Detection of Malicious uniform Resource Locator

P. Varaprasada Rao, S. Govinda Rao, P. Chandrasekhar Reddy, B. S. Anil Kumar, G. Anil Kumar

Abstract: ***With the growing use of internet across the world ,the threats posed by it are numerous. The information you get and share across the internet is accessible, can be tracked and modified. Malicious websites play a pivotal role in effecting your system. These websites reach users through emails, text messages, pop ups or devious advertisements. The outcome of these websites or Uniform Resource Locators (URLs) would often be a downloaded malware, spyware, ransomware and compromised accounts. A malicious website or URL requires action on the users side, however in the case of drive by only downloads, the website will attempt to install software on the computer without asking users permission first. We put forward a model to forecast a URL is malicious or benign, based on the application layer and network characteristics. Machine learning algorithms for classification are used to develop a classifier using the targeted dataset. The targeted dataset is divided into training and validation sets. These sets are used to train and validate the classifier model. The hyper parameters are tuned to refine the model and generate better results.***

Keywords: Malicious URLs or Websites, Malware, Spyware, Ransomware, compromised accounts, Drive by only downloads, Application layer, Network characteristics, Machine learning, Classification, Classifier, Hyperparameters.

I. INTRODUCTION

The emergence of new communication technologies has a huge impact in the growth of businesses over many applications including online-banking, e-commerce and social networking. Indeed, in today's world it is important for a person to be online for having a prosperous career. It is evident that the significance of internet is growing continuously. However, the advancement in technologies have come with threats (cybercrime) that put its users into a unpleasant situation. Such threats include rogue security software, adware, spyware, phishing, DOS and DDOS. These threats always lead to exploitation of user's information and wealth.

Revised Manuscript Received on July 04 2019.

Dr P Varaprasada Rao: Professor in CSE, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET). Hyderabad,India.

Dr S Govinda Rao: Professor in CSE, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET). Hyderabad,India.

Dr P Chandrasekhar Reddy: Professor in CSE, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET). Hyderabad,India.

B.S.Anil Kumar: Assistant Professor in CSE, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET). Hyderabad,India.

G.Anil Kumar: Assistant Professor in CSE, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET). Hyderabad,India

A system that is not protected using appropriate security controls can be infected with malicious software anyone can breach into your system and access you're information. Malicious URLs and infected websites are the root for such threats. A malicious URL is a compromised URL which often leads to downloaded malware, spyware, ransom ware and compromised accounts. Uniform Resource Locator (URL) is a unique identifier used to locate a resource over the internet as is defined as the global address of documents and other resources on the world wide web. A URL typically contains two parts protocol name and domain name or IP. Malicious URLs account to one third of total URLs that exist. Naive users of internet are the primary targets of such sites. Most frequent attacks tend to be drive by download, phishing and social engineering and spam. Drive by download is a harmful software code that is installed on a user's system without the user's permission. Phishing and social engineering attacks use compromised websites seek personal information by acting as a trustworthy organization. Spam can be a dangerous source of threat, and is a part of phishing scam. To overcome this many researchers have worked to put forth various designs that are effective solutions for detecting such URLs. The most common method employed by antivirus groups is blacklisting method. It uses a database where all malicious URLs available and identified are stored. This database is updated frequently when a new malicious URL is found. This approach of finding a URL is fast as it has a simple query overhead. Moreover, such techniques would have a low false rate (it suffers from non-trivial false-positive rates). However, it is difficult to maintain a comprehensive list of blacklisted URLs. Attackers are creative enough to overcome the blacklisted URLs and generate URLs. which appear to be legitimate. The some of the most common ways to detect a URL is compromised or not is using IP address, long or short URLs, sub and multi domains, domain registration, nonstandard ports, pop-up windows, age of domain. These attackers use an algorithmic approach to generate URLs which make them look legitimate and are not blacklisted out. Thus, blacklisting methods have acute limitations, and it is insignificant to overcome them, as blacklists are useless whenever a new URL is generated. To overcome these drawbacks, researchers have considered machine learning as a better approach. Machine learning is used to predict when the patterns are hidden within the dataset to make meaningful predictions. The most common method for pattern discovery is pattern classification. The advantage over blacklisting is that we can predict a URL which is not in database. To develop a model using machine learning the primary requirement is targeted training dataset. The targeted dataset for malicious URL dataset would be large number of URLs with their features. We use

Honeypot which is a network attached system used to detect, deflect or study hacking attempts. Machine learning is classified into supervised, unsupervised, semi-supervised and reinforced learning on the basis of availability of training data.

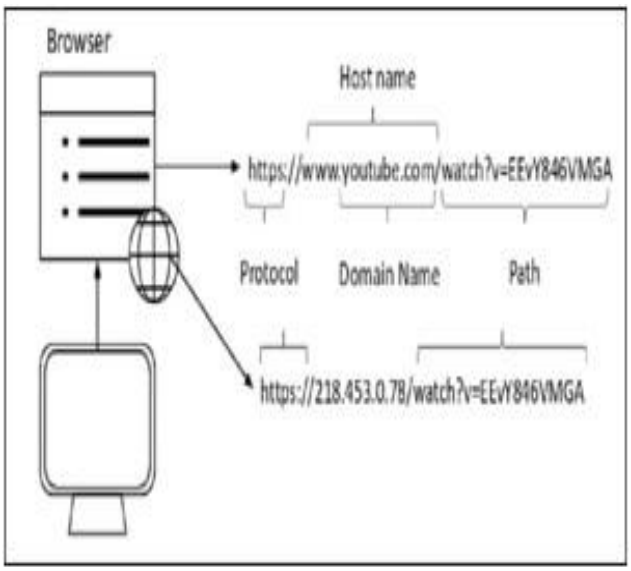

Figure1: Kaggle repository of malicious and benign URLs

The dataset is collected from Kaggle repository of malicious and benign URLs. After the dataset is collected, it processed to remove NULL values and interpolate the data. The processing is done based the attributes collected. In a dataset all the attributes are do not play a decisive role in designing the model. After identifying those attributes and pre-processing of data is done an efficient machine learning algorithm is considered for developing the model. Here our primary focus is on binary classification. In classification we group data into a category and then predict the output. Binary classification is the most Pernicious URL recognition is two-crease process that incorporates:

1) **Feature Representation**: Infer the reasonable element portrayal: $u \rightarrow x$ where $x \in R^d$ is d-dimensional component vector speaking to URL.
2) **Machine Learning:** Learning an expectation work $f : R^d \rightarrow R$ which predicts the class task for any URL occurrence x utilizing fitting element portrayals.

We think about paired order. The fundamental target of AI for vindictive URL identification is to boost the prescient exactness. Both the component portrayal and AI are imperative to accomplish this goal. While the initial segment of highlight portrayal depends on area learning and application layer, the later spotlights on preparing the order display by means of an information driven enhancement approach. Fig.2 outlines the design of comprehending Detection of Malicious URL utilizing AI. The initial step is to change over a URL u into highlight vector x, where a few sorts of data can be viewed as, for example, application layer and layer attributes. Utilizing space learning and related capability, an element portrayal is developed by social affair all pertinent data about a URL. These highlights extend from lexical data (length of URL, the words utilized in the URL, and so on.) to have based data (WHOIS information, IP address, and so forth.). Subsequent to social occasion data, it is prepared to store in an element vector x. In view of the sort of data utilized, $x \in R^d$ created from a URL is a d-dimensional vector. A one of a kind test that influences this issue explanation is the quantity of highlights that may not be fixed or known ahead of time. It is henceforth a provoking assignment to structure a decent component portrayal that is vigorous to unrevealed information. In the wake of getting the component vector x for the preparation information, to get familiar with the forecast. Streamlining issue to such an extent that the exactness is boosted. The capacity f for the most part parameterized by a

d−dimensional weight vector w,

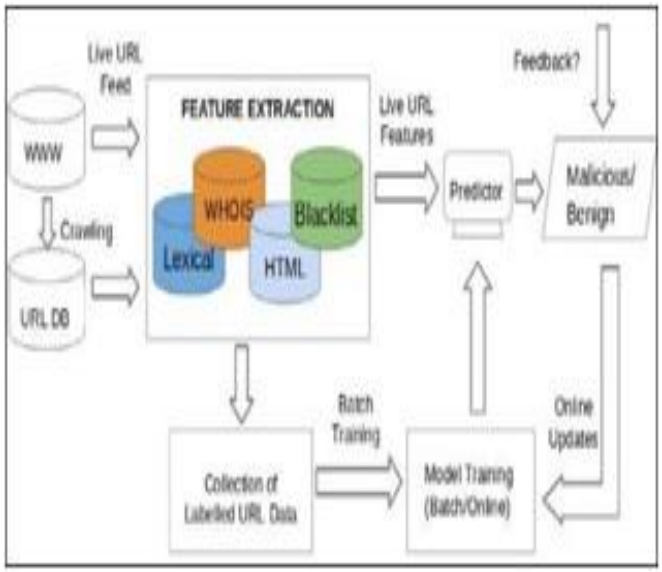

Figure2: Architecture for Malicious URL detection

basic way of classification where we classify things as a positive and a negative category. For detection of malicious URL we classify URLs into malicious or benign URLs. This way it would predict whether a URL is malicious or not (positive or negative). This research is aimed at developing a machine learning model which classifies the URLs. We implement this using Extreme Gradient Boosting (XGBoost) classifier which is an ensemble learner. Calculate the contributions made by the attributes and the accuracy gained by each one of them. Also find the future enhancements to be made for better results. We focus on the modules that are to be implemented and algorithm used to build the model. We also focus on the features that help the model to be more accurate.

II. STATEMENT OF THE PROBLEM

The issue of dangers URL discovery is a parallel investigation for forecast: "malignant" and "kindhearted". For a given dataset T URLs { (u_1,y_1), .. , (u_t,y_t) }, where u_t for t relating mark which speaks to malignant or kindhearted

Figure3: Features of web sources URLs

function f : $R^d \rightarrow R$, it is often formulated as defined

$f(x) = (w^t , x)$. Let $\hat{y}_t = \text{sign}(f(x_t))$ indicate the class name expectation made by the capacity. The quantity of mistakes made by the Prediction on the whole preparing information is given by. Since the marker work isn't curved, the streamlining can be hard to comprehend. made by the attributes and the accuracy gained by each one of them. Also find the future enhancements to be made for better results. We focus on the modules that are to be implemented and algorithm used to build the model.

Feature selection and its generator is done by honeypot and PyShark. Each URL was executed on honeypot for 4 seconds with its default configuration and in parallel to the process a script captured the web traffic through PyShark. Another tool was developed in Python for processing HTTP content and WHOIS properties.

With the above, the following characteristics were obtained:

- URL: The unknown distinguishing proof of the URL investigated in the examination.
- URL_LENGTH: The quantity of characters in a given URL.
- NUMBER_SPECIAL_CHARACTERS: The quantity of unique characters distinguished in the URL.
  Unique characters set {“/”, “%”, “#”, “&”, “. “, “=”}.
- CHARSET: It is a categorical value and is the operative system of the server got from the packet response.
- SERVER: It is an all out esteem and is the employable arrangement of the server got from the bundle reaction.
- CONTENT_LENGTH: Represents the substance size of the HTTP header.
- WHOIS_COUNTRY: It is a clear cut variable and are the nations got from the server reaction (explicitly, our content utilized the API of WHOIS).
- WHOIS_STATEPRO: It is a clear cut variable and are the states got from the server reaction (explicitly, our content utilized the API of WHOIS).
- WHOIS_REGDATE: Provides the server enrollment date and the variable has date. Qualities with organization DD/MM/YYYY HH:MM.
- WHOIS_UPDATED_DATE: The last update date from the server examined.
- TCP_CONVERSATION_EXCHANGE:The number of TCP parcels traded between the server and honey pot customer.
- DIST_REMOTE_TCP_PORT:The quantity of the ports identified and particular to TCP.
- REMOTE_IPS: this variable has the all out number of IPs associated with the honeypot
- APP_BYTES: The quantity of bytes exchanged.
- SOURCE_APP_PACKETS: The bundles sent from the honeypot to the server.
- REMOTE_APP_PACKETS: The bundles got from the server.
- APP_PACKETS: The complete number of IP bundles created amid the correspondence between the honeypot and the server.
- DNS_QUERY_TIMES: The quantity of DNS bundles created amid the correspondence between the honeypot and the server.
- TYPE: It is an all out factor and the qualities speak to the sort of page examined, 1 if vindictive sites and 0 if considerate sites.

DOMAIN_NAME	WHOIS_COUNTRY--IN	WHOIS_COUNTRY--IT	WHOIS_COUNTRY--JP	WHOIS_COUNTRY--KR
wikia.com	0	0	0	0
healthgrades.com	0	0	0	0
kfdm.com	0	0	0	0
labradorwest.com	0	0	0	0
kusports.com	1	0	0	0
amazon.com	0	0	0	0
usedregina.com	0	0	0	1
nj.com	0	0	0	0
waatp.com	0	0	0	1
healthgrades.com	0	0	0	0
greenwood-centre-hudson.org	0	0	0	0

Figure :One hot encoding on WHOIS_COUNTRY attribute

Strategy

We define the problem statement and articulate it. Once the problem is formed we consider various perceptions. And also validate that the data is true and good enough to develop the model. Even though the data is not preprocessed one can still consider the data. Once we have all set choose the algorithm that best fits our requirements.

Dataset preprocessing

It's important to have data related to our requirements and the predictions we make using that data. After having the data together visualization of it is important. And now we decide whether it can be implemented using supervised or unsupervised learning.Data preprocessing is the most important step of developing a model. It helps to shape the data for efficient predictions. This process includes data formatting, cleaning and sampling. We eliminate NULL values and interpolate the data. One hot encoding is used to convert categorical values into form that a machine understands. Due to this process the machine understands easily and develops a model that gives more accurate results. It generally converts the data into zero's and one's. This ensures that the machine can find patterns efficiently.

Dataset splitting:

When the data is ready it is important that we train our model on some part of data and predict on the other part of it. This checks the accuracy of a model and doesn't allow the model to overfit the dataset. We split the data into training set and validation set. The more training data is used, the better the potential model performs. Consequently, more results from model testing data leads to better model performance and generalization capability.

Modeling

Supervised learning is carried out for identifying a URL is malicious or not. Supervised methods attempt to discover the relationship between input attributes and a target attribute. The relationship discovered is represented in a structure referred to as a model. It can be done using classification and regression.After selecting the algorithm model evaluation and testing is done. Cross validation is done for tuning parameters. We use boosting method to improve predictions. XGBoost is an ensemble learner which is used in our project to detect malicious URLs.

Algorithm

XG Boost method is an implementation of gradient boosted decision trees, developed for fast and performance. Trees are constructed iteratively till the error is minimized and continuous process of building models makes XGBoost classifier an ensemble learner.Key features of this algorithm are scalability, robust to outliers, can handle mixed predictors. It's advantages are good bias-variance and computation speed. It provides gradient boosting framework for C++, Java, Python, R and Julia.

Tree Building Algorithm

1)Grow the tree to the maximum depth

a)Find the best splitting point

b)Assign weight to the leaf nodes

2)Prune the tree to delete nodes with Negative gain

To understand one has to know the basic structure of the model behind it. Suppose we have **K** trees, then model generated is

$$\sum_{k=1}^{K} f_k$$

Where each f_k is the prediction from each decision tree. The model is built on collection of decision trees. After generating decision trees, we make prediction.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i)$$

Where (x_i) is the feature vector of the i^{th} data point. Similarly, the prediction at t^{th} step can be defined as

$$\hat{y}_i(t) = \sum_{k=1} f_k(x_i)$$

To train the model the loss function should be optimized. We use Log Loss for binary classification.

$$L = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(p_i) + (1 - y_i) \log(1 - p_i))$$

Regularization is a crucial part of a model.A good regularization controls the complexity and over fitting of a model.

$$\Omega = \gamma T + (1/2)\lambda \sum_{j=1}^{T} w2j$$

Where T is the number of leaves and w2 is score jth on the leaf.

Obj = L + Ω

Where loss function controls the predictive power, and regularization controls the simplicity of the model. Given an objective Obj(y,ŷ) to optimize, XGBoost is an iterative technique which calculates

$\partial \hat{y}\, Obj(y, \hat{y})$

At each iteration, we improve ŷ along the direction of the gradient to minimize the objective. We calculate second order Taylor approximation for it and we get

$$Obj^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + 1/2 h_i f_t^2(x_i) \right] + \Omega(f_t)$$

This is the objective of i th step . Our goal is to find a f^t to optimize it.

III. RESULTS

A confusion matrix is a table used to describe the performance of a classification model on test set for which true values are known.It consists of the following values:

- True Positive(TP): Observed value is positive, and is predicted to be positive.
- False Negative(FP): Observed value is positive, but is predicted negative.
- TrueNegative(TN): Observed value is negative, and is predicted to be negative.
- False Positive(FP): Observed value is negative, but is predicted positive .

Table1: Classification Rate or Accuracy is given by the relation

	Predicted: No	Predicted: Yes
Actual: No	549	9
Actual: Yes	11	55

Classification Rate or Accuracy is given by the relation:

Accuracy =TP+TN/TP+TN+FP+FN

Accuray = 0.968

Recall is defined as:

Recall = TP/TP+FN

Recall = 0.8333

High Recall indicates the class is correctly recognized.

Precision is defined as:

Precision=TP/TP+FP

Precision = 0.8593

F-measure uses Harmonic Mean in place of Arithmetic Mean as it punishes the extreme values more.

F − measure =2*Recall*P recision /Recall+Precision

F-measure = 0.8461

To visualize classification we use AUC(Area Under The Curve) ROC(Receiver Operating Characteristics) curve.

ROC curve is probability curve and AUC curve represents degree or measure of separability.

The graph below represents the AUC-ROC curve for the classifier designed in this project

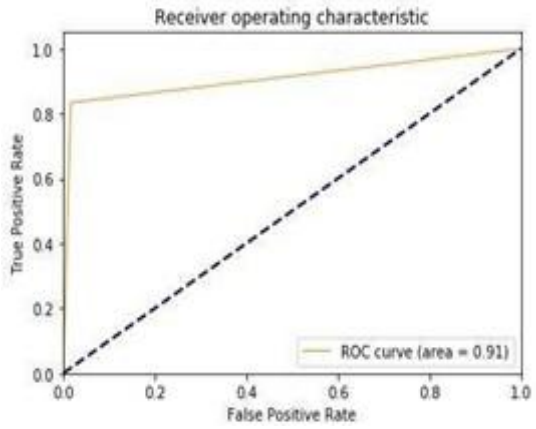

Figure: Receiver operating characteristic curve

Feature importance gives you a score for each feature of your data, higher scores are having more importance or relevance of feature towards your output variable. The importance of features for the dataset is:

NUMBER_SPECIAL_CHARACTERS 0.15879828
URL_LENGTH 0.13304721
REMOTE_APP_PACKETS 0.09656652
SERVER 0.08798283
DIST_REMOTE_TCP_PORT 0.072961375
CHARSET 0.06866953
WHOIS_COUNTRY 0.057939917
WHOIS_STATEPRO 0.05364807
DNS_QUERY_TIMES 0.051502146
CONTENT_LENGTH 0.047210302
SOURCE_APP_BYTES 0.03862661
REMOTE_IPS 0.03218884
APP_BYTES 0.025751073
WHOIS_REGDATE 0.025751073
REMOTE_APP_BYTES 0.023605151
WHOIS_UPDATED_DATE 0.021459227
TCP_CONVERSATION_EXCHANGE 0.0021459227
SOURCE_APP_PACKETS 0.0021459227
APP_PACKETS 0.0

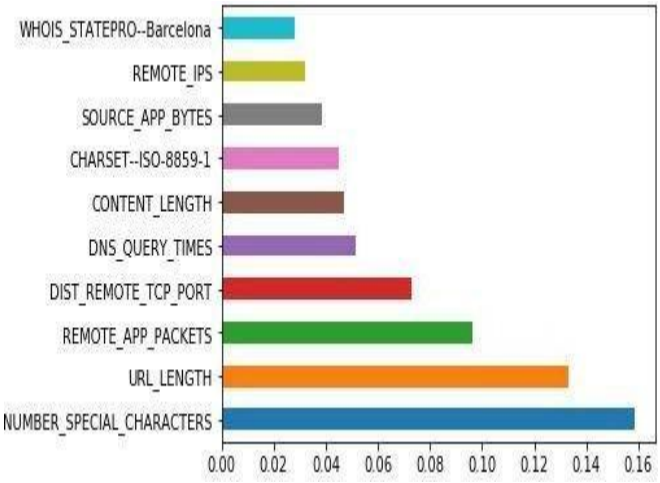

Figure:The feature importance of dataset after one hot encoding.

IV. CONCLUSION

The Threats of malware infections through malicious URLs are increasing today, so actions are needed to prevent them. As a countermeasure against malicious URLs, blacklisting URLs or domains are one among them. But techniques don't predict if a URL is not in the database. This also requires a huge amount of database for storing all blacklisted URLs. So, to overcome this machine learning is an efficient way to predict a URL as malicious or benign. Best way to do this is using classification algorithms. We classify URLs as malicious or benign, this kind of classification is known as binary classification. This can be implemented using decision trees, random forest algorithm, bayes theorem and any other classification algorithm. The best way to implement is using XGBoost algorithm which is a ensemble learner. This helps us to reduce the loss and regularize the the decision tree constructed at each step of the algorithm. One key feature of this project is that we use one hot encoding to change categorical values to machine understandable values. This enhances the speed of building the model and predicting the it. Although the prediction are done blacklisting is still used as it is faster when compared to any machine learning algorithm and is more accurate when a particular URL is within the given dataset. Machine learning techniques are increasing and their impact on different fields of engineering is also more. One way to improve the rate of prediction and accuracy is to have more features and consider the features with highest priority and building the classifier again. This can be done using feature importance. Visual elements classification by building a vast dataset of malicious websites to cross-check against, future detection and classification can be improved by analyzing the visual elements which are shared among such websites. This could be extended to DOM analysis.

REFERENCES

1. *P. Prakash, M. Kumar, R. R. Kompella, M. Gupta, "Phishnet: predictive blacklisting to detect phishing attacks", INFOCOM 2010 Proceedings IEEE., pp. 1-5, 2010.*
2. *P. de las Cuevas, Z. Chelly, A. Mora, J. Merelo, A. Esparcia-Alcazar, "An improved decision system for URL accesses based on a rough feature selection technique" in Recent Advances in Computational Intelligence in Defense and Security, Springer, pp. 139-167, 2016.*
3. *A. Mora, P. De las Cuevas, J. Merelo, "Going a step beyond the black and white lists for URL accesses in the enterprise by means of categorical classifiers", ECTA, pp. 125-134, 2014.*
4. *M.-Y. Kan, H. O. N. Thi, "Fast webpage classification using url features", Proceedings of the 14th ACM International Conference on Information and knowledge management, pp. 325-326, 2005.*
5. *E. Baykan, M. Henzinger, L. Marian, I. Weber, "Purely URL-based topic classification", Proceedings of the 18th International Conference on World wide web., pp. 1109-1110, 2009.*
6. *J. Ma, L. K. Saul, S. Savage, G. M. Voelker, "Beyond blacklists: learning to detect malicious web sites from suspicious urls", Proceedings of the 15th ACM SIGKDD International Conference on Knowledge discovery and data mining., pp. 1245-1254, 2009.*
7. *"Learning to detect malicious URLs", ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, no. 3, pp. 30, 2011.*
8. *P. Zhao, S. C. Hoi, "Cost-sensitive online active learning with application to malicious URL detection", Proceedings of the 19th ACM SIGKDD International Conference on Knowledge discovery and data mining., pp. 919-927, 2013.*
9. *Y. Sun, A. K. Wong, M. S. Kamel, "Classification of imbalanced data: a review", International Journal of Pattern Recognition and Artificial Intelligence, vol. 23, no. 04, pp. 687-719, 2009.*
10. *V. López, A. Fernández, S. García, V. Palade, F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics", Information Sciences, vol. 250, pp. 113-141, 2013.*
11. *Rao, Govinda, Varaprasada Rao, and R. Rambabu. "A Novel*
12. *Approach in Clustering Algorithm to Evaluate the*
13. *Performance of Regression Analysis." 2018 IEEE 8th*
14. *International Advance Computing Conference (IACC). IEEE,*

AUTHORS PROFILE

Dr P Varaprasada Rao is working as Professor in CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET), Hyderabad. Completed his PhD From JNTUK and M.Tech From Andhra University. He is around 14 years of teaching experience..He is life member of MIE.

Dr S Govinda Rao is working as Professor in CSE, Gokaraju Rangaraju Institute of Engineering and Technology (GRIET), Hyderabad. Completed his PhD From JNTUK and M.Tech From Andhra University. He is around 14 years of teaching experience. He is life member of MIE.

Dr. P. Chandra Sekhar Reddy completed his B.Tech from Sri Krishna Devaraya University. He received M.Tech from JNTUH .He recieved Ph.D. Degree from JNTU Anantapur. He is currently working as Professor in GRIET, Hyderabad.. He is the member of professional bodies like IEEE, IAENG, CSI and CSTA.

B S Anil Kumar Working as Assistant Professor in CSE,Gokaraju Rangaraju Institute of Engineering and Technology(GRIET). Completed his M.Tech From Sathyabama University,chennai. His area of Interest Computer Networks and Datamining..
.

G Anil Kumar Working as Assistant Professor in CSE,Gokaraju Rangaraju Institute of Engineering and Technology(GRIET). Completed his M.Tech From JNTUH University,Hyderabad. His area of Interest Computer Networks and Network Security.