International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-1S3, December 2019

An Efficient Malware Detection System using Hybrid Feature Selection Methods

S. Abijah Roseline, S. Geetha

Abstract: ***Malware is a serious threat to individuals and users. The security researchers present various solutions, striving to achieve efficient malware detection. Malware attackers devise detection avoidance techniques to escape from detection systems. The key challenge is that growth of malware increases every hour, leading to large damages to users' privacy. The training process takes much longer time, mining the unnecessary features. Feature Selection is effective in achieving unique feature set in detecting malware. In this paper, we propose a malware detection system using hybrid feature selection approach to detect malware efficiently with a reduced feature set. Machine learning based classification is performed on eight classifiers with two malware datasets. The experiments were done without and with feature selection. The empirical results show that the classification using selected feature set and XGB classifier identifies malware efficiently with an accuracy of 98.9% and 99.26% for the two datasets.***

Keywords: ***Malware, Malware Features, Malware Detection, Malware Feature Selection, PE Files.***

I. INTRODUCTION

Malware detection and classification is one of the crucial challenges in the security domain. Malware attackers widely use Windows operating system programs to perform their malicious actions. The file format used in Windows is Portable Executable (PE) format. So, the proposed approach uses extracted static features from PE files to classify malware.

The traditional feature extraction methods extract mostly all features and require more processing time to identify malware. The extracted features consist of unnecessary features that should be eliminated in order to achieve an efficient malware detection and classification system in terms of space and time. The low performance of the malware detection system is due to a large number of features that does not contribute to the generalization and identification of malware samples. Also, reducing the number of features is crucial to malware detection, but it needs to be done to maintain high degree of accuracy for unknown malware.

With the flow of huge amounts of malware, some machine learning classifiers easily overfits to high dimensional data. Dimensionality reduction is a good solution to tackle this challenge to achieve highest predictive accuracy. Feature selection is an essential step for better analysis and real-time malware detection in terms of time, training effort and detection accuracy.

Manuscript published on 30 December 2019.
* Correspondence Author (s)
S. Abijah Roseline*, School of Computing Science and Engineering, Vellore Insitute of Technology, Chennai, India. Email: abijahroseline.s2017@vitstudent.ac.in

S. Geetha, School of Computing Science and Engineering, Vellore Insitute of Technology, Chennai, India. Email: geetha.s@vit.ac.in

The contributions of this paper are,

- We build a unique feature set from the extracted PE feature set using a novel and hybrid feature selection approach.
- We perform classification on machine learning classifiers using the new feature set.
- The efficiency of the proposed approach is demonstrated with two malware datasets using the standard classification metrics.
- The results are compared for the malware classifiers with feature selection and without feature selection.

A. Portable Executable (PE) File Features

The .exe, .dll files, object code in Windows uses PE file format [16] which follows COFF (Common Object File Format) specification. It consists of MS-DOS MZ header, a stub program, the PE file signature, the PE file header, the PE optional header, section headers such as .text header, .bss header, .rdata header, .debug header, etc., and section bodies.

The first 64 bytes of the PE file is occupied by the MS-DOS MZ header. The validity of the PE file is checked in this section. The two main fields in this section are e_magic and e_ifanew. The e-magic is the first field which checks for compatibility with the MS_DOS type. The e_ifanew contains the offset.

The PE File section consists main content of the file such as code, data, resources, and other exe files. Each section has a header and a body. The .text section is the actual code section. The .rdata contains the import and export information. It includes read-only information such as literals, constant strings, etc. in the code. The .data section contains the global data in the code. The .rsrc section includes resources such as images, icons, menu, etc. in the file.

The rest of the paper is organized as follows: Literature survey for classification methods without feature selection and with feature selection is presented in section II. Section III presents the proposed methodology which includes the hybrid approach used and the flow of the malware detection model. The experimental results are discussed in section IV, and conclusion wraps up the paper.

II. LITERATURE SURVEY

A. Classification Methods without feature selection

Much research on malware analysis and detection have been done based on static, dynamic and machine learning approaches.

The malware detection techniques in the literature focusses on feature extraction from malware samples and the extracted features are directly given to the classification models [3] [13]. This section presents a survey of various methods employed for identifying malware.

In static analysis, features such as byte sequences, string sequences, opcode sequences [15], function length frequency [9], functional call graph [17] and PE file features [6] [14] are extracted. Schultz et al. [7] proposed an approach to extract different static features from binaries and trained with different machine learning classifiers. Static analysis is inefficient to code obfuscations and does not allow automated analysis. Traditional signature-based malware detection methods involve analyzing malicious code, signature generation and storage in signature database. By the time it performs all these steps, it becomes risky performing malicious actions.

Dynamic analysis extracts behavioral features such as system calls [18], instruction sequences, network activities, etc. Imran et al. [19] presented a similarity-based malware classification system. API call sequences were extracted using Hidden Markov Models (HMMs) and similarity scores were estimated for classifying malware. Their system works well for fewer data and requires a high computational cost. Dynamic analysis is ineffective when malware has the ability to change its behavior while executing in the virtualized environments.

Hybrid methods [20] involve training classifiers based on features extracted from static, dynamic or machine learning methods [8]. Rieck et al. [21] extracted the API calls dynamically and trained those features on Support Vector Machine (SVM). Islam et al. [20] demonstrated that the hybrid method is effective than the individual static or dynamic methods. Nataraj et al. [4] employed novel technique based on image processing by visualizing binary files as gray-scale images that achieves faster malware classification.

B. Classification Methods with Feature Selection

There are few methods that employ feature selection methods presented in the literature, to detect and classify malware. Fang et al. [12] proposed an automatic feature selection framework based on reinforcement learning that selects distinguishing feature set for malware detection. Moskovitch et al. [1] presented an approach for malware classification using notions of information retrieval and text classification. They extracted n-grams features and computed Term Frequency (TF) and TF Inverse Document Frequency (TF-IDF) representations for each n-gram. They evaluated the machine learning algorithms using selected features based on measures such as Document Frequency (DF), Gain Ratio (GR) and Fisher Score (FS).

O'Kane et al. [2] selected six unique opcodes as potential indicators Support Vector Machine (SVM) classification tests. Then, they used a pre-filtering approach using eigen vectors which discards the unnecessary features and reduces the effort in training malware samples. Lin et al. used TF-IDF, Principal Component Analysis (PCA) and Kernel Principal Component Analysis (KPCA) methods for feature selection and based on the selected features, malware samples are classified using SVM.

Shafiq et al. [6] presented an automatic approach for PE file feature extraction and performed feature selection based on Redundant Feature Removal (RFR), Principal Component Analysis (PCA) and Haar Wavelet Transform (HWT)). They trained the those selected features using machine learning classifiers. Ye et al. [10] developed an integrated detection system based on Objective-Oriented Association (OOA) mining approach for PE files. The features were selected using Max-Relevance method.

III. PROPOSED METHODOLOGY

A. Forward Feature Selection (FFS)

Given a set of features, the Forward Feature Selection (FFS) [22] step starts with an empty list and the relevant features are appended to it. The selection process first selects one feature and evaluates performance metrics using cross validation. The feature that shows the best value is selected and added to the feature list. The next iteration is performed with two features, where one feature is selected from the previous iteration and the other feature is selected from the feature list and that is not already selected in the previous iteration. The metrics is evaluated for two features and the feature contributing to highest value is selected and appended to the feature list. The procedure repeats until all features are processed.

B. Backward Feature Elimination (BFE)

Backward Feature Elimination [22] initially chooses a threshold level (mostly 5%). Then, all features are trained using machine learning model and the feature with the highest P-value is identified. If the highest P-value obtained from training is greater than the threshold level, the feature is discarded from the initial feature list. Otherwise, the feature is selected. The updated feature set is then trained again and the process is repeated until the feature with highest P-value is lesser than the threshold limit.

C. Simulated Annealing (SA)

Simulated annealing (SA) [11] is a global search technique that performs perturbations on an initial candidate solution randomly, achieving a new solution. The performances of both the previous and new solutions are compared. The new solution is considered if it is greater than the previous solution. Otherwise, a probability value is computed using the differences among the two solutions and the current search state. This value allows the search to generate better solutions in the next iterations.

D. Feature Selection

The features of PE binaries are extracted and given to select essential features. The proposed model selects

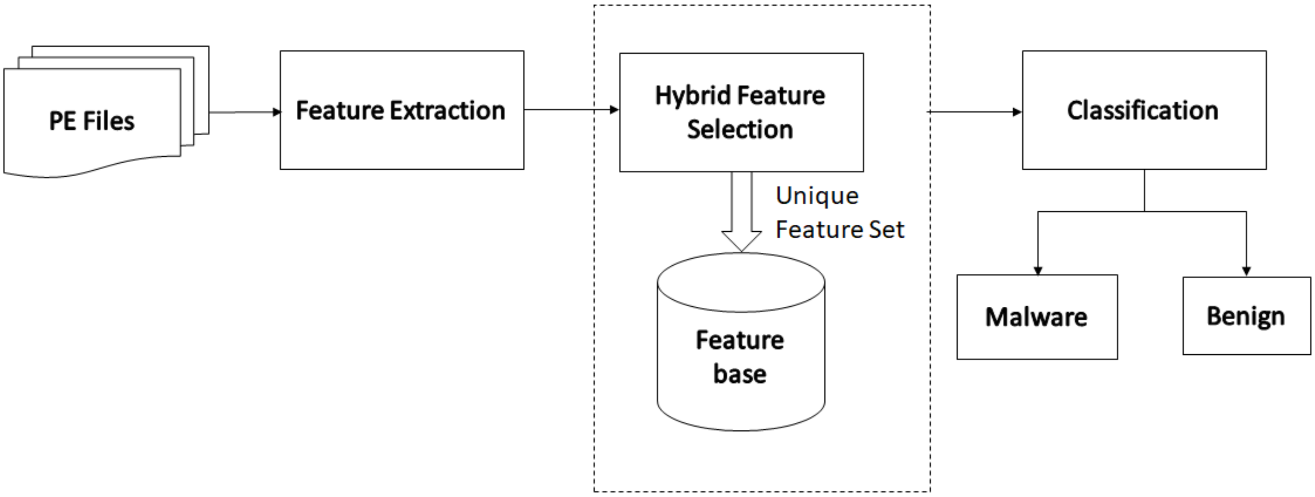

Fig. 1.Proposed System Model.

features based on a greedy algorithm called flow selection, which involves two phases. The first phase consists of forward, backward and simulated annealing selection process in a sequential manner. The non-trainable features are defined. Then, the initial features list are defined (empty list for forward selection and list with all trainable features for backward selection). First, the forward selection process generates the candidate features list, which is then given as input to the backward selection process. The evaluation metrics is calculated using the estimator to append or remove features in the candidate list. If there is no improvement in performance, the simulated annealing selection process is performed. In the next iteration, The selected feature is sent for forward selection. If there is improvement in performance, the selected feature from backward selection process is not sent to the simulated annealing process, but to the forward selection process in the next iteration.

In the second phase, cross-term is calculated for the selected features and three step validations (with selected features, with selected and all features, with entire set of features) are performed. First, the performance improvement is checked with selected features. If there is no improvement, validate with selected and entire list of features. Still, if there is no improvement the complete feature set is sent for evaluating cross-term in the next iteration. After the second phase, the best feature combinations are obtained if there is no further performance improvement. If there is improvement, the cross-term features are passed on to the first phase.

E. Proposed Malware Detection Model

The PE files are scanned to extract features that belong to malware. The feature extraction phase involves obtaining all features (relevant and irrelevant). The extracted features are allowed for feature selection using an approach that involves three selection methods such as FFS, BFE and SA. The features are selected sequentially using the methods based on their rules. The unique feature set obtained from feature selection phase is stored in a feature base, and relevant and useful features are retrieved by the classification phase. Various machine learning classifiers are trained with the unique features. The classification results show whether it is a malware or a benign file.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Datasets

The performance of the proposed malware detection model is evaluated with two PE malware datasets. The first dataset (referred as Dataset 1) is ClaMP (Classification of Malware with PE headers) [5] malware dataset with 2722 malware and 2488 benign samples. The features are

extracted from header field values such as DOS Header, File Header and Optional Header of PE files. The integrated dataset includes 68 features, of which 28 features are from raw dataset, 26 features are Boolean (obtained from File Header and Optional Header) and 14 derived features. The second dataset (referred as Dataset 2) consists of 56 features, 41324 malware, and 96724 benign samples.

B. Experimental Settings and Results

The experiments are performed for training machine learning classifiers in two settings, without feature selection (without FS) and with feature selection (with FS). The datasets are split with 70% training and 30% testing data. The machine learning algorithms used to detect malware are Naïve Bayes (NB), Logistic Regression (LR), K Nearest Neighbor (KNN), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), XGBoost (XGB), and MultiLayer Perceptron (MLP). In the first setting, the results are taken with the extracted features directly as input to the machine learning classifiers.

In the second setting, the extracted features are subjected to feature selection phase. Accuracy score and classifier is used as

Table- I: Performance Results for malware detection without and with feature selection

Models	**Dataset 1**								**Dataset 2**							
	Accuracy (%)		**Precision**		**Recall**		**F-score**		**Accuracy (%)**		**Precision**		**Recall**		**F-score**	
	Without FS	With FS	Without FS	With FS	Without FS	With FS	Without FS	With FS	Without FS	With FS	Without FS	With FS	Without FS	With FS	Without FS	With FS
NB	57	69.47	0.732	0.705	0.204	0.716	0.3370	0.9689	70.4	72.37	0.695	0.718	0.704	0.736	0.6995	0.7269
LR	79.5	82.32	0.756	0.821	0.911	0.834	0.8260	0.8315	84.39	85.42	0.838	0.853	0.846	0.854	0.8420	0.8535
KNN	90.7	92.25	0.913	0.914	0.914	0.926	0.9140	0.9260	93.26	92.18	0.926	0.916	0.933	0.932	0.9295	0.9239
DT	95.4	98	0.978	0.972	0.967	0.974	0.9720	0.9740	96.48	97.46	0.957	0.975	0.974	0.984	0.9654	0.9795
RF	97.1	98.63	0.992	0.979	0.988	0.982	0.9900	0.9765	98.04	98.88	0.98	0.977	0.981	0.995	0.9805	0.9859
SVM	60	78.68	0.981	0.755	0.255	0.79	0.4060	0.7678	74.64	86.34	0.748	0.863	0.759	0.863	0.7535	0.8630
XGB	97.47	**98.9**	0.989	**0.983**	0.99	**0.994**	0.9900	**0.9905**	98.35	**99.26**	0.968	**0.985**	0.994	**0.996**	0.9808	**0.9905**
MLP	88.9	90.56	0.878	0.873	0.923	0.915	0.8990	0.9016	91.2	92.72	0.906	0.918	0.912	0.934	0.9090	0.9259

validation method to append or remove features from the feature list. 5-fold cross validation is used to check the performance improvement to update the feature list. The selected features are fed into the machine learning classifiers.

Table-I shows the performance results of the machine learning models in both the settings for the two malware datasets. The results are evaluated for performance metrics such as accuracy, precision, recall and f1-score. From the table, we infer that the results for malware detection with FS shows comparably better results than without FS. The XGB classifier shows highest values for all performance metrics compared to other machine learning models. For dataset 1, XGB with FS shows 98.9% accuracy, 0.983 precision value, 0.994 recall value, 0.9905 f1-score value. For dataset 2, XGB with FS shows 99.26% accuracy, 0.985 precision value, 0.996 recall value, 0.9905 f1-score value.

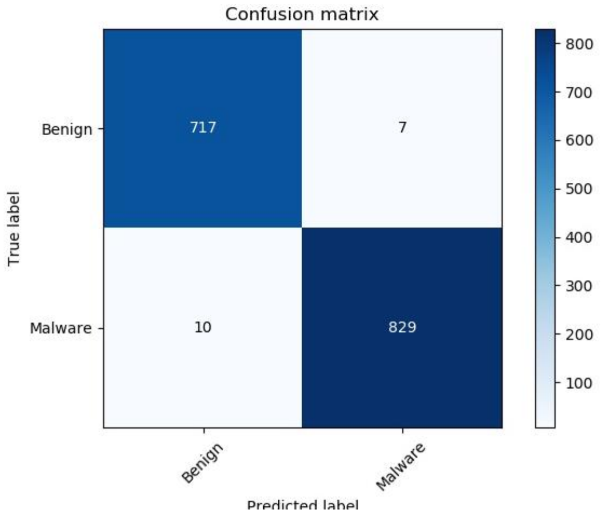

Fig. 2.Confusion Matrix for XGB classifier for dataset 1.

Fig. 2 and Fig. 3 shows the confusion matrix for the outperforming XGB classifier with selected features on both datasets 1 and 2 respectively.

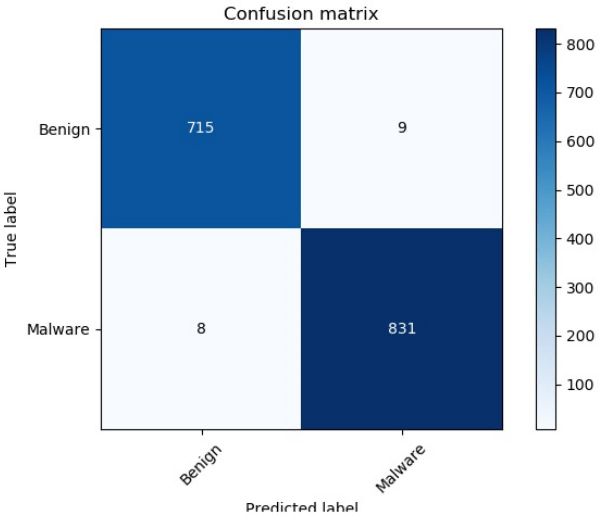

Fig. 3.Confusion Matrix for XGB classifier for dataset 2.

V. CONCLUSION

The detection of PE malware files from benign is essential, as most malware is written in PE file format. The significant discriminant features of malware are selected using a hybrid feature selection approach that improves the performance of the training classifiers. The new malware are variants of the previously existing malware. Thus, the trained features aid in detecting new malware. We achieve a generalized malware detection system. The training performance of the proposed feature selection approach outperforms the machine learning classifiers trained without reduced features.

REFERENCES

1. Moskovitch, R., Stopel, D., Feher, C., Nissim, N., Japkowicz, N., and Elovici, Y, "Unknown Malcode Detection and the Imbalance Problem," Journal in Computer Virology, Springer. 5: 295-308, 2009.
2. O'Kane, P., Sezer, S., McLaughlin, K. and Im, E.G., "SVM training phase reduction using dataset feature filtering for malware detection," IEEE transactions on information forensics and security, 8(3), pp.500-509, 2013.
3. Chih-Ta Lin, Nai-Jian Wang, Han Xiao and Claudia Eckert, "Feature Selection and Extraction for Malware Classification," National Taiwan University of Science and Technology, Taipei, Taiwan, 2015.
4. Nataraj, L., Karthikeyan, S., Jacob, G. and Manjunath, B, "Malware Images: Visualization and Automatic Classification," Proceedings of the 8th International Symposium on Visualization for Cyber Security, Article No. 4, 2011.
5. Kumar, A., Kuppusamy, K.S. and Aghila, G., "A learning model to detect maliciousness of portable executable using integrated feature set," Journal of King Saud University-Computer and Information Sciences, 2017.
6. Shafiq, M.Z., Tabish, S.M., Mirza, F. and Farooq, M., "Pe-miner: Mining structural information to detect malicious executables in realtime," In International Workshop on Recent Advances in Intrusion Detection, pp. 121-141, Springer, Berlin, Heidelberg, 2009.
7. M. G. Schultz, E. Eskin, F. Zadok, and S. J. Stolfo, ''Data mining methods for detection of new malicious executables,'' in Proc. IEEE Symp. Secur. Privacy. S&P, pp. 38–49, 2000.
8. J. Z. Kolter and M. A. Maloof, ''Learning to detect malicious executables in the wild,'' in Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2004, pp. 470–478.
9. R. Tian, L. M. Batten, and S. C. Versteeg, "Function length as a tool for malware classification", *Proceedings of the third international conference on malicious and unwanted software (MALWARE)*, pp.69–76, 2008.
10. Ye, Y., Wang, D., Li, T., Ye, D., Jiang, Q., 2008. An intelligent pe-malware detection system based on association mining. J. Comput. Virol. 4 (4), 323–334. Yonts, J., 2012. Attributes of Malicious Files. SANS Institute InfoSec Reading Room.
11. Meiri, R. and Zahavi, J., 2006. Using simulated annealing to optimize the feature selection problem in marketing applications. European Journal of Operational Research, 171(3), pp.842-858.
12. Fang, Z., Wang, J., Geng, J. and Kan, "Feature Selection for Malware Detection based on Reinforcement Learning," *IEEE Access*, pp.1-1, 2019.
13. J. Bai and J. Wang, ''Improving malware detection using multiview ensemble learning,'' Secur. Commun. Netw., vol. 9, no. 17, pp. 4227–4241, 2016.
14. S. Kim, ''PE header analysis for malware detection,'' M.S. thesis, 2018, vol. 624.
15. I. Santos, F. Brezo, X. Ugarte-Pedrero, and P. G. Bringas, "Opcode sequences as representation of executables for data-mining-based unknown malware detection", *Information Sciences*, 231, pp. 64-82, 2013.
16. Wang, T.-Y., Wu, C.-H., Hsieh, C.-C., "Detecting unknown malicious executables using portable executable headers" In: INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on. IEEE, pp. 278–284.
17. D. Kong and G. Yan, "Discriminant malware distance learning on structural information for automated malware classification" In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 1357-1365, 2013.
18. H. Kim, J. Kim, Y. Kim, I. Kim, K. J. Kim, and H. Kim, ''Improvement of malware detection and classification using API call sequence alignment and visualization,'' in *Cluster Computing*. New York, NY, USA: Springer-Verlag, 2017, pp. 1–9. [Online]. Available: http://dx.doi.org/10.1007/s10586-017-1110-2
19. M. Imran, M. T. Afzal, and M. A. Qadir, ''Similarity-based malware classification using hidden Markov model,'' in *Proc. 4th Int. Conf. Cyber Secur*., Cyber Warfare, Digit. Forensic (CyberSec), Oct. 2015, pp. 129–134.
20. R. Islam, R. Tian, L. Batten, and S. Versteeg, "Classification of malware based on integrated static and dynamic features", *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 646-656, 2013. Available: 10.1016/j.jnca.2012.10.004.
21. K. Rieck, P. Trinius, C. Willems and T. Holz, "Automatic analysis of malware behavior using machine learning", *Journal of Computer Security*, vol. 19, no. 4, pp. 639-668, 2011. Available: 10.3233/jcs-2010-0410.
22. Sutter, J.M. and Kalivas, J.H., "Comparison of forward selection, backward elimination, and generalized simulated annealing for variable selection," *Microchemical journal*, *47*(1-2), pp.60-66, 1993.

AUTHORS PROFILE

S. Abijah Roseline is currently pursuing Ph.D. degree in School of Computing Science and Engineering in VIT University, Chennai, India. She received the B.E. degree in Computer Science and Engineering from Vel's Srinivasa College of Engineering and Technology (Affiliated to Anna University), Chennai, India in 2008, and the M.E. degree in Computer Science and Engineering from MNM Jain Engineering College (Affiliated to Anna University), Chennai, India in 2011. She has published papers in reputed International Conferences. Her research interests include cybersecurity, malware detection, computer vision, and machine learning.

Dr. S. Geetha is a Professor in School of Computing Science and Engineering, VIT University, Chennai Campus, India. She has received the B.E., and M.E., degrees in Computer Science and Engineering from Madurai Kamaraj University, India in 2000 and Anna University of Chennai, India in 2004, Ph.D. Degree from Anna University in 2011, respectively. She has more than 18 years of rich teaching and research experience. She has published more than 80 papers in reputed International Conferences and refereed Journals. Her research interests include steganography, steganalysis, multimedia security, intrusion detection systems, machine learning paradigms, and information forensics. She joins the review committee and editorial advisory board of journals like IEEE Transactions on Information Forensics and Security and IEEE Transactions on Image Processing, Springer Multimedia Tools and Security, Elsevier – Information Sciences. She has published 4 books. She has given many expert lectures, keynote addresses at international and national conferences. She has organized many workshops, conferences, and FDPs. She is a recipient of University Rank and Academic Topper Award in B.E. and M.E. in 2000 and 2004 respectively. She is also the proud recipient of ASDF Best Academic Researcher Award 2013, ASDF Best Professor Award 2014, Research Award-2016 and High Performer Award – 2016, from VIT University. She serves as a Life Member in HKCBEES, ISCA, IACSIT, and IAENG.