An Effectual Analytics and Approach for Avoidance of Malware in Android using Deep Neural Networks

*Kapil Aggarwal[1], Santosh Kumar Yadav[2]
[1]Banasthali University, Rajasthan, India
[1*]kapil594@gmail.com
[2]Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan, India
[2]drskyaadav@hotmail.com

Abstract.
Due to the rise in smartphone apps and Android use by people who use their phones a lot, there are a lot of security issues. Security issues need to be addressed in order to prevent vulnerabilities and find them before they happen. People who use smartphones are linked to a warning about the risks. Most people who use mobile phones don't have to think about a few bad things when they install APK files from different sources. It is important to make and use a system that can tell if code in Android apps is bad. Our first step is to look at the Android APK datasets. Both good and bad APKs are analyzed and dataset is processed. In this study, we'll look for and extract the signatures that are hidden in the APKs. This will make it easier to build a training dataset. As a whole, we're going to look at assorted dataset of APK files, and we think that about half of them will be safe and the other half will be dangerous. Then, we check to see if each APK has the permissions it needs and how it affects the way it works. Once it's been cleaned, a dataset will be made so that the model can be trained so that it can predict what will happen. To finish predictive analytics, any APK outside of the APKs is chosen and used. That's when it's possible to figure out how likely it is that the new APK being looked at will have bad code in it. Machine learning is used to track the results of different prediction measures, such as how long it takes, how accurate they are, and how much they cost. To compare two things, we use machine learning to combine our predictions. This article describes a machine learning technique to solve functional selection by safeguarding the selection and mutation operators of genetic algorithms. During population calculations in the training set, the proposed method is adaptable. Furthermore, for various population sizes, the proposed method gives the best possible probability of resolving function selection difficulties during training process. Furthermore, the proposed work is combined with a better classifier in order to detect the different malware categories. The proposed approach is compared and validated with current techniques by using different datasets.Using this approach, the accuracy is compared and found the elevated results in proposed approach.

Keywords: Android malware, apk analytics, android apk fingerprinting, smartphone security.
DOI Number: 10.14704/nq.2022.20.11.NQ66049 **NeuroQuantology 2022; 20(11): 470-478**

1 Introduction

A report from ThreatFabric recently found that more than 300,000 Android users didn't know they had downloaded malware with banking trojan abilities, and that it was able to get around Google Play Store restrictions.The cybercriminals came up with a way to infect Android users with different banking trojans, which are meant to get into people's account information [1].

It was the first step to submit apps to the Google Play Store that had almost no malicious footprint and looked like useful apps, like QR Code scanners, PDF scanners, cryptocurrency apps, or fitness apps. Afterward, these apps asked the user to update them, which was done outside of the Google Play Store and installed malicious content on their Android phones. There was a way to install malicious content after you had installed the app. This way, it was completely unnoticeable to Google Play when the app was installed, so it was not dangerous[2].

The attackers were careful enough to submit an early version of their apps to the Google Play store that didn't have any download or install features. Later, they updated the apps with more permissions, which made it easier for people to download and install the malware[3, 4]. They have also put limits on how the payload can be used. They used mechanisms to make sure the payload was only installed on real victims' devices and not in test environments, making it even more difficult to find.

Anatsa, Alien, Hydra, and Ermac are four different types of banking Trojans that have been found by ThreatFabric. Anatsa is the most common. Researchers at Zimperium have found a new virus that looks like a system update programme, which makes it hard to

eISSN 1303-5150

find. Installed, it takes control of Android phones and steals data, texts, and pictures, for example. In the words of the researchers, hackers can record audio and phone calls as well as take photos. They can also steal messages and files, as well as access instant messenger accounts. Hackers can also look into the user's browser, taking their search history and bookmarks. It is possible for them to see what the user is copying to the clipboard, as well as get information about the user's device.Android Package (APK) is a type of file format used by the Android operating system. It is used for both installation and distribution of mobile apps. Apk is an extension used by the Android Operating System (OS) to identify application files [5-8].

Table 1: Traditional structure of Android APK

Description	APK File / Folder
Optional Folder for AssetManager	assets/
Compiled Code of Application	classes.dex
Resources without compilation	res/
Application resources	resources.arsc
Information of Metadata	META-INF/
XML format Manifest File	AndroidManifest.xml
Optional Folder with Compiled Code	lib/

Android APK Permissions and Vulnerability Aspects are hereby underlined. Apps like QR code readers, scanners, fitness trackers, and cryptocurrency trading platforms aren't always real[9, 10]. That's what researchers at ThreatFabric found when they looked at apps like these. Hackers have been able to make harmful versions of these apps that look just like the real ones. So that people don't think anything, these apps would advertise what they do in the best way possible[11]. These ads convince people to download these apps, which makes them easy prey for hackers.

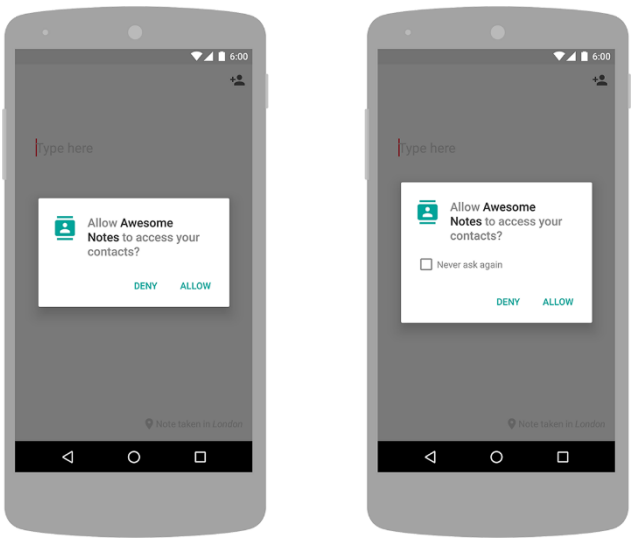

Fig. 1 Permissions in APK

eISSN 1303-5150

- REQUEST_DELETE_PACKAGES
- REQUEST_COMPANION_USE_DATA_IN_BACKGROUND
- GET_PACKAGE_SIZE
- TRANSMIT_IR
- SET_ALARM
- FOREGROUND_SERVICE
- KILL_BACKGROUND_PROCESSES
- CHANGE_WIFI_MULTICAST_STATE
- MODIFY_AUDIO_SETTINGS
- CHANGE_NETWORK_STATE
- READ_SYNC_SETTINGS
- SET_WALLPAPER_HINTS
- ACCESS_NETWORK_STATE
- EXPAND_STATUS_BAR
- VIBRATE
- DISABLE_KEYGUARD
- REQUEST_COMPANION_RUN_IN_BACKGND
- INSTALL_SHORTCUT
- USE_FINGERPRINT
- CHANGE_WIFI_STATE
- ACCESS_LOCATION_EXTRA_COMMANDS
- MANAGE_OWN_CALLS
- BLUETOOTH
- SET_WALLPAPER
- READ_SYNC_STATS
- BROADCAST_STICKY
- WRITE_SYNC_SETTINGS
- REORDER_TASKS
- WAKE_LOCK
- ACCESS_NOTIFICATION_POLICY
- REQUEST_IGNORE_BATTERY_OPTIMIZATIONS
- ACCESS_WIFI_STATE
- NFC
- INTERNET
- BLUETOOTH_ADMIN
- RECEIVE_BOOT_COMPLETED

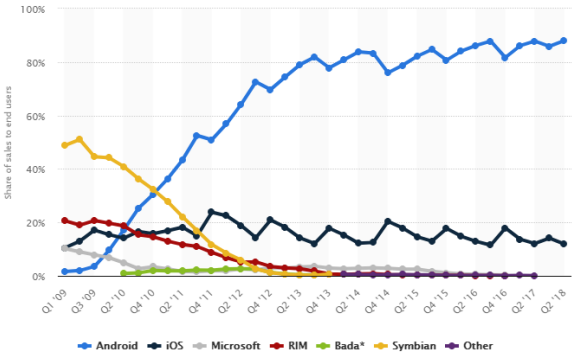

Fig. 2 Market Span of Smartphone O.S.

2 Malware Patterns and Machine Learning

Malware, which can come in many different forms, can be made to do bad things to a computer system and achieve a number of bad goals, like destroying the computer, getting money, or getting into the computer in an illegal way, which could lead to less security or the leak of system information. Some harmful packets can be sent in many different ways, including Trojans, Rootkits, Beasts, Suspicious packers, Scareware, Evasion, Backdoors, Keyloggers, Trojan Spy and Trojan GameThief. There are many more. It can spread in two ways. Polymorphic malware changes its code each time it copies itself, but

keeps its original coding and makes it look like it's always new. A lot of different IDS tools are out there, and some of them are free. They can both classify attacks (using PCAP Files) and monitor network traffic [12-17].

Deep Learning has a very low rate of mistakes, compared to machine learning, which has a lot of mistakes. Deep Learning, also known as Deep Structured Learning and Hierarchical Learning, is when people learn by looking at a lot of different features or representations in a way that makes sense to them. This is called deep learning[18-26]whereby the advanced algorithms of neural networks and dynamic analytics is done.

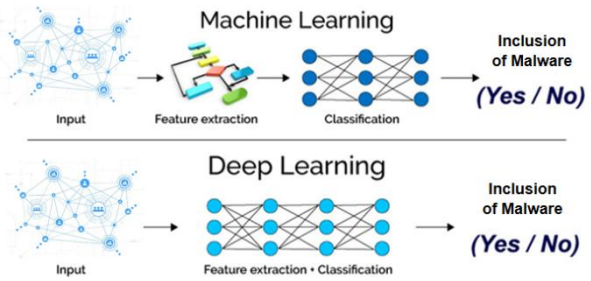

Fig. 3 Machine Learning and Deep Learning

3 Research Statement and Goal

While there are more and more problems in the Android ecosystem, because there are so many Android phones around the world, people get their apps from a lot of different places. If you have a way to show the model and technique, it's easier to find APK vulnerabilities [27-30].

Following are the major Suspicious Apps released on different online repositories

- com.family.cleaner – Cleaner: Safe and Fast
- com.op.blinkingcamera – Blinking Camera
- com.use.clever.camera – Clever Camera
- com.touch.smile.camera – Smile Camera
- com.just.parrot.album – com.qti.atfwd.core

eISSN 1303-5150

- com.color.rainbow.camera – Rainbow Camera
- com.flappy.game.cat – FlappyCat
- com.bunny.h5game.parkour – Easter Rush
- com.op.blinking.camera – Blinking Camera
- cm.com.hipornv2 – HiPorn

4 Experimental Results and Outcome

The first thing that is done is to get APK files from Online Repositories so that the permission-based dataset can be made. It is also taught how to use machine learning and deep learning.

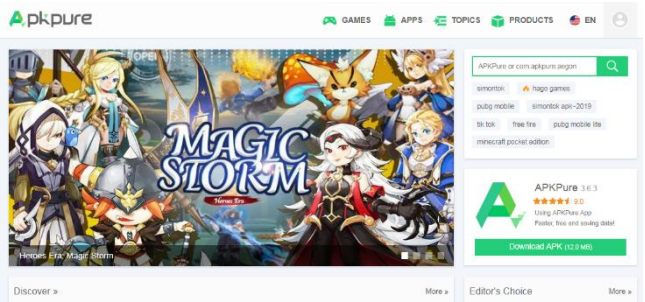

Fig. 4APK Repository of APKPure

The APKPure repository has a lot of Android apps that are analysed with the APK Tool so that the analytics on their permissions and suspicious parameters can be looked at.

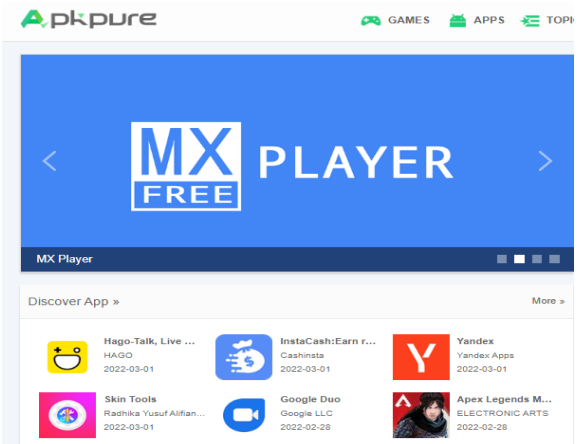

Fig. 5APKPure Apps Repository

Following is the generation panel of dataset from Android Apps using Android AAPT. The toolkit of Android AAPT provides the instructions to fetch the permissions associated with Android App without installation in the smartphone. It is done so that detailed fingerprinting can be done.

Table 2: Formation of Dataset for Implementations

Call	Suspicious (1-Malignant, 2-Benign)	Bluetooth	SMS	Contact
0	**1**	0	1	0
1	**2**	1	1	1
0	**1**	0	0	1

eISSN 1303-5150

1	**2**	1	1	1
1	**1**	1	1	1
0	**1**	0	1	1
1	**1**	0	0	0
1	**1**	1	0	0
1	**1**	1	1	0
1	**1**	1	1	1
0	**2**	1	0	1
1	**1**	1	1	1
1	**1**	1	0	1
1	**1**	0	1	1
1	**1**	1	0	1
1	**1**	0	0	1
1	**1**	1	0	0
0	**1**	1	0	1
0	**1**	1	1	1
0	**1**	1	1	1
1	**1**	0	0	1
1	**2**	0	1	1
1	**1**	0	1	0
1	**1**	1	1	0

Table 3: Evaluation of Parameters

Traditional Machine Learning Approach (Accuracy)	**Deep Learning Based Approach (Accuracy)**
84	93
82	93
75	95
81	94
86	97

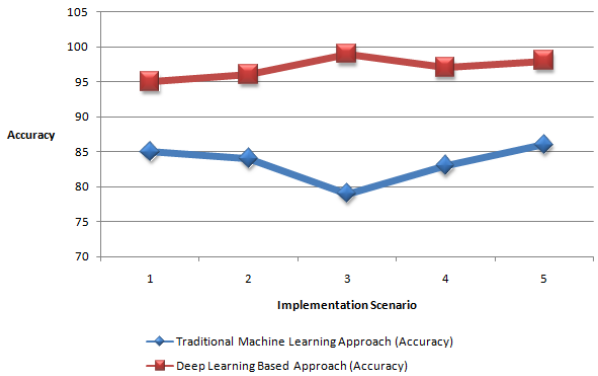

Fig. 6 Evaluation of Execution Time

eISSN 1303-5150

Figure 6 shows how well deep learning and traditional machine learning do when it comes to how long it takes to do. The execution time of deep neural network based learning is found to be more accurate in performance than the classical approach of traditional machine learning in all assorted attempts. The implementation of deep learning based algorithm raises the performance of the overall system in terms of accuracy as compared in using traditional approach.

Conclusion

This work is about how to use deep learning to make better malware detection and predictions with a higher level of accuracy. Specialist methods were used to find out that by classifying the Android APK, it ispossibleto predict with ahigh level of certainty by looking at the footprints and signs left by the APK. The methods given help get results that take into account a lot of different things while also making sure that each process knows how to work efficiently. It's possible to use Blockchain Technology in apps like Android APKs to make them more secure or private. Cryptocurrency has become one of the most important fields of study because of all the work being done on Blockchains. Many digital coins are popular and have gained a lot of attention around the world, even though there have been some complaints and scandals. There are many different types of "cryptocurrency," like BitCoin and Ethereum. These include BitCoin, LiteCoin, GridCoin, PrimeCoin, Ripple, Nxt, Dogecoin, NameCoin, AuroraCoin and many more. These cryptocurrencies work on blockchains, which means they don't record transaction details through a bank or payment service. That is the main reason that many countries have banned cryptocurrency. Despite the fact that these well-known and used digital currencies have very good security features, they still use the blockchain. Dynamic cryptography is used to manage a blockchain's block of records. This means that all transactions can be encrypted, and the blockchain is resistant to hackers and sniffers.We need to understand the drawbacks of our proposed deep learning algorithm. The method described in this article is designed to solve the specific problem of optimizing functionality selection. The redundancy of the Crossover operator is specific for this optimization problem category. Secondly, the selection problem for a data set is optimized, and the number of samples and the dimension of data is the main factors influencing runtime or the method called runtime. For future study, it is thus an essential job to use the method for large cases without adding computing complexity.

References

1. Vivekanandam, B. “Design an Adaptive Hybrid Approach for Genetic Algorithm to Detect Effective Malware Detection in Android Division.” Journal of Ubiquitous Computing and Communication Technologies 3, no. 2 (2021): 135-149
2. Jose, Rinu Rani, and A. Salim. “Integrated static analysis for malware variants detection.” In International Conference on Inventive Computation Technologies, pp. 622-629. Springer, Cham, 2019
3. Kumar, Ashwin A., G. P. Anoosh, M. S. Abhishek, and C. Shraddha. “An Effective Machine Learning-Based File Malware Detection—A Survey.” In International Conference on Communication, Computing and Electronics Systems, pp. 355-360. Springer, Singapore, 2020
4. Deshotels, L., Notani, V., &Lakhotia, A. (2014, January). Droidlegacy: Automated familial classification of android malware. In Proceedings of ACM SIGPLAN on program protection and reverse engineering workshop 2014 (p. 3). ACM
5. Yerima, S. Y., Sezer, S., & McWilliams, G. (2014). Analysis of Bayesian classification-based approaches for Android malware detection. IET Information Security, 8(1), 25-36
6. Talha, K. A., Alper, D. I., & Aydin, C. (2015). APK Auditor: Permission-based Android malware detection system. Digital Investigation, 13, 1-14
7. Faruki, P., Ganmoor, V., Laxmi, V., Gaur, M. S., &Bharmal, A. (2013, November). AndroSimilar: robust statistical feature signature for Android malware detection. In Proceedings of the 6th International Conference on Security of Information and Networks (pp. 152-159). ACM
8. Yerima, S. Y., Sezer, S., &Muttik, I. (2015). High accuracy android malware detection using ensemble learning. IET Information Security, 9(6), 313-320
9. Cai, H., Meng, N., Ryder, B., & Yao, D. (2019). Droidcat: Effective android malware detection and categorization via app-level profiling. IEEE Transactions on Information Forensics and Security, 14(6), 1455-1470
10. Adebayo, O. S., & Aziz, N. A. (2019). The trend of mobilemalwares and effective detection techniques. In Multigenerational Online Behavior and Media Use: Concepts, Methodologies, Tools, and Applications (pp. 668-682). IGI Global
11. Onwuzurike, L., Mariconti, E., Andriotis, P., Cristofaro, E. D., Ross, G., &Stringhini, G. (2019). MaMaDroid: Detecting android malware by building markov chains of behavioral models (extended version). ACM Transactions

eISSN 1303-5150

on Privacy and Security (TOPS), 22(2), 14

12. Kim, T., Kang, B., Rho, M., Sezer, S., &Im, E. G. (2019). A Multimodal Deep Learning Method for Android Malware Detection Using Various Features. IEEE Transactions on Information Forensics and Security, 14(3), 773-788
13. Chen, S., Xue, M., Fan, L., Ma, L., Liu, Y., & Xu, L. (2019, February). How Can We Craft Large-Scale Android Malware? An Automated Poisoning Attack. In 2019 IEEE 1st International Workshop on Artificial Intelligence for Mobile (AI4Mobile) (pp. 21-24). IEEE
14. Sharma, A., & Sahay, S. K. (2019). Group-wise classification approach to improve Android malicious apps detection accuracy. arXiv preprint arXiv:1904.02122
15. Su, D., Liu, J., Wang, X., & Wang, W. (2019). Detecting Android Locker-Ransomware on Chinese Social Networks. IEEE Access, 7, 20381-20393
16. Saracino, A., Sgandurra, D., Dini, G., &Martinelli, F. (2018). Madam: Effective and efficient behavior-based android malware detection and prevention. IEEE Transactions on Dependable and Secure Computing, 15(1), 83-97
17. Li, J., Sun, L., Yan, Q., Li, Z., Srisa-an, W., & Ye, H. (2018). Significant permission identification for machine-learning-based android malware detection. IEEE Transactions on Industrial Informatics, 14(7), 3216-3225
18. Shen, F., Del Vecchio, J., Mohaisen, A., Ko, S. Y., &Ziarek, L. (2018). Android malware detection using complex-flows. IEEE Transactions on Mobile Computing
19. Karbab, E. B., Debbabi, M., Derhab, A., &Mouheb, D. (2018). MalDozer: Automatic framework for android malware detection using deep learning. Digital Investigation, 24, S48-S59
20. Zhu, H. J., You, Z. H., Zhu, Z. X., Shi, W. L., Chen, X., & Cheng, L. (2018). DroidDet: effective and robust detection of android malware using static analysis along with rotation forest model. Neurocomputing, 272, 638-646
21. Garcia, J., Hammad, M., &Malek, S. (2018). Lightweight, obfuscation-resilient detection and family identification of Android malware. ACM Transactions on Software Engineering and Methodology (TOSEM), 26(3), 11
22. Vinayakumar, R., Soman, K. P., Poornachandran, P., &Sachin Kumar, S. (2018). Detecting Android malware using long short-term memory (LSTM). Journal of Intelligent &Fuzzy Systems, 34(3), 1277-1288
23. Narayanan, A., Chandramohan, M., Chen, L., & Liu, Y. (2018). A multi-view context-aware approach to Android malware detection and malicious code localization. Empirical Software Engineering, 1-53
24. Chen, S., Fan, L., Chen, C., Su, T., Li, W., Liu, Y., & Xu, L. (2019). StoryDroid: Automated generation of storyboard for Android apps. arXiv preprint arXiv:1902.00476
25. Wang, W., Li, Y., Wang, X., Liu, J., & Zhang, X. (2018). Detecting Android malicious apps and categorizing benign apps with ensemble of classifiers. Future Generation Computer Systems, 78, 987-994
26. Hsien-De Huang, T., & Kao, H. Y. (2018, December). R2-d2: Color-inspired convolutional neural network (cnn)-based android malware detections. In 2018 IEEE International Conference on Big Data (Big Data) (pp. 2633-2642). IEEE
27. McLaughlin, N., Martinez del Rincon, J., Kang, B., Yerima, S., Miller, P., Sezer, S., ... &JoonAhn, G. (2017, March). Deep android malware detection. In Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy (pp. 301-308). ACM
28. Demontis, A., Melis, M., Biggio, B., Maiorca, D., Arp, D., Rieck, K., ... &Roli, F. (2017). Yes, machine learning can be more secure! a case study on android malware detection. IEEE Transactions on Dependable and Secure Computing
29. Milosevic, N., Dehghantanha, A., & Choo, K. K. R. (2017). Machine learning aided Android malware classification. Computers & Electrical Engineering, 61, 266-274
30. Feizollah, A., Anuar, N. B., Salleh, R., Suarez-Tangil, G., &Furnell, S. (2017). Androdialysis: Analysis of android intent effectiveness in malware detection. computers & security, 65, 121-134
31. Hou, S., Ye, Y., Song, Y., &Abdulhayoglu, M. (2017, August). Hindroid: An intelligent android malware detection system based on structured heterogeneous information network. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1507-1515). ACM
32. Idrees, F., Rajarajan, M., Conti, M., Chen, T. M., &Rahulamathavan, Y. (2017). PIndroid: A novel Android malware detection system using ensemble learning methods. Computers & Security, 68, 36-46
33. Tam, K., Feizollah, A., Anuar, N. B., Salleh, R., &Cavallaro, L. (2017). The evolution of android malware and android analysis techniques. ACM Computing Surveys (CSUR), 49(4), 76

Kapil Aggarwal is working as Assistant Professor in Department of Computer Science & Engineering, BanasthaliUniversity, Rajasthan, India. He has done Doctor of Philosophy in Computer Science and Engineering from Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan, India, Master of Technology from IETE, New Delhi, India and Bachelors of Engineering (Electronics) from Nagpur University, India. He is in academics for more than fourteen years. He has taught many Computing subjects like Artificial Intelligence, Machine Learning, Mobile Computing,Compiler Design, Software Engineering in Masters Levelcourses. His research area is in Network Security, Cloud Computing Internet of Things (IoT) and Neural Networks. He is a memberof Computer Society of India, IEEE, USA and Associate Member of Institution of Electronics and TelecommunicationEngineers, New Delhi, India.

Santosh Kumar Yadav is presently working as Pro Vice-President in Shri Jagdishprasad Jhabarmal Tibrewala University,Jhunjhunu, Rajasthan, India. He is also Honorary Secretary of ComputerSociety ofIndi, India. His area ofResearchin Network Security, Internet of Things, Cloud Technologies, VLSI technologies. Currently, guidingnumber of researchscholars in their Ph. D. research work. He is having morethan 20 years in academics and research. He is executivemember of editorial board of many reputed refereedcomputing international journals. He isexecutive memberof AIMA, IETE etc. He has guided more than hundred research scholars.

eISSN 1303-5150