

Here, D signifies problem solving dimensionality. Levy(D) signifies Levy flights that might be computed as shown:

$$Levy(D)=s\times\ (\mu\ \times\sigma)/|v|\hat{}(1/\beta) \quad (4)$$

Here, s=0.01 signifies constant parameter, random numbers among 0 and 1 is signified by μ and v. And σ is signified by:

$$\sigma=(\Gamma(1+\beta)\times\sin(\pi\beta/2))/(\Gamma((1+\beta)/2)\times\beta\times2^{((\beta-1)/2)}) \quad (5)$$

Here, β signifies a constant value that is fixed to 1.5. Γ signifies gamma function in Mathematics.

The random candidate selected is represented by X_R (t) at present iteration. The spiral shape is signified by y and x and is computed by:

$$y=r\times\cos(\theta) \quad (6)$$

$$x=r\times\sin(\theta) \quad (7)$$

$$r=r\_1+U\times D\_(1) \quad (8)$$

$$\theta=-\omega\times D\_1+\theta\_1 \quad (9)$$

$$\theta\_1=3\pi/(2) \quad (10)$$

Here, r_1 denotes the fixed number among 1 and 20. The integer from 1 to that of the problem length is represented by D_(1). the constant fixed number is ω=0.005.

Stage 3: Expanded exploration

At the time of exploration process, Aquila may reinitialize itself once they might be unsuccessful to identify the target, after that they are responsible for updating positions by the following equation:

$$Xi(t+1)=\alpha\times[X\_best(t)-X\_M(t)]+\delta\times[(UB-LB)\times rand+LB] \quad (11)$$

In this, [UB-LB] are the given problems definitional domain. The two fixed small numbers are signified by α and δ.

Stage 4: Narrowed exploitation

Once the Aquila becomes closer to prey, they might take narrowed exploitation by the subsequent equation:

$$X\_i(t+1)=QF\times X\_best(t)-G\_1\times X\_i(t)\times rand-G\_2\times Levy(D)+rand\times G\_1 \quad (12)$$

Here, QF signifies quality function employed for search space equilibrium and is computed by means of:

$$QF=t^{((2\times rand-1)/〖(1-T)〗^2)} \quad (13)$$

$$G\_1=2\times rand-1 \quad (14)$$

$$G\_2=2\times(1-t/T) \quad (15)$$

Thus, from this an optimal feature is selected to improve the accuracy rate of classifier.

C. Optimization based deep neural network classifier

Once the optimal features are selected, the optimization based deep neural network classifier is employed for the purpose of classifying malwares.

The Opti Deep Nnet also termed as optimization based deep neural network is proposed. This proposed DNN is the feed-forward, artificial neural network (ANN) which is having more than one number of hidden units' layer among its output and its input. Typically, each of the hidden unit employs logistic function for

mapping their total input from a layer below x_j to scalar state y_j which they send to above layer. It is expressed by:

$$y\_j=\text{logistic}\ (x\_j\ )=1/(1+e\text{^}(-x\_j\ )\ ),x\_j=b\_j+\sum\_i\_y\_i\ \ w\_ij \tag{16}$$

Here, b_j signifies the unit j bias and isignify the index over units at layer under &w_ij represents the connection weight to j unit from the unit i at below layers. In the binary classification, output j thus converts their total input x_j as a class probability p_j on employing nonlinearity softmax and is expressed by:

$$p\_j=(\exp(x\_j))/(\sum\_k\ 〖\exp(x\_(k\ ))〗\ ) \tag{17}$$

In this, k is class index. The proposed Opti Deep Nnet classifier is trained discriminatively by the derivatives of back propagation of the cost function which estimates the discrepancy among actual output and target output attained for every training instances. Once the softmax output function is employed, C denotes the cross entropy among the softmax output p and target probabilities d, and is expressed by:

$$C=-\sum\_j\ 〖d\_j\ \log p\_j\ 〗 \tag{18}$$

Once the target probabilities usually take the values of 1 to 0, are the supervised data given for training the Opti Deep Nnet classifier. Thus, the malwares are classified effectively as Adwares, ransomware, SMS malware, and scareware accordingly.

D. Blowfish based Encoder-Decoder

Blowfish based encoder-decoder algorithm is employed so as to store the classified outcomes in the android platform. This algorithm employs a Feistel network for encrypting data that iterates the function to 16 times. Every round covers key dependent substitutions and data dependent permutation. The data encryption process with various steps of this are given below:

Step 1: Split 64-bit block as two blocks of equal that are having size of 32 bits each (XR and XL). The XL left block is XOR'd having first element as P-block, and therefore result attained is thus fed to function F.

Step 2: In the function block F, the operation of substitution is thus carried at which the input given 32-bit input is thus transformed to other 32-bit output.

Step 3: the F block output is the XOR'd with the right XR half and the attained results are swapped once the round is completed successfully, hence formed right half becomes new left half or the vice versa. This procedure is continued till 16 rounds.

Step 4: The final right and left halves are not thus swapped however XOR'd having 16th and 18th P box elements. Hence, the obtained outcome is a cipher text that is non-understandable to the outside attackers.

Therefore, by this Blowfish based encoder-decoder algorithm, privacy of android system is enhanced thereby reducing the rate of malware occurrences.

IV. Performance Analysis

In this section, the performance analysis of the proposed model is carried and the comparative estimation of attained outcome with existing ones are given.

A. Dataset Description

A recently published android malware dataset named (CICInvesAndMal2019) is employed. It is the second part of dataset CICAndMal2017 where the malware and benign android app's are thus tested on real-time smart device. It comprises of various information's concerning the Android malware, intents as the permission of statistical features, dynamic features, API calls, entire generated log files. Also, this dataset covers logs, packages, logs processes, battery state and so on. The samples of malware are thus classified as 4 kinds like SMS Malware, Adware, Scareware, and Ransomware.

B. Performance and comparative estimation

The performance and comparative estimation of proposed and existing techniques are carried and the attained outcomes are projected here.

Figure 2 signifies the Malware category classification result on the static layer at which the proposed approach provides high performance rate for metrics like accuracy, precision, recall, and F-score. On comparing existing models, proposed system offers enhanced output.

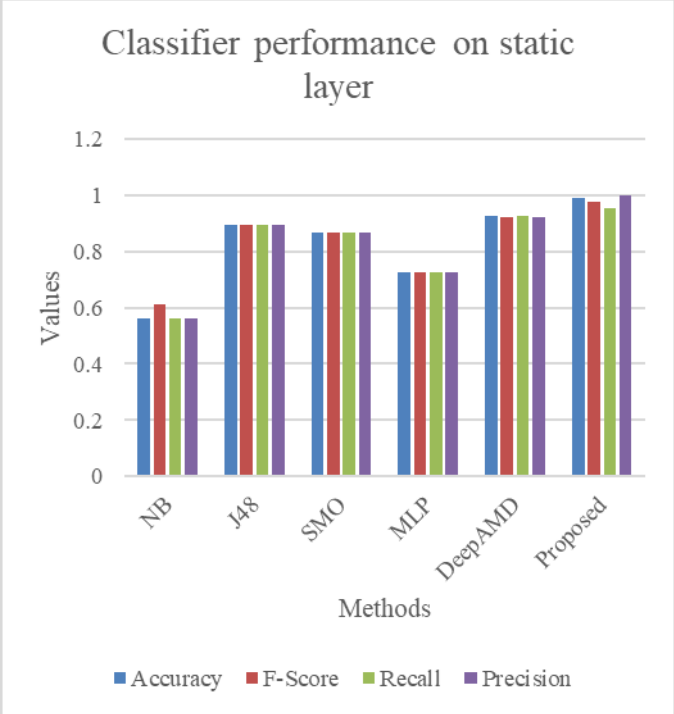

Fig.2 Estimation of performance metrics on static layer

Figure 3 signifies the Malware category classification result on the dynamic layer at which the proposed approach provides high performance rate for metrics like accuracy, precision, recall, and F-score. On comparing existing models, proposed system offers enhanced output.

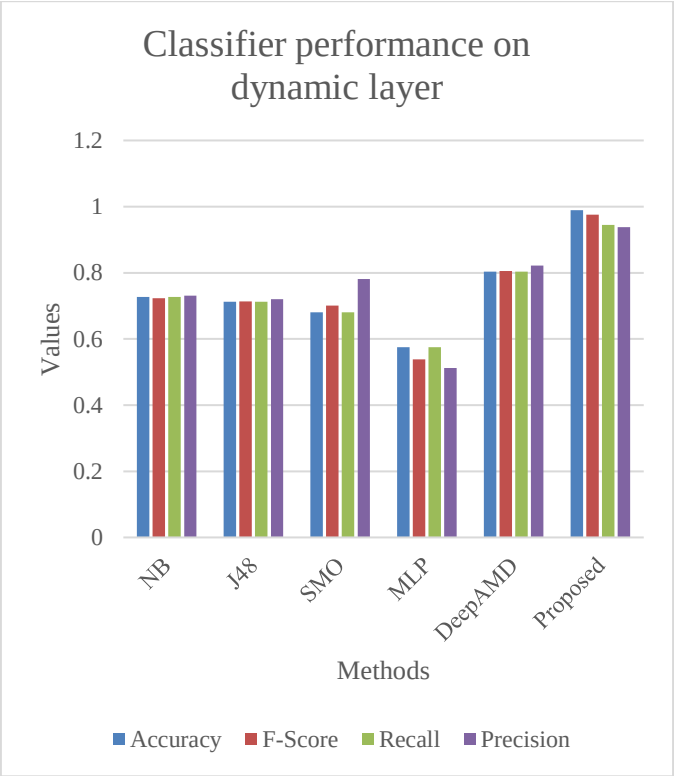

Fig.3 Estimation of performance metrics on dynamic layer

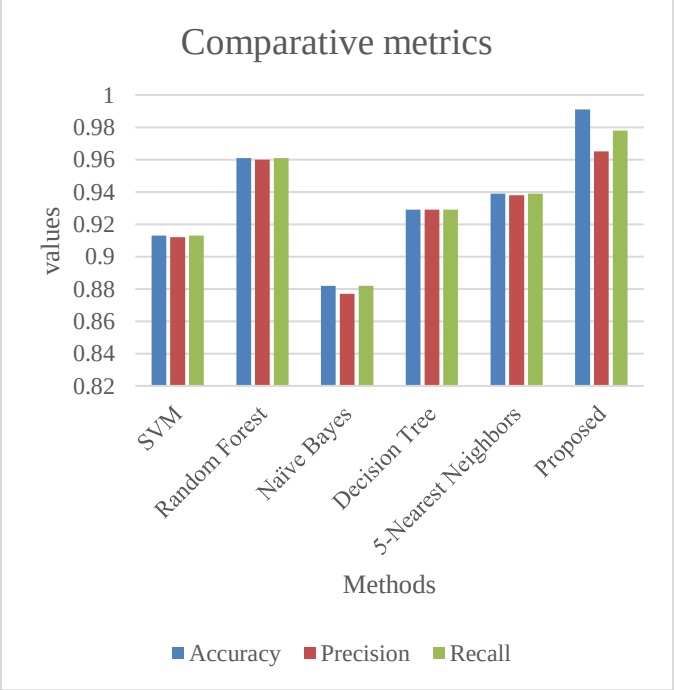

Figure 4 Comparative Estimation of performance metrics accuracy, precision, recall

Figure 4 signifies the comparative analysis of several techniques such as RF, Naïve Bayes, DT, 5-nearset neighbor with proposed model for the metrics accuracy, precision and recall. The outcome reveals that the proposed approach offers improved result than existing models.

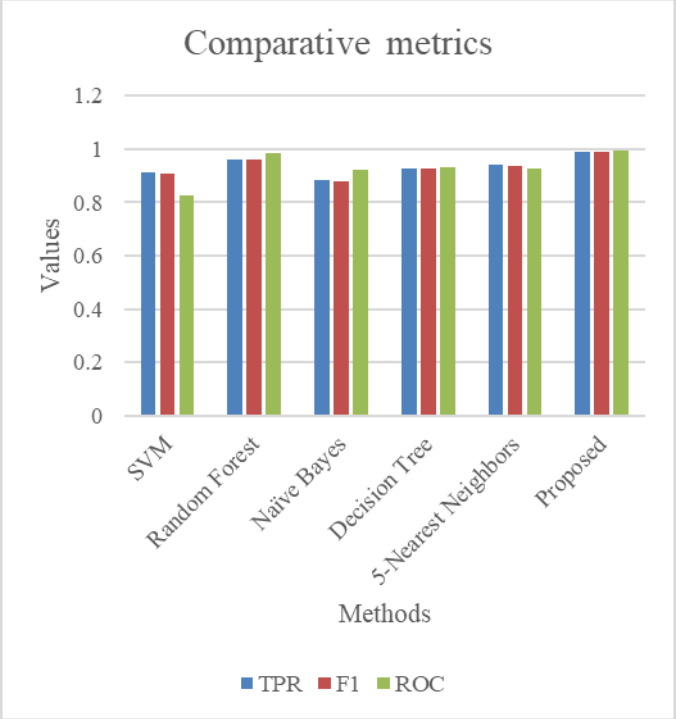

Figure 5 Comparative Estimation of performance metrics TPR, F1, ROC

Figure 5 represents the comparative analysis of several techniques such as RF, Naïve Bayes, DT, 5-nearset neighbor with proposed model for the metrics TPR, F1-score, and ROC curve. The outcome reveals that the proposed approach offers improved result than existing models.

Figure 6 denotes the accuracy convergence representation of suggested scheme regarding epoch. The suggested method system offersimproved test accuracy. The training accuracy starts at 0.75% and it goes till 0.985%. Then, training accuracy convergence keeps on increasing. The testing accuracy slows down at some point and rises steadily.

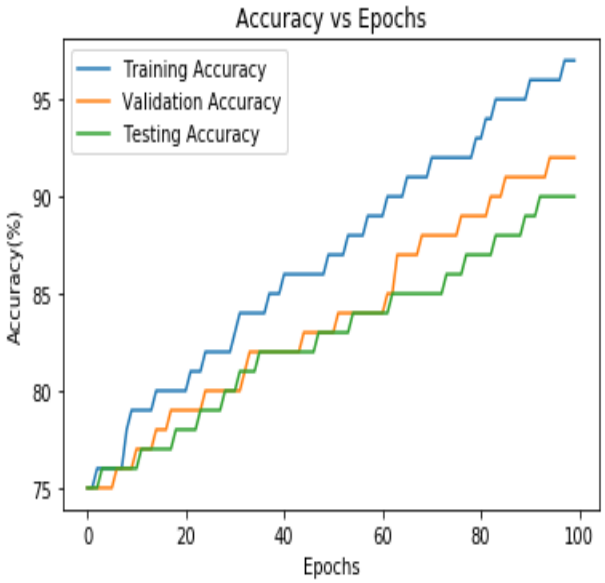

Figure 6 accuracy vs Epochs Analysis

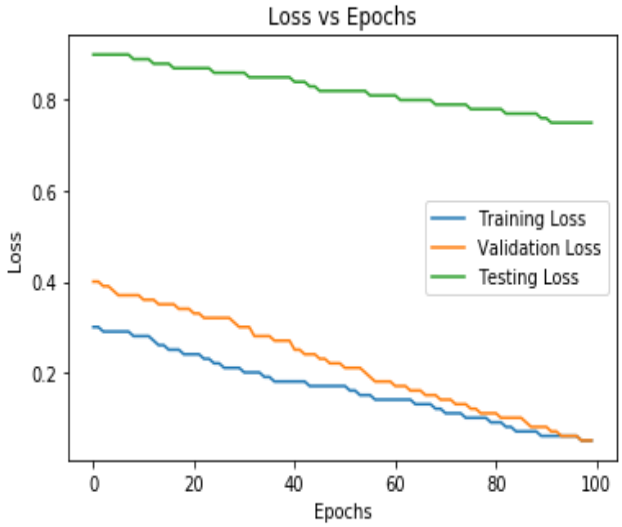

Figure 7 Loss vs Epochs analysis

Figure 7denotes loss convergence analysis of proposed system regarding epochs. The projected scheme achieve slowest loss. The training loss starts from 0.3% and goes down gradually. Then, the convergence of loss will be stable. The convergence of test loss starts above 0.85% and slows down till 0.79%.

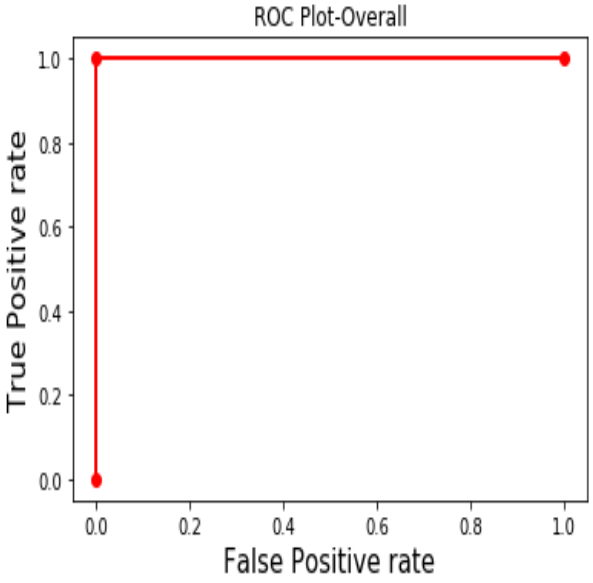

Figure 8 overall ROC plot analysis

Figure 8 denotes the overall analysis representation made on ROC plot for true positive rate and false positive rate. The output layer for classes classification such as ‘‘ransomware’’, ‘‘scareware’’, ‘‘SMS malware’’ and ‘‘adware’’ and family’s classification respectively.

V. CONCLUSIONS

A new scheme was proposed in this manuscript to detect Android malware with the use of deep learning-based classification approach. To begin with, am input android malware dataset was taken and preprocessing was carried to remove redundant data. After which, features were extracted from preprocessed data followed by optimal selection of features by employing MH-ASO. An Opti Deep Nnet classifier was used for classifying the malwares. The proposed classifier was account able for detecting and classifying android malwares as four kinds of classes like Adware, Scareware, SMS Malware, and Ransomware. The blowfish-based encoder-decoder is employed for protecting data from attackers. By this way, the android device security/privacy was

maintained. To conclude, the performance analysis is trained, tested and verified over dataset CICInvesAndMal2019 and the results were compared with existing models for metrics like accuracy, F-score, precision, recall, ROC plot, and TPR (True positive rate). The investigation exposes that the proposed model is effective than traditional models.

VI. REFERENCES

[1] Kim, J., Ban, Y., Ko, E., Cho, H., & Yi, J. H. (2022). MAPAS: a practical deep learning-based android malware detection system. International Journal of Information Security, 21(4), 725-738.

[2] Hammood, L., Doğru, İ. A., &Kılıç, K. (2023). Machine Learning-Based Adaptive Genetic Algorithm for Android Malware Detection in Auto-Driving Vehicles. Applied Sciences, 13(9), 5403.

[3] Ravi, V., &Chaganti, R. (2022). EfficientNet deep learning meta-classifier approach for image-based android malware detection. Multimedia Tools and Applications, 1-27.

[4] Bayazit, E. C., Sahingoz, O. K., & Dogan, B. (2022, June). A Deep Learning Based Android Malware Detection System with Static Analysis. In 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-6). IEEE.

[5] Fallah, S., &Bidgoly, A. J. (2022). Android malware detection using network traffic based on sequential deep learning models. Software: Practice and Experience, 52(9), 1987-2004.

[6] Akhtar, M. S. (2023). Analyzing and comparing the effectiveness of various machine learning algorithms for Android malware detection. Advances in Mobile Learning Educational Research, 3(1), 570-578.

[7] Ravi, V., Alazab, M., Selvaganapathy, S., &Chaganti, R. (2022). A Multi-View attention-based deep learning framework for malware detection in smart healthcare systems. Computer Communications, 195, 73-81.

[8] Alzubi, O. A., Alzubi, J. A., Al-Zoubi, A. M., Hassonah, M. A., &Kose, U. (2022). An efficient malware detection approach with feature weighting based on Harris Hawks optimization. Cluster Computing, 1-19.

[9] Fatima, A., Maurya, R., Dutta, M. K., Burget, R., &Masek, J. (2019, July). Android malware detection using genetic algorithm based optimized feature selection and machine learning. In 2019 42nd International conference on telecommunications and signal processing (TSP) (pp. 220-223). IEEE.

[10] Sihag, V., Vardhan, M., Singh, P., Choudhary, G., & Son, S. (2021). De-LADY: Deep learning based Android malware detection using Dynamic features. J. Internet Serv. Inf. Secur., 11(2), 34-45.

[11] Alzaylaee, M. K., Yerima, S. Y., &Sezer, S. (2020). DL-Droid: Deep learning based android malware detection using real devices. Computers & Security, 89, 101663.

[12] Mahindru, A., &Sangal, A. L. (2021). MLDroid—framework for Android malware detection using machine learning techniques. Neural Computing and Applications, 33(10), 5183-5240.

[13] Elayan, O. N., & Mustafa, A. M. (2021). Android malware detection using deep learning. Procedia Computer Science, 184, 847-852.

[14] Han, H., Lim, S., Suh, K., Park, S., Cho, S. J., & Park, M. (2020, February). Enhanced android malware detection: An svm-based machine learning approach. In 2020 IEEE International Conference on Big Data and Smart Computing (BigComp) (pp. 75-81). IEEE.

[15] Shatnawi, A. S., Yassen, Q., &Yateem, A. (2022). An android malware detection approach based on static feature analysis using machine learning algorithms. Procedia Computer Science, 201, 653-658.

[16] Almomani, I., Qaddoura, R., Habib, M., Alsoghyer, S., Al Khayer, A., Aljarah, I., & Faris, H. (2021). Android ransomware detection based on a hybrid evolutionary approach in the context of highly imbalanced data. IEEE Access, 9, 57674-57691.

[17] Smmarwar, S. K., Gupta, G. P., & Kumar, S. (2022). A hybrid feature selection approach-based Android malware detection framework using machine learning techniques. In Cyber Security, Privacy and Networking: Proceedings of ICSPN 2021 (pp. 347-356). Singapore: Springer Nature Singapore.

[18] Zhang, N., Tan, Y. A., Yang, C., & Li, Y. (2021). Deep learning feature exploration for android malware detection. Applied Soft Computing, 102, 107069.

[19] AlJarrah, M. N., Yaseen, Q. M., & Mustafa, A. M. (2022). A Context-Aware Android Malware Detection Approach Using Machine Learning. Information, 13(12), 563.

[20] Majid, A. A. M., Alshaibi, A. J., Kostyuchenko, E., &Shelupanov, A. (2023). A review of artificial intelligence based malware detection using deep learning. Materials Today: Proceedings, 80, 2678-2683.