



Task Failure Prediction in Cloud Data Centers Using Deep Learning

Ganesh Karthikeyan Relli^{1*}, Saka Uma Maheswara Rao, M.Tech[Ph.D]², Tulasiraju Nethala, M.Tech[P.hD]³, Dr. P. Srinivasulu, M.Tech,P.hD⁴

¹*Student, Department of Computer Science and Engineering, Swarnandhra College Of Engineering and Technology, Seetharamapuram, W.G.Dt, Andhra Pradesh, India.

²Associate Professor, Department of Computer Science and Engineering, Swarnandhra College Of Engineering and Technology, Seetharamapuram, W.G.Dt, Andhra Pradesh, India.

³Associate Professor, Department of Computer Science and Engineering, Swarnandhra College Of Engineering and Technology, Seetharamapuram, W.G.Dt, Andhra Pradesh, India.

⁴Professor & HOD, Department Of Computer Science and Engineering, Swarnandhra College Of Engineering and Technology, Seetharamapuram, W.G.Dt, Andhra Pradesh, India.

***Corresponding Author:** - Ganesh Karthikeyan Relli

*Student, Department of Computer Science and Engineering, Swarnandhra College Of Engineering and Technology, Seetharamapuram, W.G.Dt, Andhra Pradesh, India.

Abstract:

A large-scale cloud data center needs to provide high service reliability and availability with low failure occurrence probability. However, current large-scale cloud data centers still face high failure rates due to many reasons such as hardware and software failures, which often result in task and job failures. Such failures can severely reduce the reliability of cloud services and also occupy huge amount of resources to recover the service from failures. Therefore, it is important to predict task or job failures before occurrence with high accuracy to avoid unexpected wastage. Many machine learning and deep learning based methods have been proposed for the task or job failure prediction by analyzing past system message logs and identifying the relationship between the data and the failures. In order to further improve the failure prediction accuracy of the previous machine learning and deep learning based methods, in this paper, we propose a failure prediction algorithm based on multi-layer Bidirectional Long Short Term Memory (Bi-LSTM) to identify task and job failures in the cloud. The goal of Bi-LSTM prediction algorithm is to predict whether the tasks and jobs are failed or completed. The trace-driven experiments show that our algorithm outperforms other state-of-art prediction methods with 93% accuracy and 87% for task failure and job failures respectively.

Keywords –Cloud datacenters and deep learning.

1. INTRODUCTION

Nowadays, cloud computing service has been widely used because it provides high reliability, resource saving and also on-demand services. The cloud data centers include processors, memory units, disk drives, networking devices, and various types of sensors that support many applications (i.e., jobs) from users. The users can send requests such as store data and run applications to the cloud. Each cloud data center is composed with physical machines (PMs) and each PM can support a set of virtual machines (VMs). The tasks that are sent from users are processed in each VM. Such a large scale cloud data center can host hundreds of thousands of servers which often run tons of applications and receive work requests every second from users all over the world. A cloud data center with such heterogeneity and intensive workloads may sometimes be vulnerable to different types of failures (e.g., hardware, software, disk failures). Take software failures as an example, Ya-hoo Inc. and Microsoft's search engine, Bing, crashed for 20 mins in January 2015, which cost about \$9000 per minute to reboot the system. Previous research found that hardware failure, especially disk failure, is a major contributing factor to the outages of cloud services. These many different types of failures will lead to the application running failures. Thus, accurate prediction for the occurrence of application failures beforehand can improve the efficiency of recovering the failure and application running.

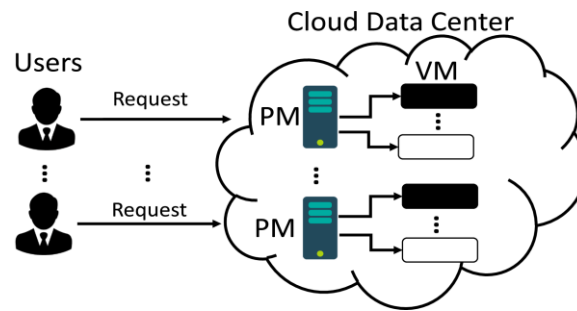


Fig.1: Example figure

A job is comprised of one or more tasks, each of which is accompanied by a set of resource requirements. A job fails when one of its tasks fails. The previous works [3], [7]–[13] use statistical and machine learning approaches such as Hidden Semi-markov Model (HSMM) and Support Vector Machine (SVM) to predict the task and job failures in cloud data centers. They use CPU usage and memory usage, unmapped page cache, mean disk I/O time and disk usage as inputs and the task failure or job failure as the output. However, HSMM and SVM assume that all their inputs are stationary and independent of each other which are not true in the cloud data centers. Thus, they cannot handle the sequence data or high dimensional data, in which data in time points or different features may be dependent to each other. In the cloud data centers, the input features and noisy data are diverse in nature and have dependencies on the past events. Thus HSMM and SVM can't handle the failure prediction in cloud data centers.

2. LITERATURE REVIEW

Diehard: reliable scheduling to survive correlated failures in cloud data centers:

In large scale data centers, a single fault can lead to correlated failures of several physical machines and the tasks running on them, simultaneously. Such correlated failures can severely damage the reliability of a service or a job. This paper models the impact of stochastic and correlated failures on job reliability in a data center. We focus on correlated failures caused by power outages or failures of network components, on jobs running multiple replicas of identical tasks. We present a statistical reliability model and an approximation technique for computing a job's reliability in the presence of correlated failures. In addition, we address the problem of scheduling a job with reliability constraints. We formulate the scheduling problem as an optimization problem, with the aim being to achieve the desired reliability with the minimum number of extra tasks. We present a scheduling algorithm that approximates the minimum number of required tasks and a placement to achieve a desired job reliability. We study the efficiency of our algorithm using an analytical approach and by simulating a cluster with different failure sources and reliabilities. The results show that the algorithm can effectively approximate the minimum number of extra tasks required to achieve the job's reliability.

Failure prediction of data centers using time series and fault tree analysis

This paper proposes a framework for online failure prediction of data centers. A data center often has a high failure rate as it features a number of servers and components. Moreover, long running applications and intensive workloads are common in such facilities. Performance of the system depends on the availability of the machines, which can be easily compromised if failure cannot be handled gracefully. The main idea of this paper is to create an effective prediction model focusing on hardware failure. Accurate prediction may enhance the overall system performance. In this work, we employ two methods, namely, ARMA (Auto Regressive Moving Average) and Fault Tree Analysis. Experiments were then performed on a simulated cluster built based on Simi's platform. The results show prediction accuracy of 97%, which is very high. We thus believe that our framework is practical and can be adapted to use in data centers in the future.

Partial-parallel-repair (ppr): a distributed technique for repairing erasure coded storage

With the explosion of data in applications all around us, erasure coded storage has emerged as an attractive alternative to replication because even with significantly lower storage overhead, they provide better reliability against data loss. Reed-Solomon code is the most widely used erasure code because it provides maximum reliability for a given storage overhead and is flexible in the choice of coding parameters that determine the achievable reliability. However, reconstruction time for unavailable data becomes prohibitively long mainly because of network bottlenecks. Some proposed solutions either use additional storage or limit the coding parameters that can be used. In this paper, we propose a novel distributed reconstruction technique, called Partial Parallel Repair (PPR), which divides the reconstruction operation to small partial operations and schedules them on multiple nodes already involved in the data reconstruction. Then a distributed protocol progressively combines these partial results to reconstruct the unavailable data blocks and this technique reduces the network pressure. Theoretically, our technique can complete the network transfer in $\lceil \log_2(k+1) \rceil$ time, compared to k time needed for a (k, m) Reed-Solomon code. Our experiments show that PPR reduces repair time and degraded read time significantly. Moreover, our technique is compatible with existing erasure codes and does not require any additional storage overhead. We demonstrate this by overlaying PPR on top of two prior schemes, Local Reconstruction Code and Rotated Reed-Solomon code, to gain additional savings in reconstruction time.

Approaches for resilience against cascading failures in cloud datacenters

In a modern cloud datacenter, a cascading failure will cause many Service Level Objective (SLO) violations. In a cascading failure, when a set of physical machines (PMs) in a failure domain are failed, their workloads are transferred to the PMs in another failure domain to continue. However, the new domain receiving additional workloads may become overloaded due to the resource oversubscription feature in the cloud, which easily leads to domain failures and subsequent workload transfer to other domains. This process repeats and a cascading failure is created finally. However, few previous methods can effectively handle the cascading failures. To handle this problem, we propose a Cascading Failure Resilience System (CFRS), which incorporates three methods: Overload-Avoidance VM Reassignment (OAVR), VM backup set placement (VMset) and Dynamic Oversubscription Ratio Adjustment (DOA). The experiments in trace-driven simulation show that CFRS outperforms other comparison methods in terms of the number of domain failures, the number of failed PMs and the number of SLO violations.

Proactive incast congestion control in a datacenter serving web applications

With the rapid development of web applications in datacenters, network latency becomes more important to user experience. The network latency will be greatly increased by incast congestion, in which a huge number of requests arrive at the front-end server simultaneously. Previous incast problem solutions usually handle the data transmission between the data servers and the front-end server directly, and they are not sufficiently effective in proactively avoiding incast congestion. To further improve the effectiveness, in this paper, we propose a Proactive Incast Congestion Control system (PICC). Since each connection has bandwidth limit, PICC novelly limits the number of data servers concurrently connected to the front-end server to avoid the incast congestion through data placement. Specifically, the front-end server gathers popular data objects (i.e., frequently requested data objects) into as few data servers as possible, but without overloading them. It also re-allocates the data objects that are likely to be concurrently or sequentially requested into the same server. As a result, PICC reduces the number of data servers concurrently connected to the front-end server (which avoids the incast congestion), and also the number of connection establishments (which reduces the network latency). Since the selected data servers tend to have long queues to send out data, to reduce the queuing latency, PICC incorporates a queuing delay reduction algorithm that assigns higher transmission priorities to data objects with smaller sizes and longer queuing times. The experimental results on simulation and a real cluster based on a benchmark show the superior performance of PICC over previous incast congestion problem solutions.

3.METHODOLOGY

However, the LSTM based prediction methods still have a few shortcomings. First, the methods only consider CPU usage, memory usage, cache memory usage, mean disk I/O time, and disk usage as input features. More input features may further increase the prediction accuracy. Second, the LSTM based prediction model used single-layer LSTM construction which can not handle multiple input features well compared to multi-layer construction. Third, in the cloud data center, input features like CPU usage and memory usage are highly related over time. For a given time for prediction, the LSTM based prediction model always sets higher weights on the data closer to the time, and lower weights on the data further away from the time, with the assumption that the data further away from the time always has lower impact to the prediction. However, such settings cannot accurately reflect the impact degree as the further data may still have higher impact on the failure (e.g., failures in long term jobs). The performance can be better if the weights of data items are determined based on the real data trace. Thus, a new prediction model is needed to build to implement failure prediction in the cloud data center in order to achieve better accuracy in prediction.

Disadvantages:

1. However, current large-scale cloud data centers still face high failure rates due to many reasons such as hardware and software failures, which often result in task and job failures.
2. Such failures can severely reduce the reliability of cloud services and also occupy huge amount of resources to recover the service from failures.

To overcome the shortcomings, in this paper, we build a failure prediction model based on multi-layer Bidirectional LSTM and name it as Bi-LSTM. First, Bi-LSTM has more input features than previous methods, which include task priority, task resubmissions and scheduling delay. Second, Bi-LSTM has multi-layer structure which can better handle multiple input features for higher accuracy. Multi-layer construction can narrow the number of parameters in the calculating functions but still with the same number of neurons which can reduce the calculation time. Third, Bi-LSTM can determine the weights of data items based on their real impact to the failure rather than simply setting higher weights to data item closer to the given time for prediction than the data items further away from the time. We perform the trace-driven failure prediction study by using Google cluster trace and we compare the performance of Bi-LSTM with other state-of-art prediction methods.

Advantages:

1. Our algorithm detects task failures and job failures with high accuracy.
2. We also observe that the time cost overhead for Bi-LSTM is almost the same compared with RNN and LSTM, which means Bi-LSTM can achieve higher prediction performance with no further time cost.

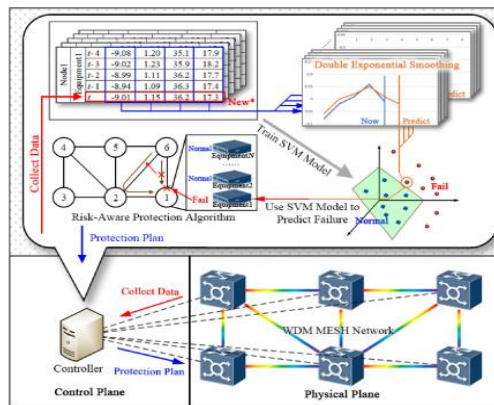


Fig.2: System architecture

MODULES:

To implement aforementioned project we have designed following modules

- Data exploration: using this module we will load data into system
- Processing: Using the module we will read data for processing
- Splitting data into train & test: using this module data will be divided into train & test
- Model generation: Random Forest - Decision Tree - KMM - Support Vector Machine - Voting Classifier - CNN - CNN+LSTM - LSTM - BiLSTM - RNN - CNN with KFold Validation.
- User signup & login: Using this module will get registration and login
- User input: Using this module will give input for prediction
- Prediction: final predicted displayed

4. IMPLEMENTATION

ALGORITHMS:

Random Forest:

Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

Decision Tree:

Decision trees use multiple algorithms to decide to split a node into two or more sub-nodes. The creation of sub-nodes increases the homogeneity of resultant sub-nodes. In other words, we can say that the purity of the node increases with respect to the target variable.

KNN:

K Nearest Neighbour is a simple algorithm that stores all the available cases and classifies the new data or case based on a similarity measure. It is mostly used to classify a data point based on how its neighbours are classified.

Support Vector Machine:

Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. Though we say regression problems as well its best suited for classification. The objective of SVM algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points.

Voting Classifier:

A voting classifier is a machine learning estimator that trains various base models or estimators and predicts on the basis of aggregating the findings of each base estimator. The aggregating criteria can be combined decision of voting for each estimator output.

CNN:

A CNN is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data. There are other types of neural networks in deep learning, but for identifying and recognizing objects, CNNs are the network architecture of choice.

LSTM:

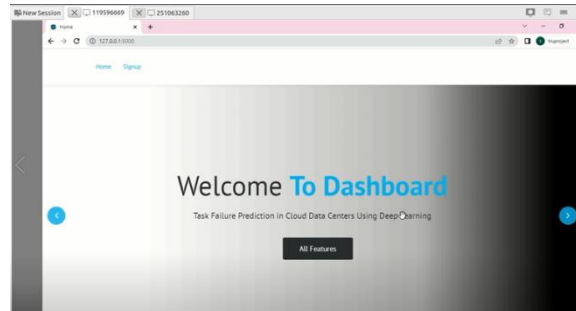
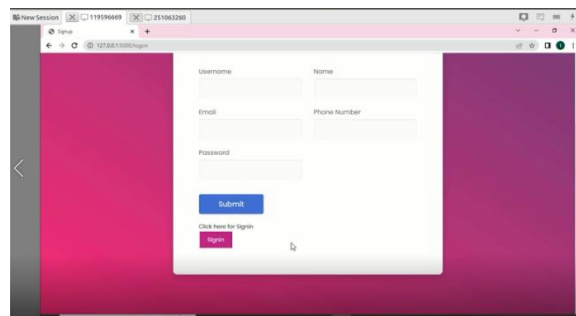
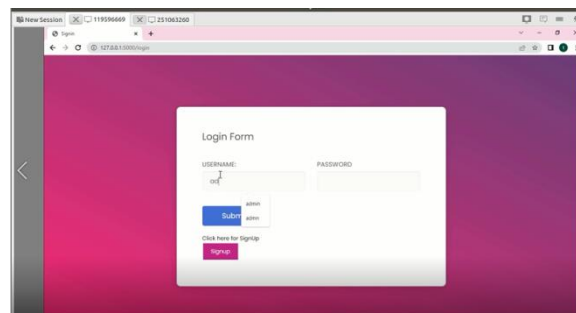
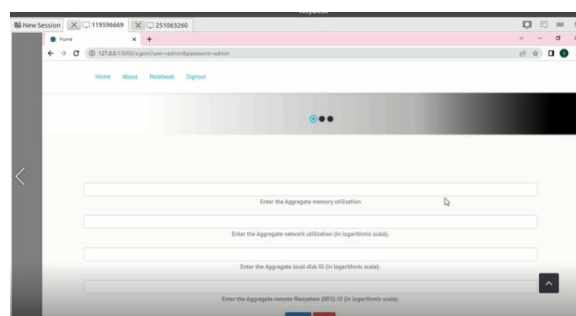
Long short-term memory (LSTM) is an artificial neural network used in the fields of artificial intelligence and deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. Such a recurrent neural network (RNN) can process not only single data points (such as images), but also entire sequences of data (such as speech or video).

BiLSTM:

Bidirectional Long Short-Term Memory (BiLSTM) In general time series processing, LSTM often ignores future information. BiLSTM uses two separate hidden layers to process series data in forward and reverse directions on the basis of LSTM, connecting the two hidden.

RNN:

A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes can create a cycle, allowing output from some nodes to affect subsequent input to the same nodes. This allows it to exhibit temporal dynamic behavior. Derived from feedforward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition. Recurrent neural networks are theoretically Turing complete and can run arbitrary programs to process arbitrary sequences of inputs.

5. EXPERIMENTAL RESULTS**Fig.3:** Home screen**Fig.4:** User registration**Fig.5:** user login**Fig.6:** Main screen

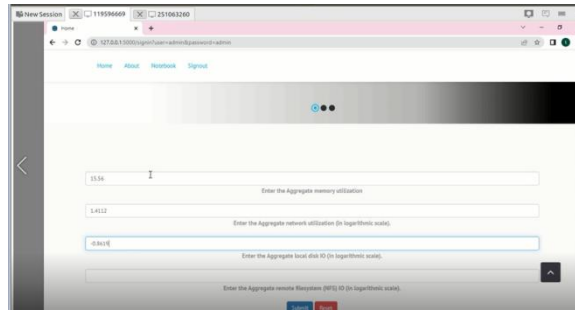


Fig.7: User input

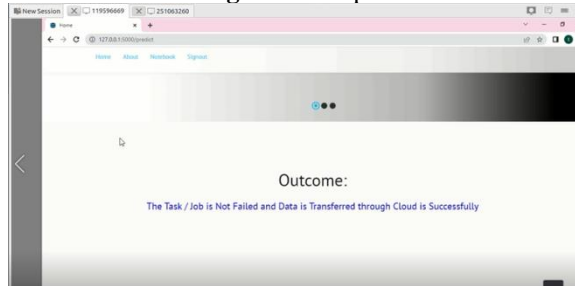


Fig.8: Prediction result

6. CONCLUSION

In cloud data centers, high service reliability and availability are crucial to application QoS. In this paper, we proposed a failure prediction model multi-layer Bidirectional LSTM (called Bi-LSTM). Bi-LSTM can more accurately predict the termination statuses of tasks and jobs using Google cluster trace compared with previous methods. In our method, we first input the data into forward state and backward state in order to adjust the weight of both closer and further input features. We then find that the further input features is essential to achieving high prediction accuracy. Secondly, in the experiments, we compare Bi-LSTM with other comparison methods including statistical, machine learning and deep learning based methods and evaluate the performance with three metrics: accuracy and F1 score, receiver operating characteristic and time cost overhead. The results show that we achieved 93% accuracy in task failure prediction and 87% accuracy in job failure prediction. We also achieved 92% F1 score in task failure prediction and 86% F1 score in job failure prediction. Our prediction method Bi-LSTM also have low FPR which can also indicate the proactive failure management based on prediction results become more effective. We also observe that the time cost overhead for Bi-LSTM is almost the same compared with RNN and LSTM, which means Bi-LSTM can achieve higher prediction performance with no further time cost.

REFERENCES

- [1] "<https://techcrunch.com/2015/01/02/following-bing-coms-brief-outagesearch-yahoo-com-goes-down-too/>, [Accessed in APR 2019]."
- [2] M. Sedaghat, E. Wadbro, J. Wilkes, S. De Luna, O. Seleznev, and E. Elmroth, "Diehard: reliable scheduling to survive correlated failures in cloud data centers," in 2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 2016.
- [3] T. Chalermarwong, T. Achalakul, and S. See, "Failure prediction of data centers using time series and fault tree analysis," in 2012 IEEE 18th International Conference on Parallel and Distributed Systems, 2012.
- [4] S. Mitra, M. Ra, and S. Bagchi, "Partial-parallel-repair (ppr): a distributed technique for repairing erasure coded storage," in Proceedings of the eleventh European conference on computer systems, 2016.
- [5] H. Wang, H. Shen, and Z. Li, "Approaches for resilience against cascading failures in cloud datacenters," in Proc. of ICDCS, 2018.
- [6] H. Wang and H. Shen, "Proactive incast congestion control in a datacenter serving web applications," in Proc. of INFOCOM, 2018.
- [7] R. Baldoni, L. Montanari, and M. Rizzuto, "On-line failure prediction in safety-critical systems," Future Generation Computer Systems, 2015.
- [8] Y. Zhao, X. Liu, S. Gan, and W. Zheng, "Predicting disk failures with hmm-and hsmm-based approaches," in Industrial Conference on Data Mining, 2010.
- [9] J. Murray, G. Hughes, and K. Kreutz-Delgado, "Machine learning methods for predicting failures in hard drives: A multiple-instance application," Journal of Machine Learning Research, 2005.
- [10] I. Fronza, A. Sillitti, G. Succi, M. Terho, and J. Vlasenko, "Failure prediction based on log files using random indexing and support vector machines," Journal of Systems and Software, 2013.
- [11] Q. Guan, Z. Zhang, and S. Fu, "Ensemble of bayesian predictors and decision trees for proactive failure management in cloud computing systems," Journal of Communications, 2012.
- [12] T. Pitakrat, D. Okanovic, A. van Hoorn, and L. Grunske, "Hora: Architecture-aware online failure prediction," Journal of Systems and Software, 2018.

- [13] S. Zhang, Y. Liu, Z. Meng, W. and Luo, J. Bu, P. Yang, S. and Liang, D. Pei, J. Xu, and Y. Zhang, "Prefix: Switch failure prediction in datacenter networks," Proceedings of the ACM on Measurement and Analysis of Computing Systems, 2018.
- [14] C. Xu, G. Wang, X. Liu, D. Guo, and T. Liu, "Health status assessment and failure prediction for hard drives with recurrent neural networks," IEEE Transactions on Computers, 2016.
- [15] Y. Cheng, H. Zhu, J. Wu, and X. Shao, "Machine health monitoring using adaptive kernel spectral clustering and deep long short-term memory recurrent neural networks," IEEE Transactions on Industrial Informatics, 2019.
- [16] SRILATHA PULI, A MACHINE LEARNING MODEL FOR AIR QUALITY PREDICTION FOR SMART CITIES, DESIGN ENGINEERING || ISSN: 0011-9342 | YEAR 2021 - ISSUE: 9 | PAGES: 18090 – 18104
- [17] SRILATHA PULI, QUALITY RISK ANALYSIS FOR SUSTAINABLE SMART WATER SUPPLY USING DATA PERCEPTION, INTERNATIONAL JOURNAL OF HEALTH SCIENCES ISSN 2550-6978 E-ISSN 2550-696X © 2022, [HTTPS://DOI.ORG/10.53730/IJHS.V6NS5.9826](https://doi.org/10.53730/IJHS.V6NS5.9826), 18 JUNE 2022
- [18] SRILATHA PULI, URBAN STREET CLEANLINESS, JOURNAL OF ALGEBRAIC STATISTICS VOLUME 13, NO. 3, 2022, P. 547-552, [HTTPS://PUBLISHOA.COM](https://publishoa.com), and ISSN: 1309-3452
- [19] SRILATHA PULI, SELF-ANNIHILATION IDEATION DETECTION, and NEUROQUANTOLOGY | JUNE 2022 | VOLUME 20 | ISSUE 6 | PAGE 7229-7239 | DOI: 10.14704/NQ.2022.20.6.NQ22727
- [20] SRILATHA PULI, CRIME ANALYSIS USING MACHINE LEARNING, YMER|| ISSN: 0044-0477, APRIL 2022
- [21] SRILATHA PULI, N-GRAMS ASSISTED YOUTUBE SPAM COMMENT DETECTION, YMER || ISSN: 0044-0477, APRIL 2022
- [22] SRILATHA PULI, ANALYSIS OF BRAND POPULARITY USING BIG DATA AND TWITTER, YMER|| ISSN: 0044-0477, APRIL 2022
- [23] SRILATHA PULI, CYBER THREAT DETECTION BASED ON ARTIFICIAL NEURAL NETWORKS USING EVENT PROFILES, THE INTERNATIONAL JOURNAL OF ANALYTICAL AND EXPERIMENTAL MODAL ANALYSIS, ISSN NO: 0886-9367
- [24] SRILATHA PULI, FACE MASK MONITORING SYSTEM, THE INTERNATIONAL JOURNAL OF ANALYTICAL AND EXPERIMENTAL MODAL ANALYSIS, ISSN NO: 0886-9367
- [25] SRILATHA PULI, IOT BASED SMART DOOR LOCK SURVEILLANCE SYSTEM USING SECURITY SENSORS, ADVANCED SCIENCE LETTERS E-ISSN: 1936-7317
- [26] SRILATHA PULI, SAFETY ALERTING SYSTEM FOR DROWSY DRIVER, 9TH INTERNATIONAL CONFERENCE ON INNOVATIONS IN ELECTRONICS & COMMUNICATION ENGINEERING (ICIECE-2021), PAGE – 40
- [27] N. SWAPNA SUHASINI, SRILATHA PULI, BIG DATA ANALYTICS FOR MALWARE DETECTION IN A VIRTUALIZED FRAMEWORK, JOURNAL OF CRITICAL REVIEWS, ISSN:2394-5125 VOL.7, ISSUE 14, JULY – 2020
- [28] SRILATHA PULI, BLOCK CHAIN BASED CERTIFICATE VALIDATION, INTERNATIONAL JOURNAL OF SCIENCE AND RESEARCH (IJSR), and ISSN: 2319-7064 SJIF (2022): 7.942, VOLUME 11 ISSUE 12, DECEMBER 2022, PAPER ID: SR221219113003, DOI: 10.21275/SR221219113003, WWW.IJSR.NET
- [29] MRS. SRILATHA PULI, ENERGY EFFICIENT TEACHING-LEARNING-BASED OPTIMIZATION FOR THE DISCRETE ROUTING PROBLEM IN WIRELESS SENSOR NETWORK, INTERNATIONAL JOURNAL OF EARLY CHILDHOOD SPECIAL EDUCATION (INT-JECS) DOI: 10.48047/INTJECSE/V14I7.296 ISSN: 1308-5581 VOL 14, ISSUE 07 2022.
- [30] MRS. SRILATHA PULI, A HYBRID BLOCK CHAIN-BASED IDENTITY AUTHENTICATION SCHEME FOR MULTI- WSN, INTERNATIONAL JOURNAL OF EARLY CHILDHOOD SPECIAL EDUCATION (INT-JECS) DOI: 10.48047/INTJECSE/V14I7.296 ISSN: 1308-5581 VOL 14, ISSUE 07 2022
- [31] MRS. SRILATHA PULI, IMPLEMENTATION OF A SECURED WATERMARKING MECHANISM BASED ON CRYPTOGRAPHY AND BIT PAIRS MATCHING, INTERNATIONAL JOURNAL OF EARLY CHILDHOOD SPECIAL EDUCATION (INT-JECS) DOI: 10.48047/INTJECSE/V14I7.296 ISSN: 1308-5581 VOL 14, ISSUE 07 2022
- [32] MRS. S.SUNITHA, MRS. SRILATHA PULI, MULTILEVEL DATA CONCEALING TECHNIQUE USING STEGANOGRAPHY AND VISUAL CRYPTOGRAPHY, INTERNATIONAL JOURNAL OF EARLY CHILDHOOD SPECIAL EDUCATION (INT-JECSE) DOI:10.48047/INTJECSE/V15I1.1 ISSN: 1308-5581 VOL 15, ISSUE 01 2023
- [33] MRS. SRILATHA PULI, BLOOD BANK MANAGEMENT DONATION AND AUTOMATION, SPECIALUSIS UGDYMAS / SPECIAL EDUCATION 2022 1 (43), [HTTPS://WWW.SUMC.LT/INDEX.PHP/SE/ARTICLE/VIEW/1995](https://www.sumc.lt/index.php/se/article/view/1995)
- [34] N. S. SUHASINI AND S. PULI, "BIG DATA ANALYTICS IN CLOUD COMPUTING," 2021 SIXTH INTERNATIONAL CONFERENCE ON IMAGE INFORMATION PROCESSING (ICIIP), SHIMLA, INDIA, 2021, PP. 320-325, DOI: 10.1109/ICIIP53038.2021.9702705.
- [35] SURARAPU SUNITHA, BLOCKCHAIN-BASED ACCESS CONTROL SYSTEM FOR CLOUD STORAGE, YMER || ISSN: 0044-0477, APRIL 2022
- [36] SURARAPU SUNITHA, ARTIFICIAL INTELLIGENCE SUPPORT FOR CLOUD COMPUTING INTRUSION, DEEP-CLOUD ISSUES, SOLID STAGE TECHNOLOGY, VOLUME:63, ISSUE:2S, PUBLICATION-2020

- [37] SURARAPU SUNITHA, CRYPTOCURRENCY PRICE ANALYSIS USING ARTIFICIAL INTELLIGENCE, JOURNAL OF ALGEBRAIC STATISTICS VOLUME 13, NO. 3, 2022, P. 486-493 [HTTPS://PUBLISHOA.COM](https://publishoa.com) ISSN: 1309-3452
- [38] SURARAPU SUNITHA, AN EMPIRICAL STUDY ON SECURITY ISSUES AND MITIGATION TECHNIQUES IN OPPORTUNITIES NETWORKS, THINK INDIA JOURNAL, ISSN:0971-1260, VOL-22, ISSUE-41, DECEMBER-2019
- [39] SURARAPU SUNITHA, A HYBRID BLOCK CHAIN-BASED IDENTITY AUTHENTICATION SCHEME FOR MULTI-WSN, INTERNATIONAL JOURNAL OF EARLY CHILDHOOD SPECIAL EDUCATION, ISSN: 1308-5581, VOL 14, ISSUE JULY-2022
- [40].SUNITHASURARAPU, CRYPTOCURRENCY PRICE PREDICTION USING NEURAL NETWORKS, DEEP LEARNING AND MACHINE LEARNING , INTERNATIONAL JOURNAL FOR INNOVATIVE ENGINEERING AND MANAGEMENT RESEARCH, ISSN:2456-5083, VOLUME 12, ISSUE 05 MAY 2023, PAGES:408-417
- [41] SUNITHA SURARAPU, MULTILEVEL DATA CONCEALING TECHNIQUE USING STEGANOGRAPHY AND VISUAL CRYPTOGRAPHY, INTERNATIONAL JOURNAL OF EARLY CHILDHOOD SPECIAL EDUCATION, VOLUME 15, ISSN:1308-5581, IISUE JANUARY 2023