IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 5, May 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305431 |

Task Failure Prediction in Cloud Data Centers Using Deep Learning

Dr. Manoranjan Dash[1], G. Teja Manoj Kumar[2], N. Suditya Sena[3], V. Yogeshwar Reddy [4]

Associate Professor, Department of Artificial Intelligence, Anurag University, Hyderabad, India[1]

U.G. Student, Department of Artificial Intelligence, Anurag University, Hyderabad, India[2,3,4]

ABSTRACT: A large-scale cloud information center needs to give tall benefit unwavering quality and accessibility with moo disappointment event likelihood. In any case, current large-scale cloud information centers still confront tall disappointment rates due to numerous reasons such as equipment and computer program disappointments, which regularly result in errand and work disappointments. Such disappointments can extremely decrease the unwavering quality of cloud administrations and too possess colossal sum of assets to recuperate the benefit from disappointments. Hence, it is critical to foresee assignment or work disappointments some time recently event with tall exactness to dodge unforeseen wastage. Numerous machine learning and profound learning based strategies have been proposed for the errand or work disappointment forecast by analyzing past framework message logs and recognizing the relationship between the information and the disappointments. In arrange to encourage make strides the disappointment expectation exactness of the past machine learning and profound learning based strategies, in this paper, we propose a disappointment expectation calculation based on multi-layer Bidirectional Long Brief Term Memory (Bi-LSTM) to distinguish errand and work disappointments in the cloud. The objective of Bi-LSTM forecast calculation is to foresee whether the assignments and occupations are fizzled or completed. The trace-driven tests appear that our calculation beats other state-of-art expectation strategies with 93% precision and 87% for errand disappointment and work disappointments individually.

I. INTRODUCTION

These days, cloud computing benefit has been fiercely utilized since it gives tall unwavering quality, asset sparing and moreover on-demand administrations. The cloud information centers incorporate processors, memory units, disk drives, organizing gadgets, and different sorts of sensors that back numerous applications (i.e., occupations) from clients. The clients can send demands such as store information and run applications to the cloud. Each cloud information center is composed with physical machines (PMs) and each PM can bolster a set of virtual machines (VMs). The assignments that are sent from clients are prepared in each VM. Such a expansive scale cloud information center can have hundreds of thousands of servers which regularly run tons of applications and get work demands each moment from clients all over the world. A cloud information center with such heterogeneity and seriously workloads may some of the time be defenseless to diverse sorts of disappointments (e.g., equipment, computer program, disk disappointments). Take program disappointments as an illustration, Ya-hoo Inc. and Microsoft's look motor, Bing, slammed for 20 mins in January 2015, which taken a toll approximately $9000 per miniature to reboot the framework. Past inquire about found that equipment disappointment, particularly disk disappointment, is a major contributing calculate to the blackouts of cloud administrations. These numerous distinctive sorts of disappointments will lead to the application running disappointments. In this way, precise expectation for the event of application disappointments in advance can make strides the effectiveness of recouping the disappointment and application running.

II. RELATED WORKS

In any case, the LSTM based forecast strategies still have a few deficiencies. To begin with, the strategies as it were consider CPU utilization, memory utilization, cache memory utilization, cruel disk I/O time, and disk utilization as input highlights. More input highlights may assist increment the forecast precision. Moment, the LSTM based expectation show utilized single-layer LSTM development which can not handle numerous input highlights well compared to multi-layer development. Third, in the cloud information center, input highlights like CPU utilization and memory utilization are profoundly related over time. For a given time for forecast, the LSTM based expectation demonstrate continuously sets higher weights on the information closer to the time, and lower weights on the information advance absent from the time, with the suspicion that the information assist absent from the time continuously has lower affect to the expectation. Be that as it may, such settings cannot precisely reflect the affect degree as the assist information may still have higher affect on the disappointment (e.g., disappointments in long term

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305431 |

employments). The execution can be way better if the weights of information things are decided based on the genuine information follow. In this way, a unused expectation show is required to construct to actualize disappointment expectation in the cloud information center in arrange to accomplish superior precision in forecast.

Limited input features: Current LSTM-based prediction methods only utilize a narrow range of input features such as CPU and memory usage, restricting the model's ability to capture the full complexity of system behavior and potential failure indicators.

Single-layer LSTM construction: The use of single-layer LSTM limits the model's capacity to effectively integrate and process multiple input features, potentially overlooking intricate relationships and patterns within the data.

Temporal weight bias: Existing models assign higher weights to recent data under the assumption that it holds greater predictive value, disregarding the potential impact of historical data, particularly for long-term job failures.

Lack of adaptability: The reliance on predetermined weight settings fails to dynamically adjust to varying system conditions and evolving failure patterns, resulting in suboptimal performance and predictive accuracy.

III. PROPOSED WORK

High failure rates in large-scale cloud data centers, stemming from hardware and software failures, lead to task and job failures, impacting service reliability and resource efficiency. Existing machine learning and deep learning methods aim to predict these failures, yet accuracy remains a challenge. This paper proposes a multi-layer Bidirectional Long Short Term Memory (Bi-LSTM) algorithm to enhance failure prediction accuracy. The algorithm targets task and job failures, aiming to forecast their occurrence. Through trace-driven experiments, the proposed algorithm demonstrates superior performance compared to state-of-the-art methods, achieving 93% accuracy for task failures and 87% for job failures, addressing critical reliability concerns in cloud computing environments.

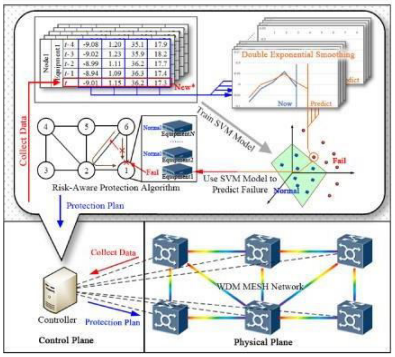

Figure 1 : System Architecture

3.1 Data Collection

The dataset comprises approximately 200GB of trace data stored in key-value pairs, loaded into a MySQL database for analysis. It includes a total of 555,555 tasks, with 50,000 designated for training and 55,555 for testing. Each task is represented by data from the first 100 time points, addressing the prevalence of longer tasks. Utilizing TensorFlow in Python on a GTX 1080 GPU, a Convolutional Neural Network (CNN) model is trained with 17 memory cells per layer. Training involves 5,000 steps per epoch over 1,000 epochs, with a batch size of 100 tasks per step, optimizing performance for accurate failure prediction.

3.2 Data Preprocessing and cleaning

For machine learning tasks, data processing involves employing Pandas Dataframe, facilitating efficient manipulation and analysis of the dataset. This includes tasks such as data cleaning, normalization, and feature engineering to prepare the data for model training. On the other hand, for deep learning tasks, Keras Categorical preprocessing is utilized to convert categorical data into numerical format suitable for neural network input. This involves techniques like one-hot encoding for categorical variables. Both approaches ensure that the data is appropriately formatted and ready for training, whether using traditional machine learning algorithms or deep learning models.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305431 |

3.3 Splitting The Dataset

The data is split into training and testing sets to assess model performance accurately. This is done using the train_test_split function from the scikit-learn library. The function randomly divides the dataset into two subsets, typically allocating a larger portion for training and a smaller portion for testing. By specifying the test_size parameter, the proportion of data allocated for testing can be controlled. This ensures that the model is trained on a sufficient amount of data while still having unseen data for evaluation. Splitting the data in this manner enables robust model assessment and validation, enhancing the reliability of the results.

3.4 MODEL EVALUATION AND PREDICTIONS:

In the final prediction phase, model performance evaluation metrics such as accuracy, precision, recall, and F1 score are compared across algorithms. The algorithm with the highest accuracy is selected to build a front-end Flask framework. In the Flask framework, users input testing data, and based on this input, the framework processes the data using the selected algorithm. The final outcome displayed to the user indicates whether the task or job is predicted to be failed or not based on the input data. This interactive and user-friendly interface enhances the practical utility of the failure prediction system in cloud data center management.

IV. RESULTS AND DISCUSSION

Here are some screenshots of the output:

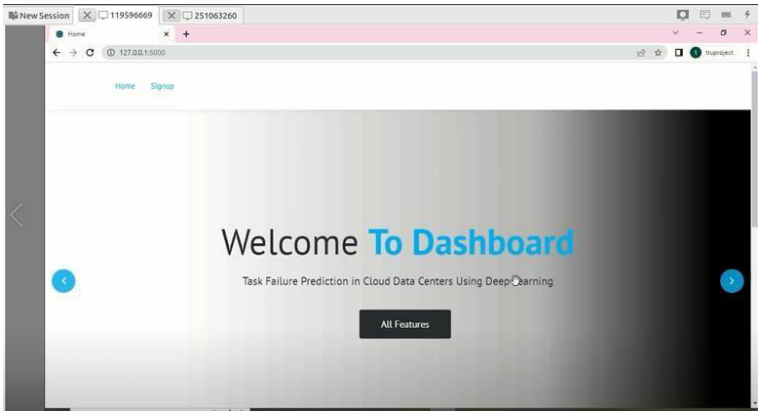

Figure 2: Home Page

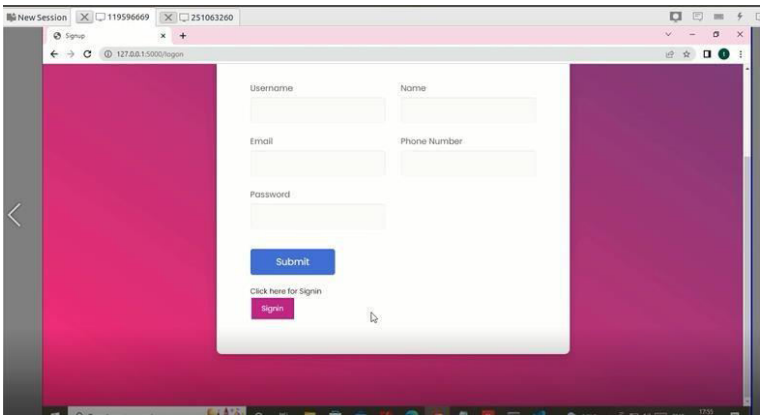

Figure 3: Registration Page

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305431 |

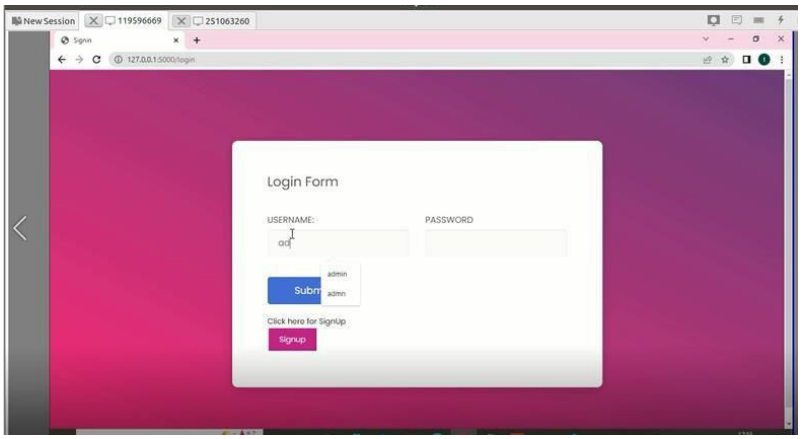

Figure 4: Login Page

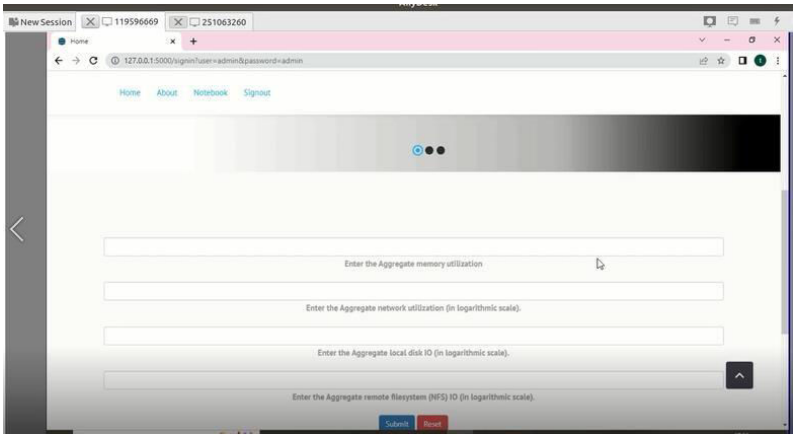

Figure 5: Input Data

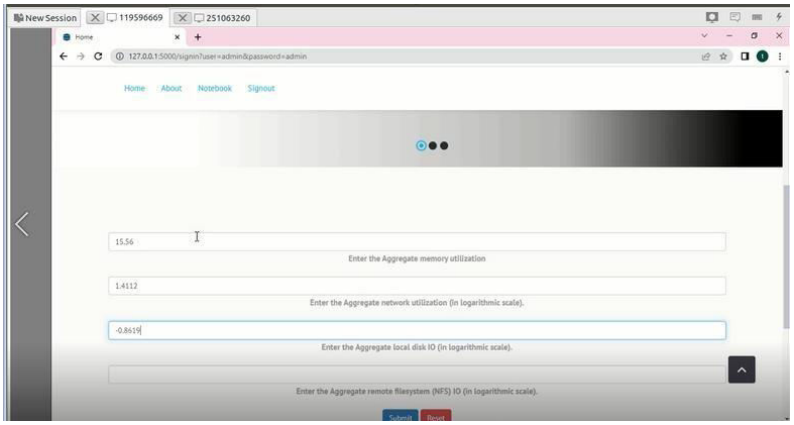

Figure 6: Upload Input Data

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305431 |

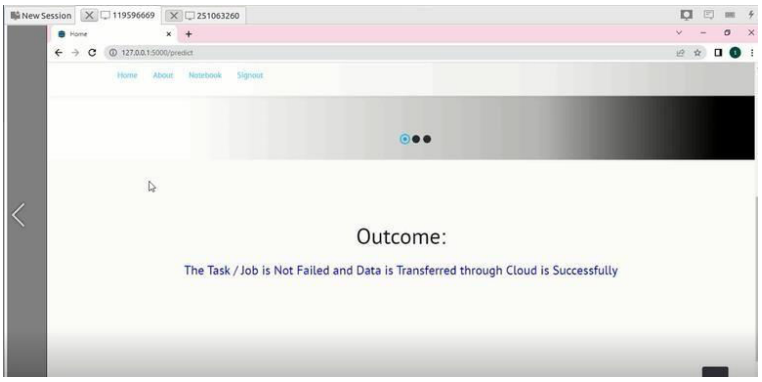

Figure 7: Output Screen

V. 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. Seleznjev, 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. Chalermarrewong, 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.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details