Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 02, February-2020

Software Piracy Detection using Deep Learning Approach

Sonal Bhattar
Dept. of Computer Enginerering
D.Y Patil Institute of Engineering and Technology,
Ambi, Pune, India.

Mrunal Gaikwad
Dept. of Computer Enginerering
D.Y Patil Institute of Engineering and Technology,
Ambi, Pune, India.

Pratibha Kasar
Dept. of Computer Enginerering
D.Y Patil Institute of Engineering and Technology,
Ambi, Pune, India.

Yash Chikane
Dept. of Computer Enginerering
D.Y Patil Institute of Engineering and Technology,
Ambi, Pune, India.

Pritam Ahire
Assistant Professor,
Dept. of Computer Enginerering
D.Y Patil Institute of Engineering and Technology,
Ambi, Pune, India.

***Abstract*— Software piracy can be referred as illegally stealing citations. Currently, every other installed software is pirated. There are many scenarios of this happening, the attacker may crack the original legal software and re-construct or re-design the logic into other programming language or may change minor details of the software. It is very exasperating to catch such assaulters malicious activities as all the programming language have their own syntax and semantic structures. Currently, software piracy is high risk for security of software. It may cause reputational and economic damages. Now a days every other software is pirated there are many scenarios in which it can occur, the programmer may crack the original legal software and reconstruct or re-design the logic into other programming languages or may change the minor details of the software so we proposed a combine Deep learning approach to detect the pirated software. The Tensor Flow deep neural network is proposed to identify pirated the techniques like Tokenization and weighting are used to filter noisy data. The dataset is collected from Google code Jam (GCJ) to find the software piracy. The process of software piracy is very exasperating to each such assaulter malicious activities as all the programming languages have their own syntax and semantic structure. The experiment result shows that how much percentage of software code is plagiarism which be effective from current available methods.**

***Keywords*— *Deep Learning, Machine Learning, Tensor Flow, Piracy, Neural Network, Plagiarism, TF-IDF*.**

I. INTRODUCTION

In today's world software piracy is high risk to compromise the security in computer world. The detection of software piracy is the main aim in the field of cyber security. In proposed system, a combined deep learning approach is proposed to identify and detect pirated software. This involves two steps: First is preprocessing of the code collected from the GCJ (Google Code Jam) which breaks code into small pieces. And the second step is identification of software piracy using plagiarism which uses Tensor Flow Neural Network.

Proposed system will help to avoid the reputational and economical damages to the software industry. The traditional methods available may solve the concern but high computational cost will be needed to do so. The proposed system will try to detect software piracy by providing less computational cost to improve the accuracy.

II. RELATED WORK

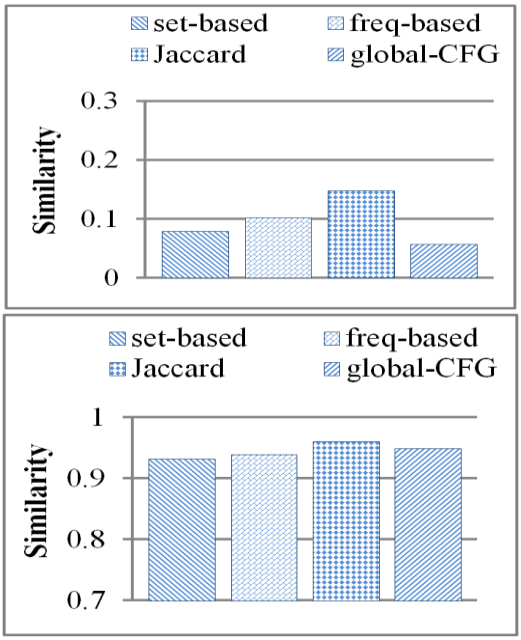
According to Author, al[4] of Camerino University, there are various challenges in cyber security one of them include software piracy. Currently, every other available installed software application is pirated. Software piracy is the act of copying distributing, using software illegally. The cracker or attacker may crack the original software and re-structure it into his own form. As different programming languages have different semantic and syntactical structures, the crackers may redesign the software into some another type of programming language. In this reference, they proposed a combined deep learning approach to identify the pirated and malware attacks on industrial IoT cloud. The TensorFlow deep neural network is designed to capture the pirated software using source code plagiarism. Further, the deep convolutions neural network is designed to capture the malicious patterns of malware through binary visualization. The combined solutions of the proposed approach are much promising in terms of classification performance.

According to Basel Halak and Mohammad El-Hajjar, al[6] they adapted the plagiarism prevention and detection techniques. Plagiarism is stealing someone else's work or publication and representing them as on original work. According to this reference of the paper there are mainly two techniques of preventing plagiarism. First is about plagiarism prevention using unique assignment and second is about plagiarism prevention using Individual presentation. In which we detect and prevent plagiarism based on unique specification this type of plagiarism is more difficult to highlight using common plagiarism detection tool. The use of individual presentation technique can also be effective in detecting plagiarism of undocumented ideas in group design project. In this reference, these two techniques is based on the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 02, February-2020

use of individual design specification in the course works and individual presentation ofs the work. They give a examples where these techniques have been used in different modules in the university of Southampton and have shown how this techniques canbe effective in reducing plagiarism and at the same time improving student's understanding.

According to Dept. of computer software Hanyang University, Korea, al[7] they Suggested a software plagiarism using API- labeled control flow graph(A-CFG). A-CFG represents the sequences and frequencies of API which are rarely changed by semantic preserving transformation attacks. They performed scalable comparison between A-CFG as showing each A-CFG as a single score vector through RwR. Ex: Results shows that our proposed system outperforms existing methods in terms of both accuracy and creditability in a reasonable computational time.



According to Author, al[8] Department of MCA, PES institute of technology proposed that estimate 60% of data mining tasks are mainly spend on preprocessing the data. So, they performed a cyclomatic complexity analysis which is used to indicate the complexity of program which is turn enables us to calculate the number of coding errors along with their source code complexity. Through this phase errors are separated both at instance and schema level from single and multiple source of data followed by this the second phase involves a sequential flow analysis for data preprocessing of spatial data, multidimensional data, web log data etc. It was observed that after performing this two phases, preprocessing is improved for further data mining process. It is known that data cleaning or data cleansing is one of the measure step in KDD process of data meaning two generate accurate and appropriated data for different data mining tasks. Data may come from various source such as homogenous databases, heterogeneous databases, flat files and other.

According to Faith Ertam and Galip Aydin, al[9] we studied about deep learning using Tensor-Flow. Deep learning approaches are rapidly increasing in today's world. It provides effective and accurate solutions of big data analysis. This study classification made by using Tensor-Flow. Tensor-Flow is an open source software library. It is developed by Google for the purpose of numerical computation. In this reference, a classification task was carried out on the MNIST data set which is widely used for this purpose. Different activation function were selected the system to test the accuracy of the classification of the system, ReLu, eLu, tanH, Sigmoid, SoftPlus and SoftSign activation functions were used for this purpose. The increase in the number of the interaction showed in increase in the accuracy values, but the total classification by applying different neural network architecture.

According to Shahzad Qaiser and Ramsha Ali, al[10], they use TF-IDF algorithm. This algorithm is easy to implement and is very powerful but one cannot neglect its limitations. In today's world of big data, world requires some new techniques for data processing, before analysis is performed. Many researches has proposed an improved form of TF-IDF. The proposed algorithm incorporated the hill climbing for boosting the performance. A variant of TF-IDF has also been observed that can be applied in cross language by using statistical translation. There are many techniques or algorithm that can be used to process data but this study is focused on one of these, known as TF-IDF. TF-IDF is a numerical statistic that shows the relevance of keyword to some specific documents or it can be said that, it provides those keywords, using which some specific documents can be identified or categorized.

According to Prafulla Bafna, Dhauya Pramod and Anagha Vaidya, al[11] we studied that document clustering using TF-IDF approach, handling repositories of unstructured and semi-structured document is difficult TF-IDF works by comparing the local documents with the main document, identifying the most similar versions of same document or to get and extract relevant set of documents from a huge repository of documents. Thus, these techniques eliminate the most similar term and extract only the relevant term from the main document. This can be used in variety of problems including email sorting, research paper sorting etc. In this reference, they work in a two phase the first phase to detect the most suitable algorithm and in second phase it is applied to extend data set. The results of the both phase are verified using different cluster analysis techniques. It can be used for wide ranges of a problem varying from email shorting research paper sorting. Results obtain after processing various data sets shows efficiency of the algorithm. Further work is to used better. Semantic relativity concept which might be domain specific but provide the better results.

According to Vijayashri Losarwar, Dr. Madhuri Joshi, al[12] they suggested the importance of data preprocessing methods and various steps involved in getting the required content effectively. It is well known that over 80% of the time required to carry out any real world data mining project is usually spent on data preprocessing. Data preprocessing lays the groundwork for data mining. Web mining is to discover and extract useful information from the world wide web. It involves the automatic discovery of patterns from one or more web servers. This helps the organizations to determine the value of specific customers, cross marketing strategies, etc. A complete preprocessing techniques is being proposed to preprocess the web log for extraction of user patterns. Data cleaning algorithm removes the irrelevant entries from web log and filtering algorithm

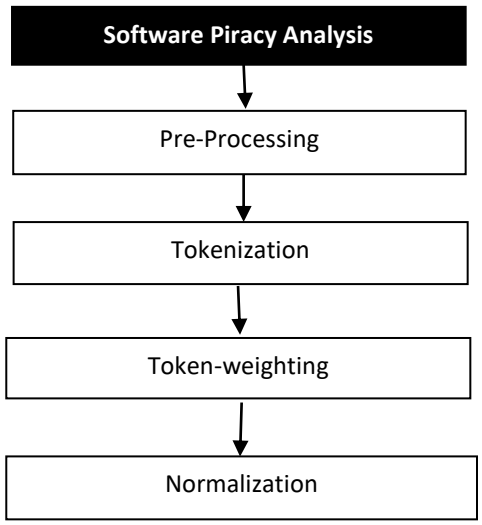
International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 02, February-2020

discards the uninterested attributes from log file. An important task in any data mining application is the creation of a suitable target data set to which data mining and statistical algorithm can be applied. This is particularly important in web usage mining due to the characteristics of click stream data and this relationship to other related data collected from multiple sources and across multiple channels. The data preparation process is often the most time consuming and computationally intensive step in the web usage mining process, and often requires the use of special algorithms and heuristics, not commonly employed in other domains.

III. OBJECTIVE OF THE SYSTEM

1. To provide detection about malware and threads in files across network.
2. To detect software piracy using deep learning approach.
3. To identify pirated software using Tensor-Flow neural network.
4. To detect malicious infections using convolutional neural network.
5. To avoid economic and reputation damages causes due to threads.

IV. PROPOSED METHODOLOGY



Module used in proposed system module:

1. *Pre-processing and feature extraction:*
Pre-processing methods are used to break the source code in small pieces for deep analysis. Then break codes are converted into meaningful information and remove the noisy data.

Pre-processing is an important issue for both data warehousing and data mining, as real world data tends to be incomplete noisy and inconsistent. The data pre-processing to remove some more anomalies specific to each data which may not allow the data mining process to continue further. The data pre-processing techniques and approaches are based on time-series data, stream data, sequence data and textual data, etc. In data pre-processing the input data is tested against specific data type like spatial or multimedia or web data or stream data or time series data etc., based on the type of data selected or chosen, respective data cleaning methods are applied and errors, inconsistencies, in completeness redundancy or duplicates from data is remove[8].

2. *Tokenization:*
Tokenization process is used to transform the cleaned data into useful tokens. This process is replacing sensitive data with unique identification symbols which retain all the essential information about the data. The purpose of Tokenization is to map out sensitive data-typically payment and an bank account numbers. For example, Tokenization is about replacing the identifying information with the substituted credentials. In programming language Tokenization is tokenizing a string. It denotes splitting a string with help of tokenization. We can make gain difficult information for Hackers mining, tokenization is used for security purpose, for example: In bank credit card system before tokenization introduced credit card number were stored in database [13].

3. *Weighting Techniques:*
Weighting technique is used to zoom the contribution of the token.

V. SYSTEM ANALYSIS

- DEEP LEARNING WITH TENSOR FLOW FRAMEWORK:

Tensor Flow has different types of layers which can be configured for complex computations, training the data. The in-depth learning approach is designed to identify similar source codes in different types of programing languages using Tensor Flow framework. Then, the extracted similar codes are used to identify the pirated software. The weighting values are used as input to the deep learning model. The deep learning approach is enhanced using drop out layer in the context of activation and loss function, optimization and learning error rate. The software plagiarism measures may be used to investigate the code similarity in pirated software. The contributions of similar tokens are measured using weighting techniques such as TFIDF and Logarithm of term frequency are used.

Mathematical Model Used:

$$TFIDF(t,d,D) = tf(t,d) \times idf(t,D)$$

Where,

t denotes token,

f denotes the number of frequency,

d denote every individual document,

D denotes all documents used in the dataset.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 02, February-2020

VI. CONCLUSION

The industrial IOT based network is rapidly growing in the coming future. The detection of software piracy and malware threats are the main challenges in the field of cyber security using IoT-based big data. This system proposed a combined deep learning based approach for the identification of pirated and malware files. First, the Tensor Flow neural network is proposed to detect the pirated feature of original software using software plagiarism. We collected 100 programmer's source code files from CGJ to investigate the proposed approach. The source code is preprocessed to clean the noise and to capture further the high-quality features which include useful tokens. Then, TFIDF and LogTF weighting techniques are used to zoom the token in terms of source code similarity. The weighting values are then used as input to the designed deep learning approach. Secondly, we proposed a novel methodology based on convolution neural network and color image visualization to detect malware using IoT. We have converted the malware files into color images to get better malware visualized features. Then, system passed these visualized features of malware into deep convolution neural network. The experimental results show that the combined approaches retrieve maximum classification results as compared to the state of the art techniques.

REFERENCES

[1] Sohail Jabar, Kaleem R. Malik, Mudassar Ahmad, Omar Aldabbas, Muhammad Asif, Shehzad Khalid, Kijun Han,"A Methodology of Real-Time Data Fusion for Localized Big Data Analytics",March 15, 2018.

[2] Manisha Mishra, Monika Srivastava, "A view of Artifical Neural Network",Dr. Virendra Swarup Group of Institution Unnao,2014.

[3] Liping Yuan, Zhiyi Qu, Yufong Zhao, Hongshuai Zhang, Qing nian, "A convolutional neural network based on TensorFlow for face recognition "Lanzhou University,China,2017

[4] Farhan Ullah, Hamad Naeem, Sohail Jabbar, Shehzad Khalid, Muhammad Ahsan, Latif Fadi AL-turjman and Leonardo Mostarda, "cyber security threads detection in internet of things using deep learning approach," Sichuan University, Chengdu 610065,China,2017.

[5] Donato Malerba, "Mining Spetial Data: Opportunities and challenges of Relation Approaches ",University of degli study,
[1] Italy.

[6] Basel Halak, Mohammed El-Hajjar, "Plagiarism Detection and Prevention Techniques In Engineering Education", University of Southampton , Southampton, UK, 2016.

[7] Dong-kyu, Jiwoon Ha, Sang-wook kim, BooJong Kang "A Software plagiarism detection : A graph-based approcach", Hanyang University, October 2013.

[8] P.Sreenivas, Dr.C.V.Srikrishna, "An Analytical approach for Data Preprocessing", PES Institute of Technology, 27 February 2014.

[9] Faith Ertam, Galip Aydin, "Data Classification with Deep Learning using Tensorflow", Firat University, Elazig, Turkey, 2017.

[10] Shahzad Qaiser, Ramsha Ali, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents", School of Computer University, School of Quantitive Sciences University, Uttar Malaysia, July 2018.

[11] Prafulla Bafna, Dhanya Pramod, Anagha Vaidya, "Document Clustering: TF-IDF Aprroach",Symbiosis International University, pune,2016.

[12] Vijayashri Losarwar, Dr. Madhuri Josji, "Data Preprocessing in Web Usage Mining",Singapore, July 15-16, 2012.

[13] Jin Guo, "Critical Tokenization and its properties ", National University of Singapore.

[14] Fco.Mario Barcala, Jesus Vilares, Miguel A. Alonso. Jorge Grana, Manuel Vilares"Tokenization and Proper Noun Recognition for Information Retrival" Departmento de Computacion, Universidade da Coruna Campus de Elvina s/n, 15071 La Coruna, Spin,2002.

[15] Dong-Kyu Chae, Jiwoon Ha, Sang –Wook Kim, BooJoong Kang, Eul Gyu Im, "Software Plagiarism Detection: A Graph-Based Approach", Hanyang University, Korea, 19 June 2015.

[16] Elfwing, S., E. Uchibe,K.Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning", Neural Networks, 2018 .

[17] Ullah,F, "Software plagiarism detection in multiprogramming languages using machine learning approach. Concurrency and Computation: Practice and Experience"2018.