Computational Grid CPU Burst Time Estimation Using Machine Learning Algorithms

Aseem Sharma
PID: 6329681

Ashfaq Ali Shafin
PID: 6265743

Kratin Tiwari
PID: 6329698

Abstract

We are recommending Machine Learning (ML) based method to forecast CPU burst time in Computational Grids (CG). Our work is an extension of [3] regarding dataset size, different ML algorithms, and the introduction of Artificial Neural Networks for estimation of the burst length of CPU.

Introduction

CPU burst time is the time spent in execution for a process. Many process scheduling algorithms such as Shortest Job First (SJF), Shortest Remaining Time First (SRTF) require the former information of process burst time before execution [1]. Different modified and improved versions of the round-robin algorithms also need to know the largest and smallest process burst time so that the time quantum or time slice can be modified dynamically instead of using static time quantum.

Grid Computing is a collection of interconnected supercomputers that collaborate as a virtual computer to handle enormous tasks like data analysis and weather forecasting. By dividing jobs among numerous machines, the processing time is greatly decreased, increasing efficiency, and reducing waste [2]. An additional dedicated machine in the grid is responsible for distributing and assigning the resources to the processes [4, 5] are greatly helped by process burst time if scheduling algorithms require CPU burst time.

A machine learning-based approach to estimate CPU burst time was implemented using a subset of data from the “GWA-T-4 AuverGrid” computational grid dataset [3]. Our implementation on the prediction of process burst time will be more extensive than [3]. Helmy et al. [3] used only the first 5000 jobs to estimate the CPU burst time and no explanation was provided for not using the entire dataset with 414,176 jobs. Our research project will try to answer the following research questions:

RQ-1. Does the performance increase or decrease when the machine learning algorithms include all the jobs as input data?

RQ-2. Does normalizing job features impact the accuracy of the prediction of CPU burst time?

RQ-3. Do other machine learning algorithms perform better than the implemented machine learning algorithms in [3]?

RQ-4. Do Artificial Neural Networks (ANN) perform better than conventional machine learning algorithms?

Related Work

SJF uses simple averaging or exponential averaging to calculate CPU burst time. In simple average, the average of n-1 process burst time is calculated to estimate the n-th process burst time. However, the exponential average uses a smoothing factor of ‘alpha’ and previous CPU burst time to calculate the current process burst time [1].

Helmy et al. [3] proposed a machine learning approach to estimate process time in CGs. The authors applied Support Vector Machine (SVM), K Nearest Neighbors (KNN), Decision Tree (DT), and Multi-layer Perceptron (MLP) machine learning algorithms using the first five thousand CG job data from the GWA-T-4 AuverGrid dataset [6]. The proposed approach gained the best performance utilizing historic features and using the K-NN algorithm with an absolute error rate of 3.46%.

Another way to forecast the length of the next CPU burst was proposed by Pourali et al. [7], which was based on a fuzzy system as a knowledge-based rule system [8]. The input of the system was the value of the previously used bursts time of a process.

Process execution data from the past can be used to anticipate process run times in the future. Smith et al. [9] suggested a method for calculating run times based on comparable historical data. The combination of two search methods - greedy and genetic - to discover the features that produce the best description of comparison was the new component of that study.

Matsunaga et al. [10] illustrated and compared the applicability of several machine learning approaches for forecasting the geographical usage of resources by programs. They were forecasting runtime, memory, and disk needs for bioinformatics programs as part of their research. SVM and K-NN were employed as classification and regression models in that investigation.

Proposed Work

Our proposed work will be based on the “GWA-T-4 AuverGrid” dataset. The dataset consists of 414,176 jobs. AuverGrid is a five clustered production grid where each cluster is composed of dual 3-GHz Pentium 4 Xenos nodes located in France. First, we will analyze the dataset and the features of the dataset to preprocess. Our preprocessing phase will go through the following steps.

1. Feature Set Reduction: Some of the features from the dataset may not contribute to predicting the process burst time. We are going to reduce the feature set by removing the insignificant features.
2. Feature Normalization: Normalizing the feature values to get similar distribution among all the features. Normalizing the features play a key factor in many machine learning models [11].
3. Training and Testing Dataset: We will divide the dataset into two parts one for training and another for testing. The ratio of training and testing data will be 70:30. The training dataset will then go through 5-fold cross-validation in the training process. Based on the cross-validation results we can modify our feature set to get improved results from the test dataset. We will also take the first five thousand jobs from the dataset to compare our model with Helmy et al [3].

Second, we will employ several machine learning algorithms like linear regression, Support Vector Machine, Decision Tree, Random Forest, AdaBoost, XGBoost to train our model. We will also investigate our data with Artificial Neural Network (ANN) models. We will use Python as our programming language and Scikit-Learn machine learning library. For the ANN algorithms, the TensorFlow library will be utilized. Based on the results from the training phase, we may remove the insignificant features from our feature list to get a better result in our testing dataset.

Finally, we will test our machine learning models on the test dataset and compare the result with Helmy et al [3]. We will also keep track of the time consumed by each model so that we can propose an efficient machine learning model.

Evaluation

Our evaluation metric will be the accuracy of prediction on the testing dataset. This will be represented in tabular format as well as column graphs wherein the X-axis different machine learning algorithms and in the Y-axis the accuracy percentage will be shown.

Another column graph will show the time consumed for each machine learning model. A table will compare the relative absolute error rate of our implemented machine learning algorithms with [3].

Timeline

Our proposed project will span ten weeks in total and timeline is discussed in the following table.

Project Week	**Course Week**	**Member 1 Aseem Sharma**	**Member 2 Ashfaq Ali Shafin**	**Member 3 Kratin Tiwari**
1	6	“GWA-T-4 AuverGrid” dataset anazlying.	Analyzing the feature sets for the machine learning algorithm	Keeping up with Artificial Neural Network, searching other datasets like “GWA-T-4 AuverGrid”, and analyzing the dataset.
2	7	Listing insignificant features from the dataset and Pre-processing the dataset for training.	Generating statistics of the dataset and representing them in a tabular format.	Generating graphs for visualization of the dataset.
3	8	Preparation for Mid-Term.		
4	9	Applying Machine learning algorithm on first 5,000 jobs.	Applying Machine Learning Algorithm on the whole training dataset.	Applying Artificial Neural Network on the training Dataset.
5	10	Evaluation of the machine learning model on the first 5,000	Testing the machine learning models with test	Testing the ANN models with test dataset and

		jobs and contributing to the Project Report.	dataset and contributing to the Project Report.	contributing to the Project Report.
6	11	Comparing the result with Helmy et al [3].	Generate graphs and tables based on the ML output result.	Generate graphs and tables based on the output results of ANN algorithms.
7	12	Contribute to the draft paper based on the previous work.	Contribute to the draft paper based on the previous work.	Contribute to the draft paper based on the previous work.
8	13	Catch Up Session		
9	14	Modification of the paper based on the review.	Modification of the paper based on the review.	Modification of the paper based on the review.
10	15	Paper Presentation		

Team Composition and Responsibilities

Student Name	Student Status	Skills for the Project	Responsibilities
Aseem Sharma	Masters in CS (2nd year)	Python. Machine Learning	He will be analyzing the dataset, pre-processing for the ml model, and comparing our work with Helmy et al [3].
Ashfaq Ali Shafin	Ph.D. in CS (2nd year)	Python. Machine Learning. Data Analysis.	He will be responsible for looking over the machine learning part of the paper. I will simulate different ml algorithms and record the output results.
Kratin Tiwari	Masters in CS (2nd year)	Python. Tensor Flow. Artificial Neural Networks	He will build the Neural Network for the project. I will also generate graphs and tables based on the result found in our ANN model.

References

[1] A. SILBERSCHATZ, P. GALVIN and G. GAGNE, *Operating System Concepts*, 8th ed. 2013.

[2] “What is grid computing?”, *azure.microsoft.com*, 2022. [Online]. Available: https://azure.microsoft.com/en-us/overview/what-is-grid-computing/. [Accessed: 10- Feb- 2022].

[3] T. Helmy, S. Al-Azani and O. Bin-Obaidellah, "A Machine Learning-Based Approach to Estimate the CPU-Burst Time for Processes in the Computational Grids," *2015 3rd International Conference on Artificial Intelligence, Modelling and Simulation (AIMS)*, Kota Kinabalu, Malaysia, 2015, pp. 3-8.

[4] S. Mehmood Shah, A. Mahmood and A. Oxley, "Analysis and evaluation of grid scheduling algorithms using real workload traces", *Proceedings of the International Conference on Management of Emergent Digital EcoSystems - MEDES '10*, Bangkok, Thailand, 2010.

[5] D. Fernandez-Baca, "Allocating modules to processors in a distributed system", *IEEE Transactions on Software Engineering*, vol. 15, no. 11, pp. 1427-1436, 1989.

[6] “GWA-T-4 AuverGrid”, *Gwa.ewi.tudelft.nl*, 2022. [Online]. Available: http://gwa.ewi.tudelft.nl/datasets/gwa-t-4-auvergrid.

[7] A. Pourali and A. M. Rahmani, "A Fuzzy-Based Scheduling Algorithm for Prediction of Next CPU-Burst Time to Implement Shortest Process Next," *2009 International Association of Computer Science and Information Technology - Spring Conference*, Singapore, 2009, pp. 217-220.

[8] B. Kosko, *Neural networks and fuzzy systems*. Englewood Cliffs: Prentice Hall, 1992.

[9] W. Smith, I. Foster and V. Taylor, "Predicting application run times with historical information", Journal of Parallel and Distributed Computing, vol. 64, no. 9, pp. 1007-1016, 2004.

[10] A. Matsunaga and J. A. B. Fortes, "On the Use of Machine Learning to Predict the Time and Resources Consumed by Applications," *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, WA, USA, 2010, pp. 495-504.

[11] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance", *Applied Soft Computing*, vol. 97, 2020, pp. 105524.