Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

A Dynamic, Novel Unified Methodology for predicting Cloud Workload on Cloud Computing

[1]M.S.Antony Vigil, [2]Akshat Soni, [3] Sachin Singh Kapkoti, [4] Shubham Shankar
[1]Assistant Professor, [2,3,4]Student Department of Computer Science and Engineering SRM Institute of Science and Technology, Ramapuram campus, Chennai, India

ABSTRACT

In cloud, good resource management is critical, and carrying out the prediction of workload is a crucial step toward accomplishing that goal. As we know that the workload of task which is being processed for a long time can be predicted by the recurrences of their past workload, whereas it is much complex job to predict the workload of the task which doesn't have a recurring pattern of workload. So, in this concept we are focusing on a method for workload prediction so the user/organization know when the workload is high or vice-versa. We have different approaches for workload prediction as in this,we use information about the workloads of a pool of tasks rather than the ancient workload for a task to predict the potential workload of that task, here we carry the knowledge about the workloads for a series of tasks to aid in the forecasting of new task workloads. As here we deal with the core of cloud computing model this is designed for carrying out the results in more efficient manner without any error in the process. It demands different factors for on and off premise of infrastructure for handling internet based applications. So in order to realize this concept, we have developed a clustering and an approach which is learning based. As we begin, the tasks are divided into several clusters. The approach Long short term which is the architecture of neural network will then be implemented to acknowledge the characteristics of every cluster's workload. Hence, the results are predicted by the core of the technology which is much faster and efficient with minimal errors. Workload prediction also depends on the popularity of an application's data.

KEYWORDS
AI-Artificial Intelligence; NN-Neural Network; Cloud computing

INTRODUCTION

Infrastructure as a Service comes under the deployment model of cloud computing which is the base of computation and application of other data intensives. They do not entertain any organization to interfere with their physical components such as performance, load, state etc. Due to the feature of multi-tenancy in the ecosystem of cloud, application output can be greatly influenced by other organizations, unknown and invisible processes (from the perspective of a specific user), the so-called background workload. So, in order depict the workload, Long Short Term Memory Algorithm is introduced in this project which is much more efficient than the existing one. As we know that Deep learning is the most crucial infrastructure of intelligence with modern computation, excels at predicting cloud workload for industry informatics. However, since deep learning models often involve a large number of parameters, efficiently training one is a difficult task. In this concept, we predict cloud workload for industry, an efficient model which is purely based on deep learning i.e. long short term memory algorithm is proposed. By the conversion of this weight matrices to the polybasic format of canonical form the parameter in the proposed model are substantially compressed. The parameters are also trained using an effective learning algorithm. Finally the proposed algorithm which is the architecture of deep learning is used to forecast virtual machine workloads in the cloud services. Experiments are then carried out on the data set obtained from Planet Lab to check if the output is genuine or not by comparing it to other machine learning approaches for virtual machine on the prediction of workload. The proposed model outperforms the deep belief machine learning based approaches in many aspects such as training efficiency and accuracy of workload prediction, demonstrating the model's potential to achieve predictive services for industry with cloud services.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

LITERATURE SURVEY

Several basic auto-encoders are collected together into the deep learning architecture, generally known as stacked auto-encoder model. Moreover, a regression layer for the directed training is normally added to the topmost layer for the purpose of differentiation. The above model is usually trained using two stages. In initial stage we obtain the required parameters by using supervised strategy to conduct up to down propagation .Classification accuracy drop is the basic error that we get while comparing with old deep learning training models.And the term reduction of parameters denotes the change between compressed parameters and the old parameters and several other comparison factors were used to differentiate between training processes Workload in a cloud industry can be plotted with the help of graphs. There are many advantages of the new algorithm introduced in this system i.e. Improved energy structure prediction model, Rationality of the structure prediction is greatly improved, High feasibility of the model, Radial basis function is applied, Ease of transferring knowledge from one model to another based on domains and tasks, the better use of unstructured data used while training, and the capability to deliver high quality products. There are different technologies used in this system to achieve the outcome which are Python, Numpy, Sci-learn, Tensor Flow & Keras, and Jupyter Notebook which are backend technologies and some other technologies such as Web Technologies, Bootstrap etc which falls under frontend technologies.

Table 1. Algorithm Comparison

S. No.	Existing System	Drawbacks of Existing System.	Advantages of Proposed System
1	Savitzky-Golay filter and wavelength decomposition.	Data is abnormal when the ratio is above or below than some feature thresholds.	Our model provides high robustness and security.
2	This system uses k-clustering prediction algorithm.	It has high prediction complexity with higher dimensions which leads to loss of information	No information is lost during decomposition and reconstruction
3	This system uses Column Generation (CG) Technique to handle optimization problem.	Not based on real time datasets	Our model handle's real time datasets.
4	This system converts the weight matrices to canonical format and using deep learning models.	It has poor application performance.	Rationality of application structure prediction is greatly improved

PROPOSED METHODOLOGY

The concept is to get an idea to identify method to predict work pattern of the new tasks.By picking out selected tasks from work history from various online servers, we create a pool of tasks and thus in the process helping the training model to come out with results. But even the task knowledge sometimes is not enough to successfully improve and improvise the prediction of the different workloads.It includes practical data-oriented applications that make use of a variety of resources while running. The source codes for different cloud service providers have been customized and are publicly accessible. We are certain that it indicates new function implementations with reduced experiment setup overheads. We now present the implementation of cloud workload prediction model.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

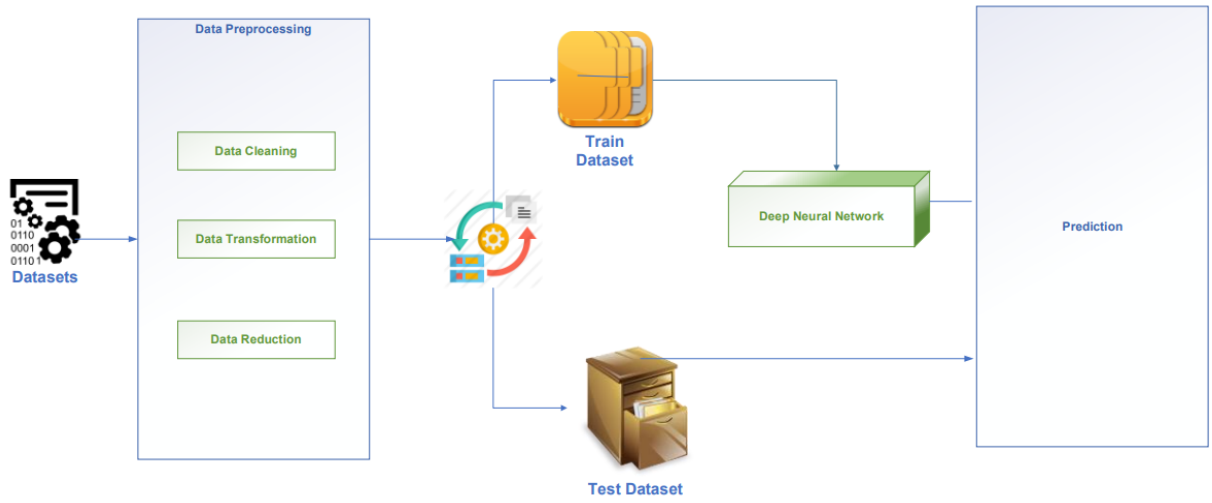

Figure 1. Architecture diagram.

Here we have introduced the method of prediction using the NN model and by using its result we then assessed its correctness for future workload prediction using traces of requests to different online servers. The proposed algorithm is long short term algorithm (LSTA) which is a type of recurrent neural network which have the ability to learn order dependence and are used in prediction problems. This algorithm carries out smooth and efficient results from the user input data and predicts the workload of a cloud system without any means of dispute and harm to the organization. Hence, it is a better approach then the algorithm used earlier i.e. deep belief algorithm which doesn't give promising results as compared to this system. Long short-term memory (LSTM) is a deep learning architecture that uses an artificial recurrent neural network (RNN). LSTM networks are very helpful in the process to classifying, analyzing, and doing predictions on the basis of time series data because there can be delays of unknown period between significant events during the given time. The main feature of LSTM is its cell or in other words cell state which provides memory to the module so it remembers the earlier inputs. Cell state is main feature to remember eg.1) the horse which ran…..was fast. 2) The horses which already ran….were fast. In the sentence the LSTM remembers he feature 'was'. In second sentence 'were' is remembered as subject is plural. In this case cell state is singular/plural statements. We have three gates is LSTM i.e. Input, Output, Forget. Input gate helps us to understand which new information the model is going to store. Output gate is mainly used to yield final output on information gathered. Forget gate is used to tell which information is not needed. For example [8, 9, 2, 1] the answer to number of time stamps would be 4. We can generalize all these 4 functions:- Sig/tanh(W.x+U.h+b) Where b,U,W are parameters where h represents the hidden layer values and the value ***x*** is the input parameter which is calculated using the values of above given parameters.

Data Evaluation

Exploratory data analysis depends largely on the factor of graphical interpretation and as well as graphical visualization. Although statistical modelling gives us a uncomplicated low dimensional description of connection among different variables. It typically requires statistical and mathematical knowledge. As we know for a fact that graphs and data visualization are usually better interpretative and simple to create, with their help you can quickly survey a data set in various aspects. Our main aim is to create a simple brief of data that will help you answer your query (s). It's not the final step in the data science process, but it's still critical. The graphs generated by EDA are not the same as the final graphs. Over the process of evaluating a dataset, you'll likely produce thousands, if not hundreds, of exploratory graphs. One might end up publishing of these graphs in their final form. Since one of data, all of the graphs and code should represent that aim. An exploratory graph does not include important details that would be included in a published graph2. Exploratory data analysis is a technique for analyzing data set and

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

determining the underlying law based on how the data is distributed. Exploratory Data Analysis (EDA) is known as a form of data analysis that employs visual techniques to reveal the data's structure. It works similar to like a human brain works while exploring a data set. It uses visuals methods to reveal the data structure. And visual analysis is the use of a number of different models in the analysis ability of a unique method of presenting data. Analysts often conduct exploratory data analysis on data until settling on the mode of structure quantity; exploratory data analysis may also expose surprise variations that a conventional model cannot. Exploratory Data Analysis' main feature is flexibility with which it applies to data structure and with which it reacts to the new discovered mode of the subsequent analysis.

Feature Engineering

Information gain is the amount of contribution when a given term in data set is present or not it is a calculation of difference entropy to determine whether the text is present or not. It can also reach its maximum limit sometimes depending on the words present in the document .The IG uses various measures and selectors to classify features in the document. The first is Distinguishing feature selector this feature when encounters a distinctive feature in the document gives it a high score and likewise when it finds a common feature it gives it a low score. Another method used is Ambiguity feature is a type of feature selection method used to define uncertainty of a feature in the document, words that appear constantly will receive a rating close to 1, and other words that are not as recurring receive a rating less than 1.Using this a threshold is taken and features are then filtered using a threshold value. The features having the ambiguity score lesser than the selected threshold value are neglected and features having score higher than threshold value are used in the learning phase .Characters with most beneficial information gain are likely to be selected as a feature. The reduction of high dimensional data is main use of these methods. By using the above given selection methods it is made sure that only the relevant information from the text is selected for further processing which in turn helps to reduce complex and higher dimension inputs data, while preserving information to further process the data. The data separated from feature pre-processing is called subsets. There are various types of subsets based on their impact on learning accuracy. The subset with the most learning accuracy is generally chosen.

Model Prediction & Evaluation

The algorithm uses propagation neural network on supervised deep learning methods. These neural networks are generally made of neuron types elements known as nodes which are arranged in layers. The transmission of input information takes place with the help of layer known as activation function and then after passing it reaches output nodes. In the beginning the various connecting layers are initialized with many small random data to check the processing and distinguishing ability. The output node then provides the result. In input layer data fed is separated into various subsets relevant, irrelevant or redundant which reduces the complexity of input data and makes it easier for the algorithm to learn the interconnection. It is connected to output layer via neuron. Between neurons in the same layer there are no couplings. If the output layer fails to come up with the probable outcome the whole process is started again. The data is divided into small subsets by various distinguishing methods. From the input data one part is taken for training set and other for test set. The amount of data depends on the evaluation of prediction model.

EXPERIMENTAL RESULTS & DISCUSSION

The first step to implementing deep neural network involves including all the necessary header files like Num-Py, panda and TensorFlow to work with our dataset. After this we write the below code to read the dataset.

```
df = pd.read_csv("Dataset.csv")
df.head(5)
df.sample(10)
df.tail(5)
```

In the above code head is used to show the first five rows of the dataset, sample is used to show any random row in the dataset and last tail is used to show last five rows of the dataset.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

	#VALUE!	CPU cores	CPU capacity provisioned [MHZ]	CPU usage [MHZ]	CPU usage [%]	Memory usage [GB]	Memory capacity provisioned [KB]	Disk read throughput [KB/s]	Disk write throughput [KB/s]	Network received throughput [KB/s]	Network transmitted throughput [KB/s]	Label	Memory usage [KB]
2693	1378741183	4	11703.99824	10816.445040	92.416667	2.639613	67108864	8.466667	9002.666667	0.0	1.933333	1	2639612.8
1631	1377785292	4	11703.99824	68.273323	0.583333	0.000000	67108864	0.000000	0.600000	0.0	1.000000	0	0.0
1413	1377588776	4	11703.99824	74.125322	0.633333	0.000000	67108864	0.000000	0.600000	0.0	1.000000	0	0.0
2820	1378855494	4	11703.99824	72.174656	0.616667	0.000000	67108864	0.000000	1.000000	0.0	1.000000	0	0.0
2694	1378742084	4	11703.99824	10244.899793	87.533333	3.221222	67108864	2211.800000	5732.333333	0.0	1.333333	1	3221222.4
1360	1377541073	4	11703.99824	81.927988	0.700000	0.000000	67108864	0.000000	0.600000	0.0	1.000000	0	0.0
2010	1378126422	4	11703.99824	74.125322	0.633333	0.000000	67108864	8.466667	3.466667	0.0	1.066667	0	0.0
1284	1377472367	4	11703.99824	81.927988	0.700000	0.000000	67108864	0.000000	0.666667	0.0	1.000000	0	0.0
1054	1377264749	4	11703.99824	70.223989	0.600000	0.000000	67108864	0.000000	1.000000	0.0	1.066667	0	0.0

Figure 2. Random ten rows from the dataset.

The above figure contains the parameters for predicting workload. Based on the all parameters we are giving label to every row whether the workload is high or low. If the value of the label is '0' then workload is low that means there is no problem in the cloud whereas when the workload is high then we show '1' in the label. Based on the label we show the pie chart below.

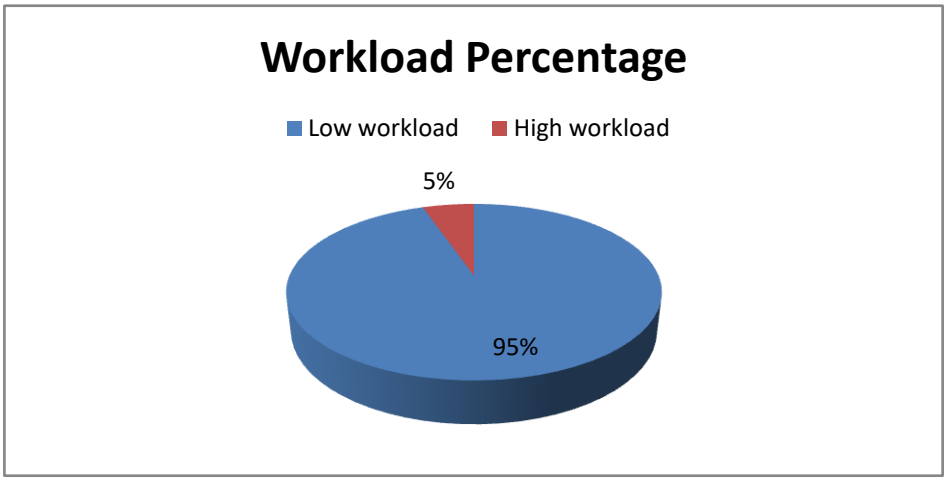

Figure 2. Workload Percentage Based On Label.

go.Figure(data=[go.Pie(labels=df["Label"].value_counts().index, values=df["Label"].value_counts().values)])
This set of code is used to create a pie chart based on the presence of 0's and 1's in the label column.

train_p, test_p, train_q, test_q = train_test_split(P, Q, test_size=0.01, random_state=415)
Here we split out dataset into two sets i.e. training set and test set. Training set is used to train our model. Based on the training we test our model using test set. The size of training set is 90% of the overall dataset and the remaining is our test set to check whether our model is predicting the required output.

sess.run(optimizer, feed_dict={input_data: train_p, output: train_q})
predict_q = sess.run(q, feed_dict={input_data: test_p})

Now after training of the model it's time for prediction. The code we wrote above is used to predict the workload of the cloud based in the dataset. After prediction we get two things accuracy and mean square error. At last we plot a graph for the mean square error with respect to iterations and accuracy with respect to iterations.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

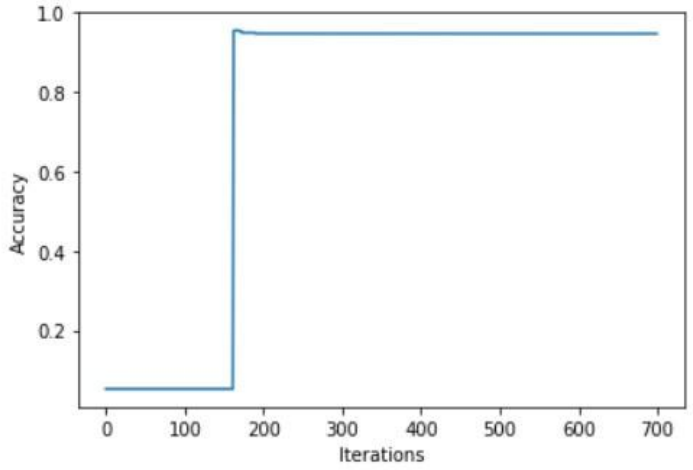

Figure 3. Accuracy

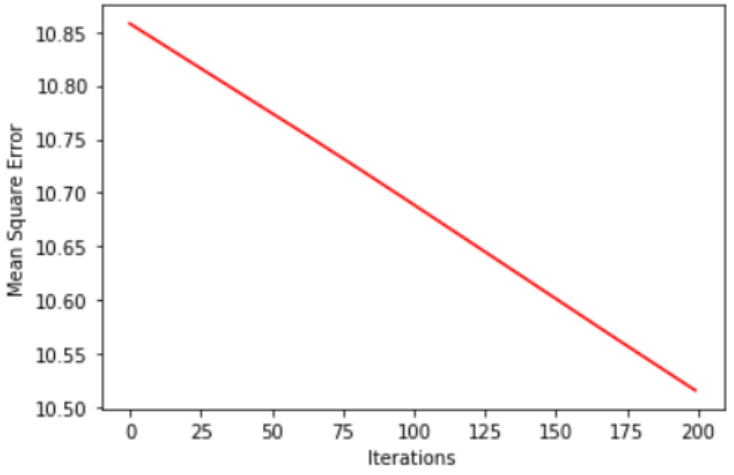

Figure 4. Mean Square Error

Now after implementing the whole set of code the accuracy we get is 93.10%.

CONCLUSION

Here a dedicated hybrid cloud computing model design is proposed to manage the workload in various platforms. The new architecture minimizes the load on hosting cloud servers, and websites with the proposed workload management technology. And it can also give firms or companies the ability to manage the daily workload on various resources in an efficient manner and manages flash crowd peak load for servers. As we know cloud is known for its elastic services and scalability of various infrastructures resources which are otherwise not easily scalable. In future we plan to develop a double line protection against QoS by handling errors in the prediction if the system and provide better compatibility for cloud providers as well as the users.

REFERENCES

1. A. Zhao, W. Dong, G. Chen, G. Min, T. Gu, and J. B u,“ Embracing corruption burstiness: ZigBee error recovery under Wi-Fi interference is quick ”, 2019.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4857 - 4863
Received 25 April 2021; Accepted 08 May 2021.

2. D. Bhuiyan, G. Wang, J. Wu, and J. Cao, “Wireless sensor networks provide reliable structural health monitoring”, 2018.
3. E. Zhang, L. T. Yang, Z. Chen, P. Li, and F. Bu, “ A crowdsourcing to cloud computing adaptive dropout deep computation model for industrial IoT big data learning” , 2018.
4. G. Chen, A. S. Ganapathi, R. Griffith, and R. H. Katz, “Analysis and takeaways from a Google cluster trace that is publicly available”, 2017.
5. G. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, “Framework for energy-efficient resource allocation and provisioning in cloud data centres”, 2019.
6. H. Yhang and Z. Chen, “ For clustering incomplete big sensor data, a distributed weighted possibility c-means algorithm was developed”, 2018.
7. L. Zhao, W. Dong, J. Bu, T. Gu, and G. Min, “For bulk data dissemination in low-power wireless networks, accurate and generic sender selection is required”, 2019.
8. L. Yi, X. Deng, M. Wang, D. Ding, and Y. Wang, “ Hole Detection in the Internet of Things for Radioactive Pollution Monitoring with Localized Confident Information Coverage”,2017.
9. M.Joseph AD, Katz RA, Konwins A, Le G, Patterson E, Rabin A. “A view of cloud computing”, 2018.
10. Q. Zhag, L. Cheg, R. Bouaba, “ Cloud computing: state-of-the-art and research challenges”, 2019.
11. R. Yag, I.T. For, J.M. Sopf, “In international parallel and distributed processing symposium”, 2016.
12. S. Kodo D, Crne W. “The Google compute cloud employs a Bayesian mode to forecast host load”, 2017.
13. X. Song, Y. Yu, Y. Zou, Z. Wag, S. Du, “ Predicting host load in cloud computing using a long short-term memory”,2019.
14. Y.Akioa S, Muaoka Y. “On the computational grid, a long-term forecast of CPU and network load is available”, 2019.
15. Z.Huag J, L i C, Y u J. “Resource prediction based on double exponential smoothing in cloud computing”, 2018.