International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Cryptojacking Malware Detection using the Bayesian Consensus Clustering with Large Iterative Multi-Tier Ensemble in the Cryptocurrency in the Cloud

S. Balamurugan, M. Thangaraj

Abstract*: *Virtual Currencies and cryptocurrency are a trending digital currency method which uses the Blockchain technology. Cryptocurrency is a digital method designed to exchange the asset between the users based on a powerful cryptography which ensures the transaction are safe and controllable. We have various legal areas identified while using the cryptocurrency, as being the virtual currency, the amount of assets used by the users increases rapidly. With the increase in the asset the security breaches are one of the key vulnerable areas to focus. Cryptocurrency mining malware or Cryptojacking remains a trending terminology which identifies the malicious software or malware developed to use the data from the smart phones and computers. The major threat of the Cryptojacking is cryptocurrency mining without user's approval. This article implemented based on our CCEC Framework method published for Malware detection in SMS's for the Smartphone users. The article explains about how the Malware detected using the CCEC Framework. Malwares created in various format so identifying the Malware takes time before which user assets remains vulnerable. So, the proposed method ensures we have a reduction in time by using various online data sources to identify the Cryptojacking malware.

Keywords*: *About four key words or phrases in alphabetical order, separated by commas.

I. INTRODUCTION

Blockchain one of the most trusted method to transfer the data without involvement of any III party agents. The preferred method of transfer to use the middle man generally the Bank for the data fund transfer between the users A and B. Computers and the smart phone connected through the network in a cloud can use the blockchain transactions with the transactions recorded in the digital ledger.

Manuscript published on 30 September 2019
* Correspondence Author
S. Balamurugan, Research Scholar, Department of Computer Science, Bharathiar University, India),
mailtobalaselvam@gmail.com
M. Thangaraj, Associate Professor, Department of Computer Science, Madurai Kamaraj University, India)

During the transaction all the computers and smart phones connected to the network will approve the transaction for the details recorded in the ledger.

The network remains public or private network. When users decide whether they can use the public or private network. For the currency transactions the private network used [5]. After this the block to the chain so the fund transferred between the users A and B. The transaction visible to the users connected to the network.

Blockchain uses cryptography method to make sure the transactions stay safer within the network. The transactions verified through the group of people named Miners. With the blockchain there is no limitation on the transfer, the block will support transfer of currency, things and even the investment properties. Advantages with the blockchain no limitation in the transfer, blockchain technique used by the migrants to transfer the limitless currency which the existing banking method doesn't support. Blockchain with cryptography reduces Fake transaction, as all the transactions between the users recorded in the Digital ledger. Blockchain used by the users who access the internet. Survey result [2] show only 0.03% of the users across the globe using this technology. But blockchain moves as demanding technology in few years with the ability and security with the transfer. Users speak about the digital currencies used very often, there are around 1300 plus crypto currencies available, few crypto currencies named as bitcoin, ripple and Ethereum [3]. In general, we can term cryptocurrency an electronic transfer between two users which remains virtual. Cryptocurrency created by the users. The basic need to have knowledge on the currency and knowledge to generate codes.

Blockchain works as a secure method for all transactions with encryption. All the transactions encrypted when the User A initiates the transaction the details encrypted by user private key from the wallet. The other clients who authorizes the transaction uses the public key. The private key will be available only with the user A who initiates the transaction [4]. Also, only user A can sign the initiated transaction [11]. With these features' cryptocurrency remains a secure method for the users. Risks increases with usage of cryptocurrency; the cyber security attacks show a clear picture the intruder's usage of the cryptocurrency.

The intruder's generates crypto mining malware or Cryptojacking to get access to the user's computer and smart phone [5]. The method used by intruders through email or links.

The phishing emails with the links remains vulnerable. The user clicks on the link the malicious software or malware installed in the user machine. The user concludes just a link will not impact the data. Intruder's also hide malwares in the links within the android or ios applications. These links named as Ad blockers in the applications remains the threat for the smart phone [12]. The ad blocker downloads the malicious codes, and this transferred to the user smart phone [6]. The scripts with java codes used by cyber criminals to get access to the user phone. Once the user phone is accessible the credentials for the cryptocurrency remains with security breach.

The intruders can get access to the cryptocurrency without the knowledge of the users [8]. This led to the impact for the users with the currency. Researchers confirms intruders are mining cryptocurrencies through smartphones causing them to generate high heat, sometimes leads to damage. This involves connecting the victim's machine and performing the computations necessary to update cryptocurrency blockchains known as mining [4].

There raises a query on security with the digital currency, as blockchain remains very secure. No intruders can make the changes to the block or interrupt the block during the transactions. Cryptojacking method implied by the intruders not after the block created for the transaction [1]. The aim of the intruders to move the digital currency from the user wallet without any intimation to the users [7]. So, the intruders use various methods to use the wallet.

The user's smart phone which access the wallet will remains as threat. Single malicious email with link remains a source for the intruders to reach the wallet [14]. The link when accessed by the user remains a simple webpage. But the scripting or the malicious software which installed in the user's smart phone.

There will be no notification to the user about the Malware installed. The malware will remain silent for the intruders until it gains the control for the wallet [2]. Once the user accesses the wallet the credential shared by the Malware to the intruders. So, the user's digital currency will remain under the risk. The intruders can get access to the wallet and the digital currency transferred by the intruders. No nodes in the network which will approve the transaction knows the transaction initiated by the intruders.

This article we use the framework named CCEC which was used for identifying the Malware detection in SMS's data sets and the results were published. The same framework used as a reference to identify the Cryptojacking malware in the cryptocurrency user's smartphone. The data sets are referred from github.com, amazonaws.com, weebly.com, hrtests.ru, a. cuntflaps.me and few other online data sources.

II. RELATED WORK

One article explains about the various methods these privacy issues pops up. Based on the technology Open-PDS one of the recent frameworks presents a model for the deployment which includes the returning of the data computations, so the usage of raw data minimized [3]. With the industry prospective few leading organization's selecting to carry out their own authentication software using OAuth protocol [8] which works as source for authentication.

Security being a major area for development many techniques targeted towards the user's personal data. Data anonymization method developed to protect client personal information. K-anomity, works as a common property of anonymized datasets which requires a sensitive information of each record is distinguishable from at least k-1 other records [7].

Recent research has demonstrated how anonymized datasets engaging these practices become de-anonymized. The small amount of data points or high dimensionality data handled securely. Other privacy-preserving approaches include discrepancy secrecy, a technique that bothers data or adds clatter to the computational process used to sharing the data and encryption schemes that let running computations and queries over encrypted data.

Encryption handled by fully homomorphic encryption approach, the method agrees any processing to use an encrypted data. But this approach is not widely used for the security implementation with blockchain [8]. Hawk framework used for building secure blockchain. Using Hawk, a normal user can easily draft a Hawk program without having to start any cryptography. Compiler build executes the program which generates a cryptographic protocol to be used by the client and nodes [2].

New approach proposed based on token. The token generally refers to an asset owned by the users, the users sell this to the other users during a sale period called Initial Coin Offering (ICO). Tokens categorized as utility and security. The security tokens usually formulate their value from an external asset. The tokens with the security subjected under the various regulations. If the ICO security tokens miss the regulations, it's considered as the security breach [2].

The strength of blockchain process uses cryptography which is not part of other systems. The cryptocurrency not managed by a single node. But there are researches who confirms cryptography are not the powerful algorithms [2]. There remains a method to break the algorithms in turn will break the entire system. Few algorithms which referred for the use are ECDS, SHA-256 and RIPEMD-160 [2].

SHA -256 stand for "Secure hash Algorithm" generates unique 256-bit of 32-byte signature for a text string. Processing time for the block with this algorithm takes six to ten minutes. This works at the hash rate of Giga hashes per second [3].

Scrypt algorithm needs high memory and works to find large-scale custom hardware attacks. The Scrypt algorithm works simple when compared with SHA-256 algorithm [2]. The hash rate measured by the kilohashes per second for the Scrypt algorithm and the mining performed on a computer based on its memory usage, Graphic processing unit [3]. The X11 hashing algorithm developed by Evan Duffield, this algorithm uses sequence of eleven scientific algorithm. The benefit of this algorithm over the other algorithm is the usage of less memory. The GPU require approximately less wattage and run 30-50% cooler when compared to Scrypt [3].

III. CCEC FRAMEWORK

The Consensus Clustering with Ensemble Classification Framework defines the process to move the unstructured data set which is preprocessed, data selection, data prepared with the selected data to derive a trained set using clustering. The trained data set remains the source to identify the malware types in new datasets which are not structured, and which requires malware identification.

A. *Unstructured Data Set as Source*

The data source is from multiple online data sets like Kaggle, UCI repository, Buzzfeed, Socrata Open Data, Google Data sets, Wikipedia, Quandl, Quantopian and Fixshare. The unstructured data set with different types are used for this experiment. We have used R platform for preprocessing the unstructured data.

B. *Data Pre-Processing*

Data preprocessing a method to transform unstructured data to readable and standard format for processing the data. The data is preprocessed using instance-based methods in R. This remains effective to identify the missing values in the data sets and identifying the noisy data by resolving the data consistency [13]. The data source remains with multiple formats from various online data sources. The data requires to be standardized to one format for processing through BCC. Data preprocessing is efficient method to eradicate the incorrect the data which improvise the results of the experiment with the real-world data.

C. *Data Transformation*

This process includes the Data preparation. In the data preparation the missing data identified, and the missing data will be eradicated following by changing the variable types. The data preparation is time consumption activity [11], but this remains very effective during the actual run and testing. The unstructured data sets contain lot of entries irrelevant to malware, this needs to be removed from the data sources. In wide-ranging, the data preparation is the process to collect the data from multiple data sources, cleaning the data and consolidation of data into one data source for analysis [1]. Data Transformation is transferring the data from multiple data formats in to one single data format. Data Transformation is used as this still can reduce the run time with the complex data sets.

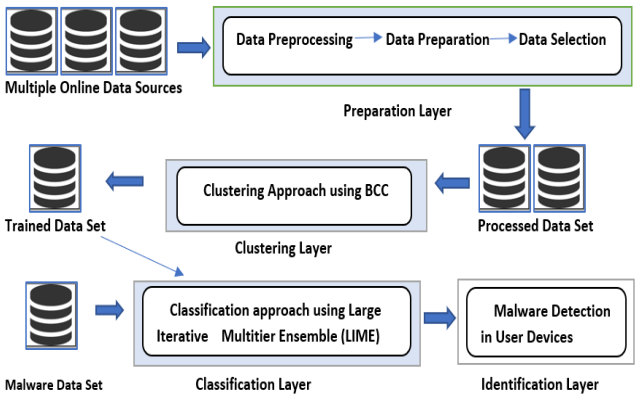

Fig.1 Illustrates the Architecture of the CCEC Framework

D. *Data Selection*

The data sets are transferred into one single format and a single file at the end of Data Preprocessing. The data preparation is processed using the data selection. The method used for the data selection by singular value decomposition (SVD). The selection required to identify which data will be considered for analysis. R programming used for the entire preparation layer.

E. *Clustering Layer – BCC*

In most of the methods derived they uses the classification technique once the preprocessing completed. Classification identified with the class labels, but with this framework after the data selection and data transformation the malware clustering implemented. Instead of classification, we have used the clustering to form the malware defined data set. Clustering is unsupervised learning [12], this reduces the duration compared to the classification methods [2]. Once the data sources are clustered into malware cluster and no malware cluster, they are grouped based on the malware family using the cluster labels. In general, the clustering works based on the objects, the method proposed in this framework is integrative statistical model that works to identify the clustering of objects for each family of malware data set. Bayesian Consensus clustering termed as cluster ensembles used with the CCEC framework.

Consensus clustering can be used with its benefits for identification of same data set from multiple data sources or making multiple runs with the same algorithm [2]. The existing clustering methods have few challenges in malware detection. The clustering technique don't address the user requirement when the size of the data set is high and increases the time complexity with the increase in the data size [2]. The cluster effectiveness calculated based on the distance between the clusters with the existing algorithms. But with the consensus clustering the it remains easier to identify the number of clusters even during the analysis the prediction of number of clusters remains impossible with multiple data sources. When there are no enough details about the external objective criteria the validation is not effective. There are few iterative clustering algorithms like SOM, k-means which derives the defined clusters and boundaries for the clusters.

Consensus clustering used to derive a method which represents the accord across iterative runs of the BCC algorithm, to derive the number of clusters in the data and to assess the stability of the discovered clusters. The CCEC framework uses the scalable Bayesian method for clustering. This method used to estimate based on the consensus clustering and the source specific clustering. Bayesian Consensus Clustering (BCC) termed under the soft clustering ensemble method [3], this method used with the large data sets as remains effective with the reduction of time complexity with each run [2].

BCC clusters the data set based on the malware family and the cluster identified based on the development of the malware and how its spread across the devices. The two clusters are derived out of the BCC Malware and No Malware.

Malware contains the various malwares which is clustered internally based on the family they belong to. No Malware cluster is the data which is not impacted by the Malware. Malware clusters formed at the end of the iterative run remains 11 from families of malware. The existing methods uses only the classification which remains less effective with the online multiple data sources [12]. As there requires a need for the better method, we have used clustering with classification. BCC works effective with the multisource data and they form a heterogeneous cluster from the multiple data sources [8].

BCC method results with high accuracy in the clustering of malware based on their families and reduced run time even with the complex data sources [11]. Also, analysis results show the multi data source is a challenging for the classification method as the results are not so effective and the time complexity increases with the larger data sets [6]. To address these challenges the CCEC framework is proposed.

F. *BCC Algorithm*

BCC algorithm uses the Gibbs sampling model which uses the full posterior probability distribution across the various variables available in the data set. Sampling is based on the Monte Carlo Markov Chain (MCMC) method which remains effective in identifying the clusters [9]. BCC identifies the malware and cluster it based on the integrative clustering. The BCC algorithm does not assume any specific form for the fm and the parameters Θmk. We use conjugate prior distributions for ∏m, A, and (if possible) Θmk.

Markov chain Monte Carlo (MCMC) proceeds by iteratively sampling from the following conditional posterior distributions:

1. Start at the data source Θm|Xm where the Bm ~ pm (Θmk | Xm, Bm) for k = 1,… K
2. Move to the new position based on the Malware identified nearer in the data set

Bm|Xm, Θm, Am, C ~ P(k | Xmn, Cn, Θmk, Am) for n = 1… N where

P(k | Xmn, Cn, Θm) α √ (k, Cn, αm) fm(Xmn | Θmk)

3. Accept the place based on the place adherence to the data and prior distribution

. αm | C, Bm ~ TBeta(Am + ŧm, bm + N - ŧm, 1\K), where ŧm is the number of samples n satisfying Bmn = Cn

4. If the Malware is added to the cluster move to the next data in the data set to identify the next place

C | Lm, ∏, A ~ P(k|∏, {Bmn, αm} mM=1) for n = 1,….N where P(K|∏ {Bmn, Am} mM=1) A∏k

∏mM=1 √ (k,Bmn, Am)

5. Repeat the iteration with the entire data set and return with the various places the malware content is detected

∏ | C ~ Dirichlet (β0 + ρ), where ρk is the number of samples allocated to cluster k in C

Input: Multi data source which contains Malware contents in Cryptocurrency

Output: Two Clusters (Malware and No Malware)

Method: Gibbs Sampling Procedure

This algorithm can be suitably modified under the assumption that A1 = …AM.. Each sampling iteration produces a different realization of the clustering's C;B1; _ _ _ ;Bm, with Malwares clustered based on the family of the Malware identified and together these samples approximate the posterior distribution for the overall and source-specific clustering.

The clusters B1 to Bm represents the family of malware clusters. The estimation of the clusters requires the interpretation of each clusters based on their family B1…. Bm. In this respect methods that aggregate over the MCMC iterations to produce two clusters. The first cluster for Malware which contains all the Malware Spams and the other cluster which do not contain any Malware and listed as No Malware cluster. BCC implementation is to identify two clusters from the online data sets with different formats. The Malware and no Malware clusters are derived from the multiple data sources of cryptocurrencies. The Figure 2 and 3 represents the various Malware clusters derived using the BCC Algorithm. The malware derived with the malwares clustered based on the family of the type of the malware. They are the derived from heterogeneous clusters which will be used as the trained data set for the Classification Layer [3].

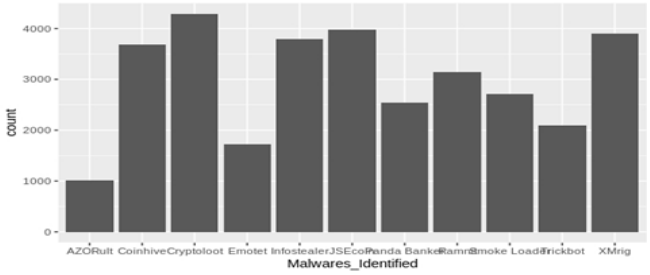

Fig 2 Malware Family identified based on BCC

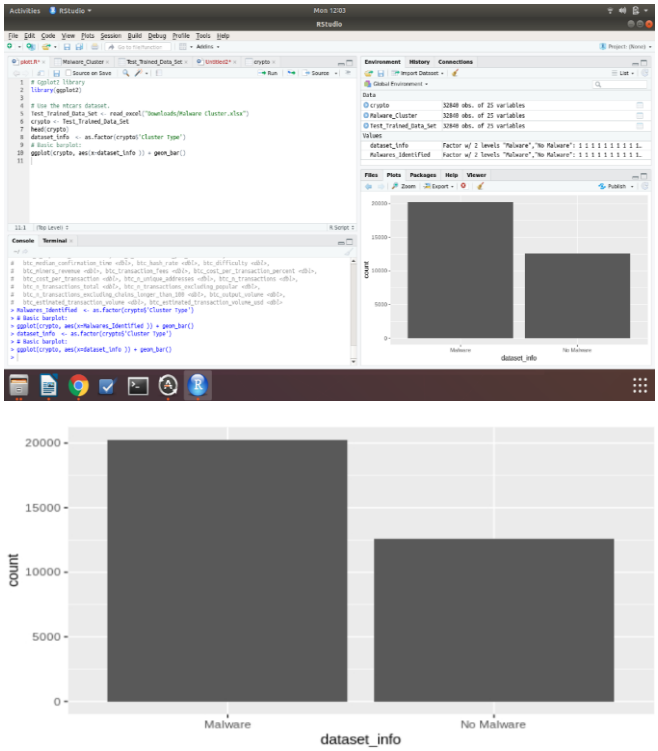

Fig. 3 Spam clustering using R

G. *Classification Layer – LIME Large Iterative Multi-Tier Ensemble (LIME) Classifier*

Data classification is defined as a technique used to collect and form the data based on the categories. Classification process in a simple term process makes the data easier to identify and track, data can be retrieved without any complexity. Data classification also involves the tagging of data. The CCEC framework uses Large Iterative Multi-Tier Ensemble classification method (LIME) for the classification of the data to detect the malware. LIME receives the trained data set from the BCC (clustering layer) and it assists the LIME method for detection of malware from the unstructured data set which requires malware identification.

LIME works based on the Base classifier and Ensemble meta classifier to identify malware from the various iterative runs. The base classifier is derived from the iterative run and further used for identifying the ensemble meta classifier. The approach with LIME is generally a bottom up ensemble generation approach [9]. With our experiment various base classifiers are used, Random Forest resulted with high accuracy in detecting the malware.

The method outperformed with other methods like ZeroR, SMO, SGD, Voted Perceptron, Naïve Bayes, OneR, J48, LWL and BayesNet. Random forest being a supervised learning method, it builds an ensemble based on the decision trees trained with the bagging method [6]. Random forest works as the collection of various random trees.

This article demonstrates how the LIME can be implemented with the various tiers. The method implemented with the various ensemble meta classifiers into the various tiers. The top most tier ensemble meta classifier is developed from the ensemble meta classifier from the below tier. The process is repeated until the lowest tier. The classification of the data with LIME implemented with Weka Explorer. As the clustering has advantages over classification, the clustering was implemented prior to the LIME.

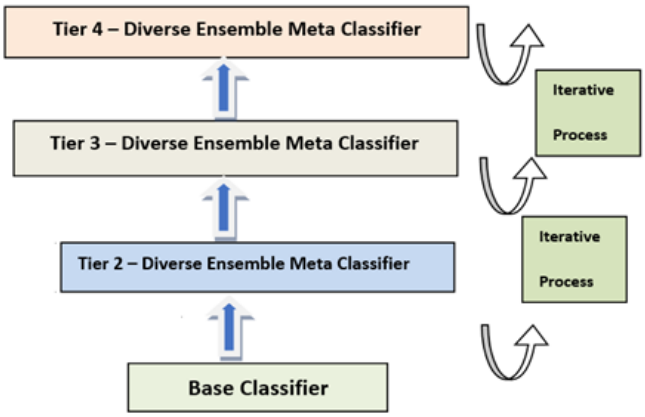

Fig. 4 LIME Classification

H. *Base Classifier*

Base classifier identified by the selecting the best methods which are available from the existing methods like ZeroR, SMO, SGD, Voted Perceptron, Naïve Bayes, OneR, J48, LWL and BayesNet. Random forest remains a better method as it has high accuracy in detecting the malware. The experiment used to compare the performance of multiple classifiers which can detect the malware from the data source. The figure 5 shows the AUC results of the base classifiers. Random Forest remains more accurate in the malware detection followed by NaïveBayes.

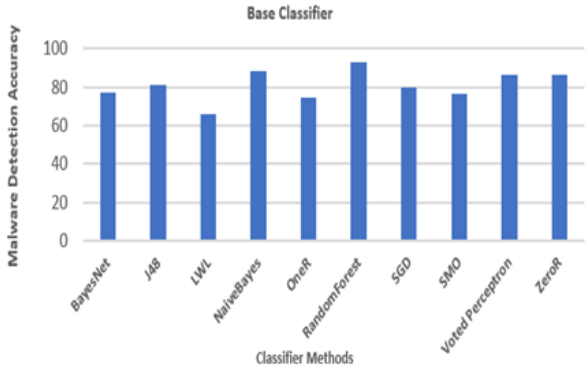

Fig. 5 AUC of the Base Classifier

I. *Ensemble Meta Classifier*

In our experimental run we have used Simple CLI command line in WEKA to investigate the AUC of the following ensemble meta classifier. Adaboosting, Bagging, Filtered Classifier, Logitboost, Multiclass Classifier, Random Committee and Stacking [1]. Multiple methods are included, and our experiment results are compared with the other ensemble classifiers, to see how the classifier works with the malware detection. There is no limitation in selecting the classifiers. We have used the maximum number of classifiers to identify the base and ensemble meta classifier.

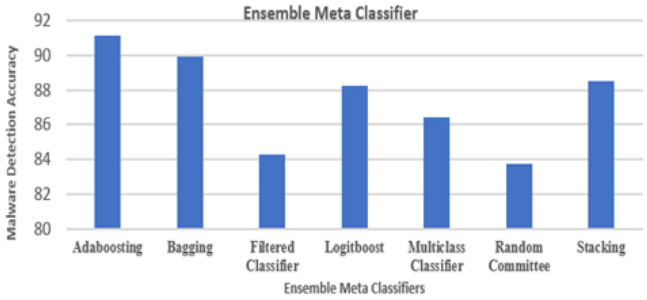

Fig. 6 AUC of the Ensemble Meta Classifier

AUC of the ensemble meta classifiers are represented in the Fig 6. The performance improved when compared with the base classifiers in malware detection. In these experimental run the Random Forest used as the base classifier.

J. *LIME Extraction*

The trained data set from the cluster layer is derived with malware family class labels and other parameters. The class with the experiment malware is the category of the classifier, the resulting cluster with the experimental output remains Malware and No Malware clusters. The malwares derived from our experiment are Coinhive, AZORult, Cryptoloot, Emotet, Infostealer, JSEcoin, Panda Banker, Ramnit, Smokeloader and Trickbot.

The method used with LIME for detecting malware is word Level n-gram analysis. N-grams works best with the Natural Language Processing (NLP). LIME detects the malwares from the unstructured data source using the trained data set. Using the trained data set reduces the runtime and the accuracy increased even with the complex unstructured data sets.

Trained data set will be refreshed in the regular intervals, so the new malware content will be updated [1].

For our experiment we have used word level n-grams method to read the data from the unstructured data set to classify the malwares. The word level n-gram used as byte level and character level n-grams. This method yields high accuracy result when compared with another n-grams method [3]. The advantage of using word level n-grams to avoid scant data that may be available when the character level n-gram alone used. The word level n-grams provides less character combinations than the word combinations. So, the less N-grams will have a very high zero frequency. The term used TF IDF (Term Frequency Inverse document frequency) to calculate the weight, to select word level n-grams to reduce the number of features [1].

Let us consider the data set D, which consist of instances of the malware b. For a sequence a and b the instance to detect the malware m is defined as N(a,b). The N is the number of times content a appears in b. The collection R = {r1… rf} is termed as the Term frequency with the malware detected from the data set.

The term frequency of word a ∑ R is the instance of the malware b defined as TF(a,b) were j occurs in the malware data set b normalized over the number of occurrences of the malware. The term frequency inverse document frequency of a word w in instance of malware b, or TF-IDF weight of b in j is defined by

$$TF\text{-}IDF(a, b) = TF(a,b) * IDF(a,b) \quad (1)$$

IV. IDENTIFICATION AND APPLICATION LAYER

Implementing BCC and LIME the malware contents are extracted from the unstructured data source. The malware content with its family type is listed so this can be shared to the end users through portal and mail to educate the users. The cryptocurrency data set contains the various list of malwares received by the users through links and mails. The cryptojacking content is identified in this layer. Application layer is the Graphical user interface which will list out the malware content in the cryptocurrency and wallets. The application layer generates the messages with the malware content classified as malware and no malware.

V. EXPERIMENT EVALUATING PERFORMANCE

The experiment results are available with various tiers, with the tier 4 the last tier derived using the meta ensemble classifier from the tier 3 shared in the Fig. 7.

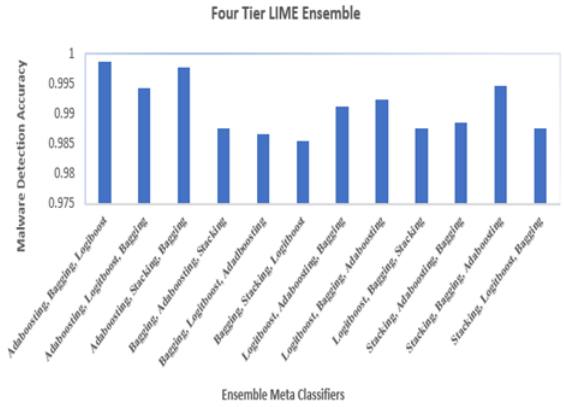

Fig. 7 Results of the AUC of Four Tier LIME Classifiers

The results for the tier 3 derived from the tier 2 is shared in the Figure 8. The results for the tier 4 with Fig 7 and tier Fig 8 for tier 3 are derived using the word level n-grams. The results shared are the high accuracy results used with iterative runs to detect the malware from the data sets. To get the results with high accuracy the various ensembles are merged and with various meta classifiers and best method is identified based on the execution.

The experiment results show the combination of meta ensemble classifiers Adaboosting, Bagging and Logitboost remains the best method with the malware detection. The other methods don't turn un effective, but this method has the highest AUC value in detecting the malwares.

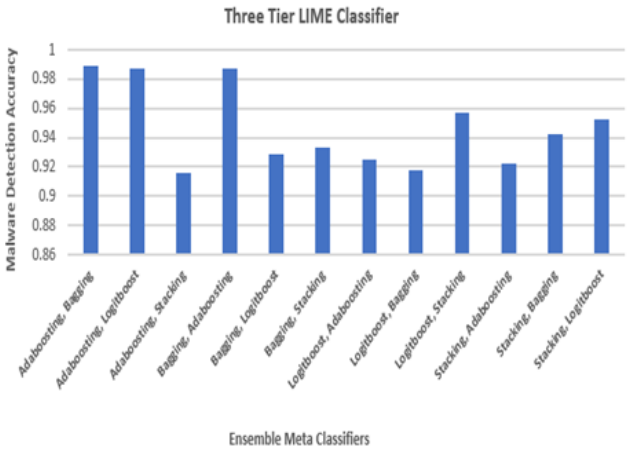

Fig. 8 Results of the AUC of Three Tier LIME Classifiers

The result shown above don't make a negative impact on the other combination of algorithms, they provide less accurate results in malware detection. Experiment was conducted to identify the various classification algorithm works in the malware detection. The result of the experiment demonstrates there are around 10+ family of malwares in relation to the cryptojacking. The Malwares classified are Coinhive, XMrig, JSEcoin,Cryptoloot, Emotet, Ramnit, Smoke Loader,Trickbot, AZORult Infostealer and Panda Banker.

The method used to identify the performance of our framework in the Weka tool is 10-Fold Cross Validation. This method remains effective when we have used this in our prior implementation for identifying SMS malwares [1]. The weighted average of the malware detection is defined with the performance metrics. To measure the performance of the framework we have used Recall, Accuracy, Precision, Sensitivity, F-Measure, TP Rate and FP Rate. The accuracy derived from the percentage of the Malware detected from the data sets.

A. *Accuracy*

Accuracy is calculated as: Accuracy is one of the performance measures and it is simply a ratio of correctly predicted observation to the total observations. Accuracy defines the identification of the Malware which is classified as Malware from the listed data source.

Accuracy = TP+TN/TP+FP+FN+TN (2)

The accuracy is measured by the data available in the given malware data set to the malware content available in the data set. The metric parameter includes the Malware and No Malware value between the value of 0 to 1.

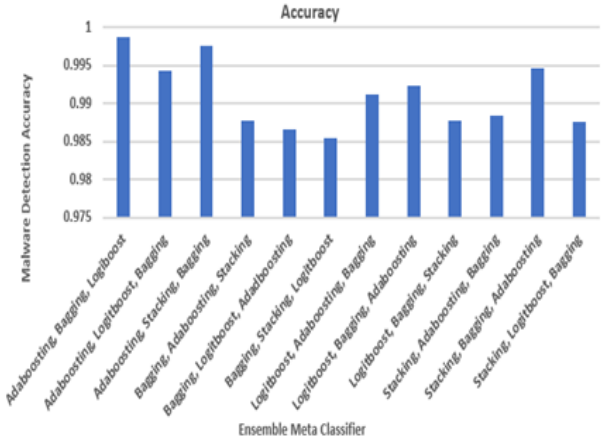

Fig. 9 Accuracy Metric

Table - I: Performance Metric Value for Accuracy

Ensemble Classifier	Accuracy
Adaboosting, Bagging, Logiboost	0.997554
Adaboosting, Logitboost, Bagging	0.976543
Adaboosting, Stacking, Bagging	0.972341
Bagging, Adaboosting, Stacking	0.974231
Bagging, Logitboost, Adadboosting	0.981234
Bagging, Stacking, Logitboost	0.975421
Logitboost, Adaboosting, Bagging	0.985643
Logitboost, Bagging, Adaboosting	0.966421
Logitboost, Bagging, Stacking	0.987651
Stacking, Adaboosting, Bagging	0.958452
Stacking, Bagging, Adaboosting	0.962312
Stacking, Logitboost, Bagging	0.972345

B. *Precision*

Precision when defined for a classifier with a given class is the ratio of true positives which should be combined to both true and false positives. True Positives (TP) is the correctly predicted Spam positive values of Malware in the data source which mean that the value of actual class is yes, and the value of predicted class is also yes. False Positives (FP) – When actual class is no, and predicted class is yes. The Fig 11 shares the experimental results after evaluating the performance of CECC Framework for Precision:

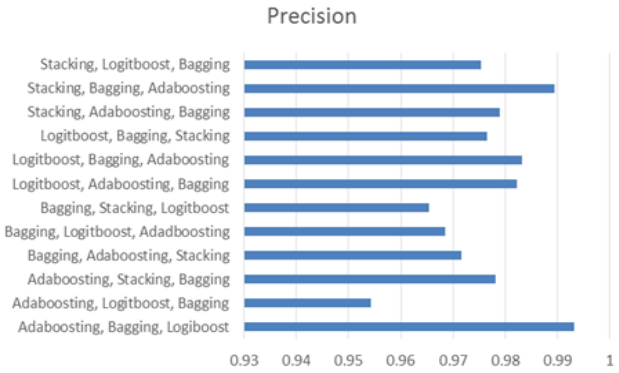

Fig. 10 Precision Metric

Table - II: Performance Metric Value for Precision

Ensemble Classifier	Precision
Adaboosting, Bagging, Logiboost	0.993306
Adaboosting, Logitboost, Bagging	0.954321
Adaboosting, Stacking, Bagging	0.978214
Bagging, Adaboosting, Stacking	0.971654
Bagging, Logitboost, Adadboosting	0.968532
Bagging, Stacking, Logitboost	0.965432
Logitboost, Adaboosting, Bagging	0.982314
Logitboost, Bagging, Adaboosting	0.983256
Logitboost, Bagging, Stacking	0.976543
Stacking, Adaboosting, Bagging	0.978923
Stacking, Bagging, Adaboosting	0.989425
Stacking, Logitboost, Bagging	0.975432

Recall

Recall is calculated with the ratio of True Positive Spam Positive words identified from the data set through BCC Clustering which shares the base data source for the LIME classification technique.

Recall = TP/TP+FN (3)

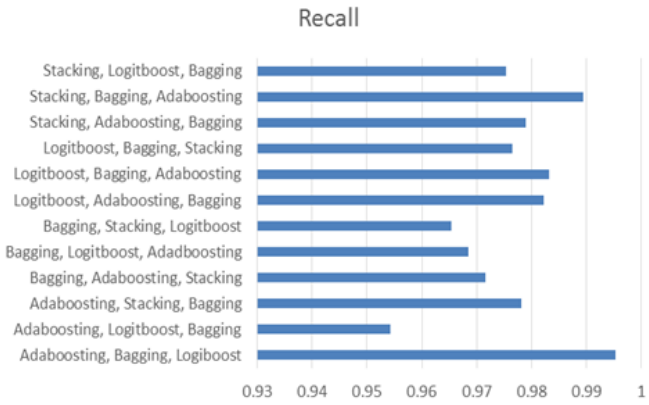

Table 3: Performance Metric Value for Recall

Ensemble Classifier	Recall
Adaboosting, Bagging, Logiboost	0.995314
Adaboosting, Logitboost, Bagging	0.954321
Adaboosting, Stacking, Bagging	0.978214
Bagging, Adaboosting, Stacking	0.971654
Bagging, Logitboost, Adadboosting	0.968532
Bagging, Stacking, Logitboost	0.965432
Logitboost, Adaboosting, Bagging	0.982314
Logitboost, Bagging, Adaboosting	0.983256
Logitboost, Bagging, Stacking	0.976543
Stacking, Adaboosting, Bagging	0.978923
Stacking, Bagging, Adaboosting	0.989425
Stacking, Logitboost, Bagging	0.975432

VI. DISCUSSION

The proposed CCEC framework works with the combination of clustering and classification to detect the malware from multiple online data sources. The clustering based on BCC is not dependent on single data source. We can use this method with multiple data sources with different format. Using this clustering method, the malwares are classified into 10 different clusters based on their family.

The accuracy and F Score obtained with the clustering method is high when compared to the Fuzzy c-means clustering algorithm, Hierarchical clustering algorithm Gaussian (EM) clustering algorithm, Quality threshold clustering algorithm, MST based clustering algorithm, kernel k-means clustering algorithm and Density based clustering algorithm [5]. The completion of clustering layer results with a trained data set which can be referred with the classification layer.

The classification layer provides more accurate results in malware detection with the fourth tier when compared with the other methods like statistical method, genetic programming, neural network and ANT colony [1]. The future of this research expands with the use of new combinations of ensemble meta classifiers with the complex data sources.

With the experiment we have identified the base classifier with multiple algorithms are tested. Random Forest provided high accuracy result in detecting the malwares. Adaboosting with the ensemble meta classifier provided better AUC of 0.998742 with the four tier LIME classifier. Adaboosting used with the fourth tier, bagging with the third tier and logitboost used in the second tier of the LIME classifier.

VII. CONCLUSION

We have introduced CCEC Framework as an effective method to detect malwares when compared with other approaches available today. Clustering was introduced using the Bayesian Consensus Clustering. Clustering layer remains important for this framework as the trained data set is derived from this layer. Using the trained data set the results improved with the malware detection with the reduced run time even with the complex data sets.

Using Random Forest as the base classifier and diverse ensemble meta classifier to form various iterative tiers using classification provided high results when the trained data set referred from the clustering layer. The AUC value with 0.9987 was obtained with the four tier LIME classification. Adaboosting used in the final tier, Bagging used in the third tier and Logitboost used in the second tier. The challenges we faced with the framework design for merging the clustering and classification as they are different learning methods. The framework formed is a semi supervised learning model. The implementations done prior used the offline data sources. But this framework we have used the online data sets with multiple data sources. The results are high using the clustering and classification with the CCEC framework when compared with the other available methods.

REFERENCES

1. "S. Balamurugan and M. Thangaraj, "Bayesian Consensus Clustering with LIME for Security in Big Data," in International Journal of Data Analysis Techniques and Strategies 2019, In Production, Publish by July 2019
2. "S. Balamurugan and M. Thangaraj, "Consensus Ensemble Clustering Algorithms for effective data analytics," in International Journal of computer engineering and applications 2017, pp 77 82
3. " L. Batten, J. Abawajy, and R. Dose, ``Prevention of information harvesting in a cloud services environment," in Proc. 1st Int. Conf. Cloud Comput. Services Science, 2011, pp. 66 72.
4. "E. Bauer and R. Kohavi, ``An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," Mach. Learn., vol. 36, nos. 1 2, pp. 105 139, 1999.
5. "G. Beliakov, J.Yearwood, and A.Kelarev, ``Application of rank correlation, clustering and classi cation in information security," J. Netw., vol. 7, no. 6, pp. 935 955, 2012.
6. L. Breiman, ``Bagging predictors," Mach. Learn., vol. 24, no. 2, pp. 123 140, 1996.
7. "L. Breiman, ``Random forests," Mach. Learn., vol. 45, no. 1, pp. 5 32, 2001.
8. " J. Chen et al., ``Big data challenge: A data management perspective," Frontiers Comput. Sci., vol. 7, pp. 157 164, Apr. 2013."
9. "J. Chen and Y. Yang, Effective and Ef cient Temporal Veri cation in Grid Work ow Enabling Timely Completion of Grid Work ow. Saarbrücken, Germany: Lambert Academic Publishing, 2011.
10. "M. Cimpoesu, D. Gavrilut, and A. Popescu, ``The proactivity of perceptron derived algorithms in malware detection," J. Comput. Virol., vol. 8, no. 4, pp. 133 140, 2012.
11. "H. Demirkan and D. Delen, ``Leveraging the capabilities of service-oriented decision support systems: Putting analytics and big data in cloud," Decision Support Syst., vol. 55, no. 1, pp. 412 421, 2013.
12. "W. Dou, Q. Chen, and J. Chen, ``A con dence-based ltering method for DDoS attack defense in cloud environment," Future Generat. Comput. Syst., vol. 29, no. 7, pp. 1838 1850, 2013.
13. "Y. Freund and R. E. Schapire, ``Experiments with a new boosting algorithm," in Proc. 13th Int. Conf. Mach. Learn., 1996, pp. 148 156.
14. "M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, ``The WEKA data mining software: An update," SIGKDD Explorations Newslett., vol. 11, no. 1, pp. 10 18, 2009.G. O. Young, "Synthetic structure of industrial plastics (Book style with paper title and editor)," in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.

AUTHORS PROFILE

Balamurugan S received his Post Graduate Degree M.Sc (S.W) Engg degree in Software from Madurai Kamaraj University in 2006. He works with IBM India Pvt Ltd as an Architect in the field of Cloud security and Cognitive. He is currently a research scholar working in the areas of increasing the security with the Big Data and identification of the malware impact in the Cloud. Area of research includes the Security with the Big Data, Cyber Security, Cloud Applications, Clustering Methods and Classifications.

Thangaraj M received his post-graduate degree in Computer Science from Alagappa University, Karaikudi, M.Tech. Degree in Computer Science from Pondicherry University and Ph.D. degree in Computer Science from Madurai Kamaraj University, Madurai. Around 30 years of Work Experience and working as Prof. and Head of the Department of Computer Science, Madurai Kamaraj University. Area of research is Big Data Analytics, Social Network Analytics and Wireless Sensor Analytics. Handled more than three research projects implementation for Government organizations like NTRO and UGC. Has published research work in more than 100 plus international journals. Senior member of International Association of Computer Science and Information Technology, China and Member of The Society of Digital Information and Wireless Communications, USA.