Journal of Analysis and Computation (JAC)
(An International Peer Reviewed Journal), www.ijaconline.com, ISSN 0973-2861
Volume XVII, Issue II, July-Dec 2023

A STUDY ON VOTING SYSTEM THROUGH FACE RECOGNITION

Rohit Chhabra[1], Mohit Gupta[2]. Hemant Singh Rathor[3]

[1]Assistant Professor, Jaipur Engineering College and Research Centre, Jaipur, India
[2]Jaipur Engineering College and Research Centre, Jaipur, India
[3]Jaipur Engineering College and Research Centre, Jaipur, India

ABSTRACT

The principal objective of this framework is to give a web based casting a ballot framework that will support lessening misrepresentation in manual democratic frameworks and prior cycles of internet casting a ballot that involved a webcam for face acknowledgment and OTP age. For electors who can't make a trip to the democratic spot, we are likewise presenting an area free democratic strategy (old neighborhood). Here, we furnish a framework with many degrees of confirmation, including face check and OTP check with approval information, to ensure the gadget's constancy. Every citizen can get to the framework subsequent to being recognized and checked against the gave information base of enrolled electors. The elector will actually want to continue on with choosing their picked competitor from the board after the coordinating face is coordinated with the information given.

Keywords – Image Processing, Python, Voting System, Face Recognition, MySQL, OTP

[1] INTRODUCTION

As per TOI information, 11 lakhs of false votes were found in Delhi on January 24, 2009. Then, at that point, in June 2013, 30000 ill-conceived citizens were found in the Sheila Dikshit electorate by the electing commission, as per India News. One more news which was claimed by LJP(LokJanshakti Party) Boss, Smash Vilas Paswan saying that Bihar political race were having 30% phony elector cards. Political race includes both public or confidential vote which relies upon the position. Probably the most pivotal positions are held by nearby, state, and central legislatures. Citizens cast their voting forms in paper-based races by setting them in fixed boxes set all through the constituent circuits of a specific country. Subsequent to finishing of political race period the cases which contains of polling form control unit are opened and casts a ballot are included physically in presence of the ensured authorities named by political decision commision. Casting a ballot is in this way a tedious activity that likewise utilizes a great deal of assets. Utilizing facial acknowledgment and OTP, we have proposed a web based casting a ballot framework in this review. The server unit gets the data on the OTP and Face for additional confirmation. The server then checks the information base for information and analyzes it to information that as of now exists there. The individual is allowed to survey the vote assuming the information matches the data that has recently been saved. On the off chance that not, a message is displayed on the screen, hence the client should do as such. In the event that not, a message is shown on the screen and subsequently the individual isn't permitted to survey the vote. For casting a ballot delegates are named by electorates. In current situation citizen necessities to show his/her elector ID card to make the choice on the corner. So this interaction is tedious as the elector ID card should be get checked by the authorities. In this manner to accelerate the democratic cycle and keep away from such kind of issues, we have proposed the new framework.

1.1 Problem Statement

Despite the fact that our nation has gained ground toward digitalizing India, the democratic cycle actually has huge blemishes. With the ongoing framework, casting a ballot must be enlisted in the event that individuals go to the surveys. At the hour of casting a ballot, citizens' names are recorded on the rundown for their particular region. They can't project a polling form farther than the democratic card's recorded location. Subsequently, citizens who have migrated to different areas can't do so actually. The weakness of this framework is shown by the ongoing Covid pandemic. Because of the prerequisite that the citizen be there face to face to project their polling form, this could bring about a disappointment of social distance all through the democratic interaction.

1.2 Literature Survey

1) Decentralized E-Voting Portal Using Blockchain

This paper addresses structures of blockchain for the E-casting a ballot framework. This execution can be utilized for limited scope decisions, for example, board rooms or inside corporate houses races. Shrewd agreement from Ethereum is utilized for this execution. The thought behind this execution is to join the innovation of blockchain with the homomorphic encryption and mystery sharing plans for the decentralized democratic applications protected from confided in outsider. It gives the general population and straightforwardness casting a ballot cycle which safeguards the namelessness of elector's personality and the security of information transmission and confirmation of voting forms during charging stage.

Advantages:

It builds straightforwardness of the democratic and safeguards the vulnerability of personality of citizen. Security to the information protection, transmission and voting forms check during the period of charging is given.

Disadvantages: Web and blockchain-based casting a ballot frameworks can have security chances.

Limitations: Client ought to know about application.

2) Electronic Voting Machine with Enhanced Security:

This article frames the turn of events and plan of a democratic framework using the ATMEGA 32 microcontroller, which incorporates three extra degrees of insurance. For the most common way of casting a ballot with paper polling forms, EVM takes a ton of time. Along these lines, to be particularly speedy and reliable, labor supply should be saved. Accordingly, without the utilization of polling form paper, casting a ballot secrecy is safeguarded in this example of framework execution. Casting a ballot machines that utilize VVPAT are at present more exorbitant than EVMs. The EVM gives 100% proof of altering, and discoveries are effectively open. Be that as it may, this EVMs are effectively modifiable by changing the equipment associations. Therefore, this article recommends adding three layers of safety.

Advantages: Speed of counting of polling forms is expanded utilizing this application . Saves the expense of paying staff as there is compelling reason need to physically count votes.

Disadvantages: Security risk present.

Limitations: Issue of Compatibility can occur.

3) Biometrically Secured Electronic Voting Machine

In this paper, Arduino and Finger impression scanner is utilized to carry out the framework which recognizes every elector, likewise count casts a ballot and dodges counterfeit votes. In this framework elector is distinguished utilizing FPS which identifies in the event that an individual is an enrolled or not and furthermore it denies for the citizen to make the subsequent option.

Advantages:

Biometric description of voter is used.

Disadvantages:

High level security framework can be expected for meaning of speculations and expenses to execute.

Limitations: Application should be known to users.

4) Multipurpose platform independent online voting system:

The elector simply requires an Aadhar card number and a cell phone that can filter the framework's standardized identification to utilize this technique. Considering that the program is totally online-based, the client can decide on any spot. This framework creates its own polling form for casting a ballot. Vote information is scrambled at the client end, and it is decoded at the nearby manager end. Accordingly, the democratic system is more verified and secure.

Advantages:

User just needs to have Aadhaar card number.

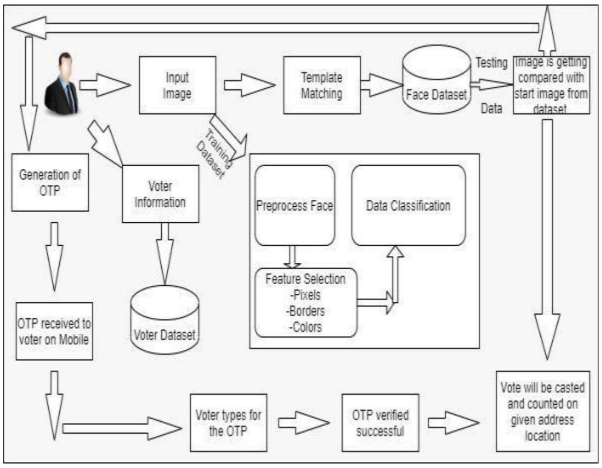
Disadvantages:

Risk of fraud can happen.

Limitations:

Smartphone is required for this system.

[2] PROPOSED SYSTEM



Description:

To use the framework, a client should initially enroll by entering data, for example, their Aadhar number, versatile number, city, age, and secret phrase. Citizen dataset contains this data. While enlisting, the framework utilizes a camera to catch the client's feedback picture. For layout coordinating, this picture is kept in the face dataset. The client should then sign into the framework with their Aadhar number and secret key to make their choice. After then, at that point, the client should answer a security question. In the event that it gets checked effectively the client continues on toward the following page where he/she can choose the contender to make the choice. The webcam turns on and confirms the client's face utilizing the gave dataset once the client taps the vote button. After the face has been effectively confirmed, an OTP will be shipped off the client's enlisted cellphone number. Casting a ballot is effective in the event that the OTP is approved.

PROPOSED WORK:

2.1 Modules

Voter(User): Here Elector is the notable individual to cast a ballot the specific competitor. The elector is checked client approved by administrator on enrollment process.

ML Process: AI process is for preparing the elector appearances to acknowledgment casting a ballot time to cast a ballot the up-and-comer.

Face and OTP Verification: The proposed design makes sense of that here 2 different ways confirmation process on casting a ballot time; one is face acknowledgment and second is OTP check.

2.2 Algorithm used

1. Local Binary Pattern Histogram

A detectable descriptor style utilized for PC vision characterization is called nearby double examples (LBP). LBP was made into a unique case from the 1990 Surface Range model. Whenever LBP first was addressed was in the year 1994. Therefore, it has been used as a surface for recognizing strong parts. Consolidating LBP with the descriptor histogram of coordinated angles works on the execution of ID on unambiguous datasets (Hoard).The flowchart for the LBPH algorithm is shown in Figure 2.

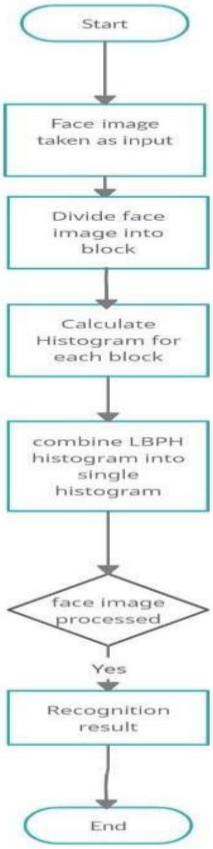

Start
Face image taken as input
Divide face image into block
Calculate Histogram for each block
combine LBPH histogram into single histogram
face image processed
Yes
Recognition result
End

Fig -2: LBPH Flowchart

To encode highlights, the information picture is apportioned into cells (4 × 4) of pixels. Via conveying the encompassing pixel esteems either clockwise or anticlockwise, the difference is accomplished. Each neighbor's power esteem is contrasted with the worth of the central pixel. The area has been doled out a 1 or 0, contingent upon whether the thing that matters is higher or lower than 0, and the outcomes are made as a 8-bit number in a solitary cell.

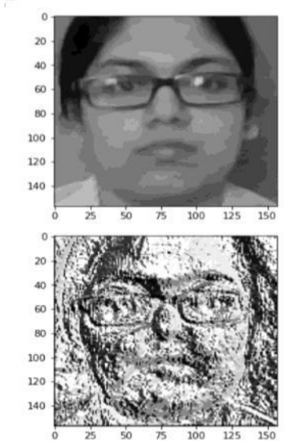


Fig -3: LBPH for Face Recognition

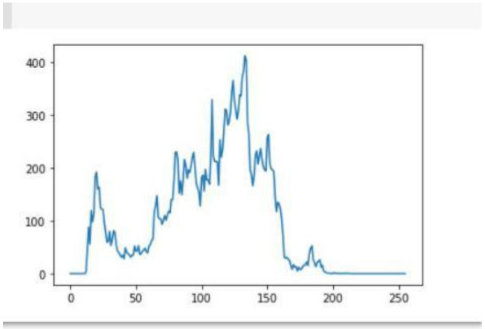


Fig -4: Histogram of Face By LBPH

2. Haar Classifier Algorithm:

The primary structure block for Haar classifier object acknowledgment is a bunch of Haar-like qualities. It changes the difference values between neighboring rectangular groupings of pixels as opposed to the pixel's power values. The difference fluctuations between the pixel groupings are utilized to appraise the overall brilliant and dim districts. A few adjoining bunches with relative difference change consolidate to produce a Haar-like element. By just raising or bringing down the size of the pixel bunch, it is easy proportional the Haar qualities, permitting them to be applied to objects of various sizes. With subimage investigation, which empowers the outpouring of classifiers, the most extreme probability of dissecting the Haar-highlights that separate a thing is accomplished. It allows a classifier's precision to change only a single time. By utilizing 200 fundamental qualities, the framework created by Viola and Jones has a 95% exactness rate for recognizing human countenances.

The initial step is to prepare the Haar classifier fountains to perceive human face attributes such the lips, eyes, and nose. Concerning the preparation of the classifier, the AdaBoost strategy should be utilized related to the Haar highlight procedure. In any case, Intel has made an open source tool compartment known as the Open PC Vision Library that simplifies it to plan PC vision-related applications (OpenCV).

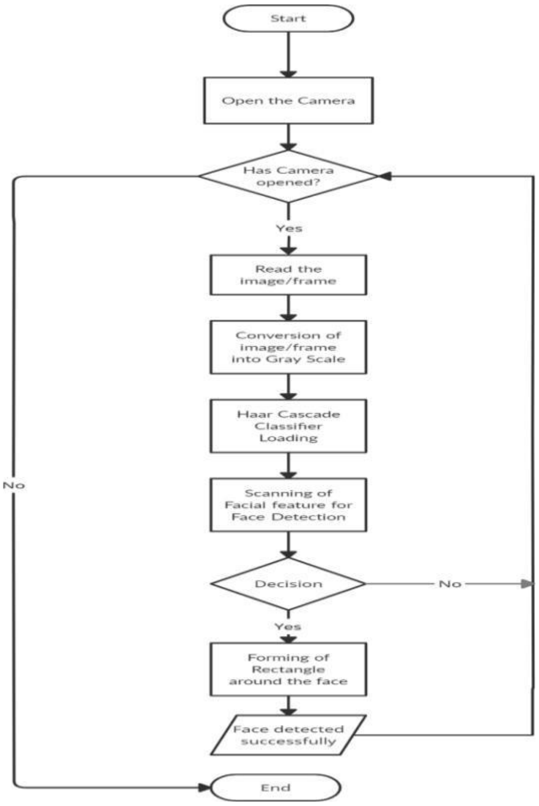
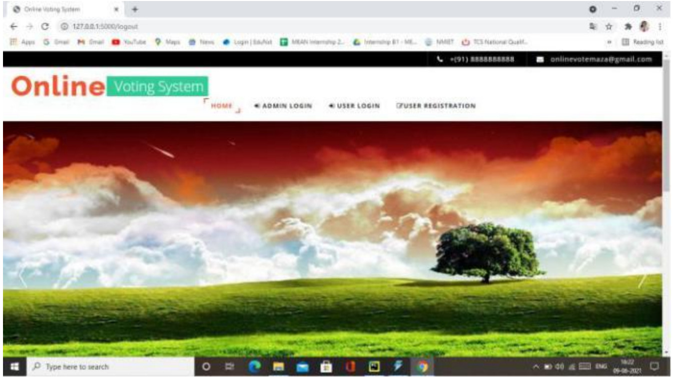

Start
Open the Camera
Has Camera opened?
Yes
No
Read the image/frame
Conversion of image/frame into Gray Scale
Haar Cascade Classifier Loading
Scanning of Facial feature for Face Detection
Decision
No
Yes
Forming of Rectangle around the face
Face detected successfully
End

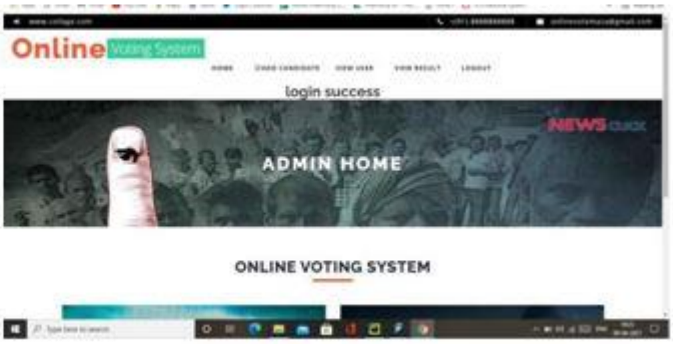
Fig -5: Haar Classifier Flowchart

2.3 RESULTS

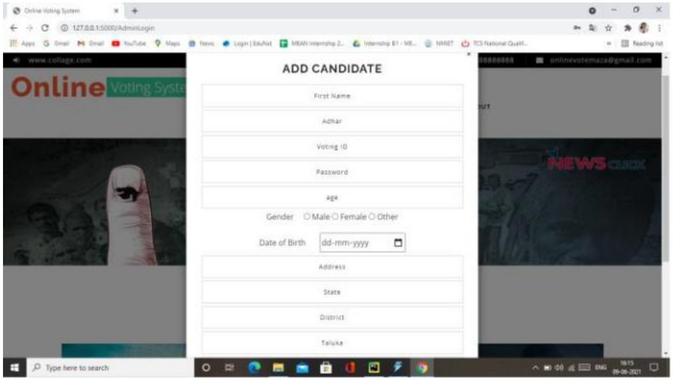
Home page:

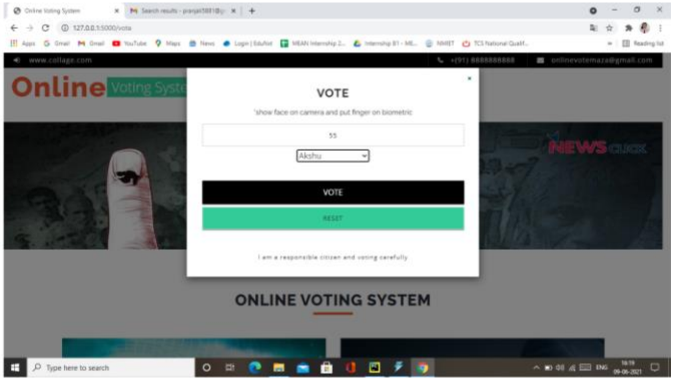


Admin page:



login success





VOTE

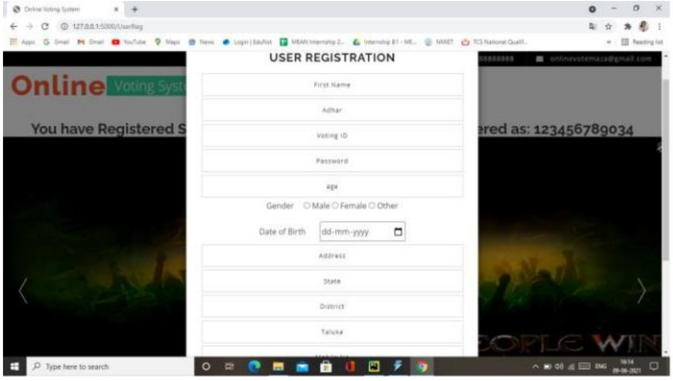
*show face on camera and put finger on biometric

55

Akshu

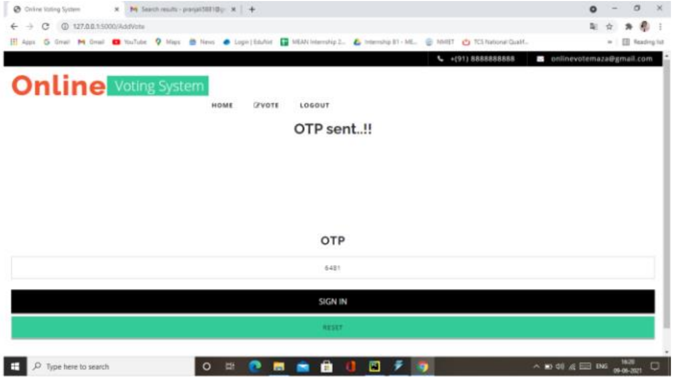
I am a responsible citizen and voting carefully

User Registration:



You have Registered S... ...ered as: 123456789034

OTP Verification Page:

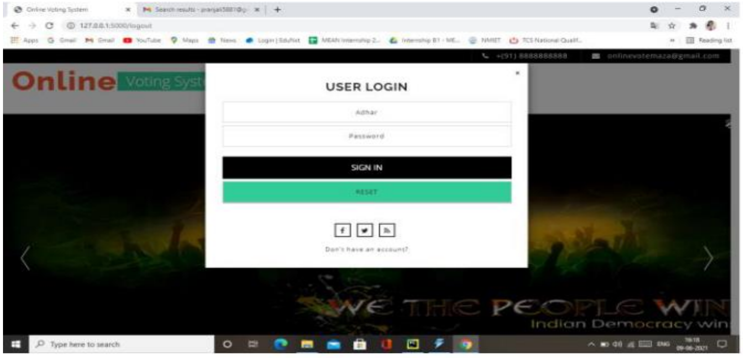


OTP sent..!!

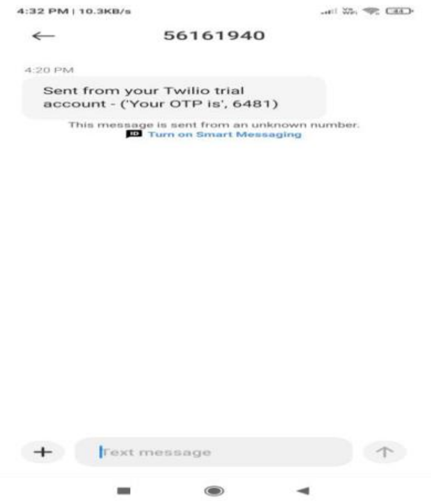
OTP

6481

User Login:



OTP Message On Mobile:


4:32 PM | 10.3KB/s

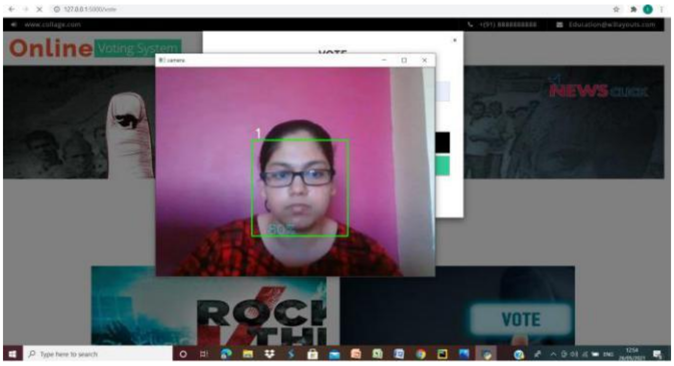
56161940

4:20 PM

Sent from your Twilio trial account - ('Your OTP is', 6481)

This message is sent from an unknown number.
Turn on Smart Messaging

Check Security Question:



[3] CONCLUSIONS

Our suggested approach combines facial identification with machine learning to enable voters to register and cast ballots from any location, regardless of where they are. This technique offers security and prevents one individual from casting numerous votes. This approach, in which we may cast our votes from many locations, is more dependable. Additionally, it reduces work, human needs, and time resources.

REFERENCES

[1] Prof. KritiPatidar, Prof. Swapnil Jain "Decentralized E-Voting Portal Using Blockchain."

[2] Prof. Shashank S Kadam, Ria N Choudhary, SujayDandekar, DebjeetBardhan, Namdeo B Vaidya "Electronic Voting Machine with Enhanced Security"

[3] RahilRezwan, Huzaifa Ahmed, M. R. N. Biplob, S. M. Shuvo,
Md. AbdurRahman "Biometrically Secured Electronic Voting Machine"

[4] Z.A. Usmani, KaifPatanwala, MukeshPanigrahi, Ajay Nair
"Multipurpose platform independent online voting system."

Log-based System Administration:

System logs keep records of an IT system's operational states, events and alerts over the time. For better system administration, log analytics should focused on system post-analysis (forensic), system failure or fault detection, and system failure prediction.[2]

Log clustering and pattern learning:

Log clustering can be done using supervised and unsupervised machine learning algorithms. In Supervised machine learning method , firstly users need to manually label a set of log patterns or categories and use classifier models such as Naive Bayes to perform log text classification [8]. Unsupervised machine learning can be use for log clustering, like hierarchical partitioning method [9] and multi-pass data summarization process [10]. After obtaining cluster structures from the system logs, we can extract the common patterns or features from multiple system logs from the cluster. [2]

Sequence modelling with deep neural networks:

Recurrent Neural Networks (RNN) such as Long Short-Term Memory (LSTM) are powerful models that are capable of learning effective feature representations of sequences when given enough training data.

[3] Analysis of Machine Learning Algorithms for IT Operations

Failure prediction:

A general log pattern can be extracted by using clustering method based on matching features, identical format and content details. Apply Term frequency - inverse document frequency (TFIDF) algorithm to consider each pattern as a word and the set of patterns in each discretized epoch as a document. As system failures occur rarely so Long Short-Term Memory (LSTM) a type of recurrent neural network(RNN) can apply to deal with the “rarity” of labelled data in the model training process.[2]

Alert Classification

The labelled alerts can be use for training the automatic classifier which uses RIPPER algorithm for learning the classification rules. This automatic alert classifier reduces the system admin’s workload. Alert classifier model classifies the alerts as true and false positive alerts. The alerts which classified as false positive by the system admin can be considered for training purpose. For the Alert classification, we can implement and compare different classifier algorithms like Random Forest , Decision Stump, RIPPER, NNge, oneR, PART. [3]

Detection of System Failure Time

The timestamp and the features are exists in the system logs within timeframe. As per these features, using the clustering algorithm we can obtain the cluster of the corresponding time and identify the fault timeframe according to the cluster. As a classifier for Detection of System Failure Time, K-means Clustering can be implement with tuning of K element. [4]

Fault Location

Once fault time detected , we can refer corresponding logs for further analysis with respect to timestamp. TFIDF (Term Frequency-Inverse Document Frequency) algorithm is a commonly use to extract keywords from inputs and weighting technique for information retrieval and mining [5]. Term frequency consider the number of occurrences of a targeted word in the input file. To prevent from biasing towards long files , these number of occurrence is normalized.[4]

Alert or Alarm handling and Noise reduction :

The root cause of any system alarm or alert is the reason for which it occurs. Most of the time system configuration problems is the root cause and does not resolve unless system admin fixes them. Alarm clustering algorithm that groups similar system alarms together and consider them as a single generalized alarm. We can drastically reduce the number of newly generating system alarms , by removing these root causes.

Alert or Alarm Correlation

Manual system alarm correlation is difficult and very time consuming so by using Machine learning algorithms like clustering algorithms we can build Alarm correlation systems (ACSs) where we identify the correlation between alarms. [13]

Anomaly detection

Anomaly detection is the process of identifying unexpected items or events in a given datasets. Clustering and anomaly detection algorithms can be used to detect and flag statistical outliers or anomaly that can be an indicators of a problem. In

- *Supervised Anomaly Detection* - The training and test data sets are fully labelled. Algorithms like decision trees, Support Vector Machines (SVM) or Artificial Neural Networks (ANN) algorithms can be used for Supervised Anomaly Detection.
- *Semi-supervised Anomaly Detection* - Only training data consists of normal data without any anomalies but test data can have anomalies. Well known algorithms such as One-class SVMs and autoencoders can be used for Semi-supervised Anomaly Detection.
- *Unsupervised Anomaly Detection* – This method does not need any labels for training data. Also there is no separation between training and test datasets.

Statistical Performance analysis

Statistical analysis algorithms can be used to discover performance trends and predict future behaviour. The researcher tried to predict the future system load conditions of a resources by considering different measures obtained from the load monitoring systems of servers. These raw data is extremely variable and dynamic which makes it difficult to forecast the behaviour of future resource measures and deduce a clear pattern or trend about the system load behaviour of a resource. We can apply load prediction algorithm based on linear regression model to predict system load trends. [18]

Bug-tracking

Bug-Tracking is very crucial part in any software development and IT industry which assures quality of products. Automatically tracking bug can be consider as classification problem, which takes the bug no or title and bug description as the input for processing and mapping it to any available developers (using labels). The major difficult is that the bug description is in text format which usually includes a combination of free unstructured strings, code snippets, which makes input data highly noisy and hard to process and analyse. The bag-of-words (BOW) model does not consider the syntactical and sequential word information available in the unstructured text. In recent research , they proposed bug report representation algorithm using an attention based deep bidirectional recurrent neural network (DBRNN-A) model which learns a syntactic and semantic feature from long word sequences in an unsupervised manner. Instead of just BOW features, the DBRNN-A based bug representation is then used for training the classifier. Using an attention mechanism enables the model to learn the context representation over a long word sequence, as in a bug report. [15]

Bug Assignment

In Quality Assurance or Software Testing , for effective bug resolution , it is very important to assign the reported bug to a respective developer or engineer. Bug assignment is an initial part of bug tracking whose objective is to assign a respective developer or engineer to the reported bug. The assigned developer or engineer can perform various checks , troubleshoots the issue and do the changes in the source code to resolve the reported issue. The selection and assignment of a respective developer for specific bug is a challenging , time and cost consuming process in the project. From recent research , we can leverage multiple Machine Learning algorithms such as Naïve Bayes, Support Vector Machine (SVM), C4.5, Expectation Maximization, Conjunctive Rules and the Nearest Neighbour (NN) algorithm for the selection and assignment of developer or engineer resources to specific bugs. [16]

[5] SUMMARY

In this paper, we analyse various machine learning algorithms and their application in a complex and critical IT systems based on logs , events and alerts. Recent research work focuses on anomaly detection in running system, alert or failure prediction , finding root cause of system failure etc. using different system logs. Also system logs can be leverage to understand the system behaviour , common features, correlation between events and recommendations for remediations based on historical resolutions. There is need of Domain based Algorithms that leverages IT Operation domain expertise (specific to any environment) to intelligently analyse, process, interpret and implement the rules , patterns and models , as directed by an organization's domain specific data and its expected outcomes. These algorithms should target IT Operations specific goals like eliminating alert or alarm noise, correlating unstructured log or event data, setup baselines or thresholds, alarming on abnormalities , try to identify possible root causes and predict remediations.

REFERENCES

[1] Ding Yuan, Haohui Mai, Weiwei Xiong, Lin Tan, Yuanyuan Zhou, Shankar Pasupathy "SherLog: Error Diagnosis by Connecting Clues from Run-time Logs" ,Association for Computing Machinery, New York, NY, USA 2010

[2] Ke Zhang , Jianwu Xu , Martin Renqiang Min , Guofei Jiang , Konstantinos Pelechrinis and Hui Zhang "Automated IT System Failure Prediction: A Deep Learning Approach", 2016 IEEE International Conference on Big Data (Big Data)

[3] Dr T Subbulakshmi, S. Mercy Shalinie "Real Time Classification and Clustering Of IDS Alerts Using Machine Learning Algorithms" , International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 1, No.1, January 2010

[4] Leyi Zhang, Lei Fan and Naiwang Guo ," Log-Based OpenStack Fault Diagnosis by Machine Learning". Journal of Physics: Conference Series, Volume 1069, conference1.

[5] Chum O, Philbin J, Zisserman A. "Near Duplicate Image Detection: min-Hash and TFIDF Weighting [J]". 2008.

[6] Ruben Sipos, Cornel U. Dmitriy Fradkin, Siemens Fabian Moerchen, Amazon Zhuang (John) Wang, Skytree, "Log-based Predictive Maintenance" , ACM New York, NY, USA KDD (Knowledge Discovery and Data Mining)

[7] Li, T., Liang, F., Ma, S., Peng, W.: "An integrated framework on mining logs files for computing system management." ACM New York, NY, USA KDD (Knowledge Discovery and Data Mining) (2005)

[8] T. Li, F. Liang, S. Ma, and W. Peng, "An integrated framework on mining logs files for computing system management," in Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005, pp. 776–781

[9] A. A. Makanju, A. N. Zincir-Heywood, and E. E. Milios, "Clustering event logs using iterative partitioning," in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009, pp. 1255–1264.

[10] R. Vaarandi et al., "A data clustering algorithm for mining patterns from event logs," in Proceedings of the 2003 IEEE Workshop on IP Operations and Management (IPOM), 2003, pp. 119–126.

[11] Klaus Julisch, IBM Research, Zurich Research Laboratory, Rüschlikon, Switzerland "Clustering intrusion detection alarms to support root cause analysis", ACM New York, NY, USA, ISSN: 1094-9224 EISSN: 1557-7406

[12] Klaus Julisch IBM Research, "Mining Alarm Clusters to Improve Alarm Handling Efficiency" , IEEE Seventeenth Annual Computer Security Applications Conference

[13] Klaus Julisch IBM Research, Zurich Research Laboratory, Rüschlikon, Switzerland, "Clustering intrusion detection alarms to support root cause analysis" , ACM Transactions on Information and System Security (TISSEC)

[14] Markus Goldstein, Seiichi Uchida, "A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data", PLOS ONE DOI:10.1371/journal.pone.0152173

[15] Senthil Mani, Anush Sankaran, Rahul Aralikatte, "DeepTriage: Exploring the Effectiveness of Deep Learning for Bug Triaging" arXiv:1801.01275v1

[16] Anjali Goyal, Neetu Sardana, "Machine Learning or Information Retrieval Techniques for Bug Triaging: Which is Better?" e-Informatica Software Engineering Journal, Volume 11, Issue 1, 2017, pages: 117–141, DOI 10.5277/e-Inf170106

[17] Mario MILICEVIC, Mirta BARANOVIC, Krunoslav ZUBRINIC, “Application of Machine Learning Algorithms for the Query Performance Prediction” Advances in Electrical and Computer Engineering (AECE)