School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Internet: office@ecs.vuw.ac.nz

Proposal for Using a Transformer Model to Predict Time Series Cloud Workload

Joel Chu

Supervisor: Hui Ma, Aaron Chen

Submitted in partial fulfilment of the requirements for
Software Engineering with Honors.

Abstract

The aim of this project is to accurately predict cloud workload using a transformer model. Doing this will allow cloud service providers to accurately determine the amount of resources to allocate at any given time. Thus preventing under and over provisioning of resources. The datasets google cluster trace and Alibaba trace will both be used as training, testing and validating datasets. The transformer model will be compared to other time series workload prediction models such as LSTM. This project is successful if the transformer model has a greater prediction accuracy than traditional models. A decrease in prediction and training will be a bonus.

Contents

1 Introduction

In cloud computing, the user will pay for a service offered by a Cloud Service Provider (CSP). The CSP is responsible for maintaining the quality and reliability of the service they are providing to the user. The user must always be satisfied by the responses time, cost and quality that the CSP is providing. This is known as quality of service (QoS). This will mean making sure that the user is always receiving sufficient resources for all times of the day. This is a challenge for the user and CSPs because throughout a day when the user requires more resources during peak hours, demand may increase where they will require more resources which has to be allocated [2]. As well as providing QoS, the CSP must also take into account the cost factor. There will be financial consequences to the end user if they allocate too many resources so it is key to find the optimize sufficient resources as well as minimizing costs[1]. CSPs tend to use offer a service elasticity service which allows the service to allocate and de-allocate resources based on the fluctuation demand during peak times. This will provide end users with QoS and can improve response times, availability and throughput to the user [10].

In the past, methods to dynamically allocate resources have been based off a reactive approach which will allocate according to the current demand [12]. The problem with allocating reactively is that there will me moments where demand is greater then the current resources allocated, thus not providing QoS to users. A better method would be to predict the future workload then allocate according to the future demand. For a CSP to be able to predict the resources required for the required workload, they will need an appropriate model to make this prediction. They can then allocate and de-allocate resources based on the prediction. In the past, models have used time series analysis to make these predictions[2][6]. These models work by analysing past workload and identifying patterns to be able to make future workload predictions. Long term sort memory (LSTM) and Autoregressive Integrated Moving Average (ARIMA) are models that have previously been used to make for time series predictions. Qihang Ma used the two models for stock price prediction [6] as well as Calheiros et al who has used an ARIMA model for dynamic resource provisioning predictions[2]. The results of these studies and others will be discussed in the later sections.

1.1 Motivation

We want to build a model that can make accurately predictions for cloud workflow. The motivations for this project will be listed below.

- In a survey by RightScale, they found that 94 percent of businesses used some form of cloud [13]. This number is substantial because an improvement in workload prediction could potentially have a positive impact on almost every business. Even a small improvement in prediction accuracy will be a large improvement due to the scale of cloud.

- The incentive to do this reaches an economic and environmental level. Since cloud computing uses a large amount of physical resources to run, if we are able to reduce wasted resources, it would mean that large data centres can save energy. By accurately predicting cloud workload, it will reduce the impact cloud computing has on the environment by saving resources and energy thus benefiting the whole world.

- If cloud workflow accuracy reaches a high enough accuracy, we will be able to avoid under provisioning resources to users. This would improve the overall user experi-

ence as CSPs can constantly maintain QoS for all users. Thus maintaining good performance and availability for all users.

- Since over provisioning can be reduced, this will reduce operating costs in general. End users will also feel the effects of this by not having to compromise the QoS provided by CSPs while being able to save money on services.

1.2 Aim

The aim of this project is to build a model that can accurately predict future cloud workload. The model will be able to analyse and train on historical time series data to make accurate predictions. We aim to achieve a greater accuracy than existing workload prediction models. This will be explained later in this proposal.

1.3 Long Short Term Memory (LSTM)

LSTM is a time series prediction model that is a kind of recurrent neural network (RNN). LSTM works by remembering and forgetting long and short term memory. The input will determine how much of each long and short term inputs are predicted[6]. An advantage of LSTM is that it can accurately make term predictions by analyzing short term and long term data where other RNN based models will only be able to make accurate predictions based on short term data[10].

What are the disadvantages?

- LSTM models require a lot of input data to be able to make accurate predictions. When there are not many training points, the LSTM will not make accurate predictions as seen by Jiechao [3]

- As noted in the study "Attention is all you need", LSTM as well as other RNNs, they rely on long sequences of computation which can not be done in parallel[8]. This will mean that computation time will be very large with large data sets that have long sequences.

1.4 Autoregressive Integrated Moving Average (ARIMA)

ARIMA is a model that is used to generate short-term forecasts on time series data. ARIMA uses previous data points to make forecasting predictions [7]. A trait of ARIMA is that it does not need to assume any underlying principles or model equations for the context of the data. This can mean it will not need to assume any basic economic terms for stock market prediction because the model should be able to learn the principles through training on the data [6]. Due to this, this means the model can be used for different time series based data sets but can still be used effectively. The downside to this is that these basic principles may mean that the training process does not allow ARIMA to pick up these these underlying principles which may make for inaccurate predictions.

What are the disadvantages?

- Different traditional ARIMA models may not perform as well as others so it will depend on the forecaster to pick the best model for the task [7]. Since the predictor may not always have a vast knowledge and be experienced with ARIMA which can cause the model to not be reliable.

- ARIMA works by making a linear prediction from the input variables and error terms. This linear model will work well for short-term predictions but may not perform as well for more complex non-linear problems [6]. Since we do not know the relationship of the data set we are using, we will not know if the linear prediction of the ARIMA model can accurately forecast jobs in cloud workload.
- In the study by Calheiros et al, they found that ARIMA has a very large computation time when the input data set is large [2]. The data set we will use for training will be large which could be a downfall for ARIMA.

1.5 Transformer

The transformer model was first proposed in 2017 as a model that could address the shortcomings of recurrent neural networks, long short-term memory and gated recurrent neural networks[8]. The transformer model works around the idea of improving existing recurrent models that have a high computation cost. Recurrent models generates sequences to hidden states which makes parallelization impossible due to the sequential nature of the model[9]. When parallelization is not available, training recurrent models can get very expensive and time consuming. Although certain factorizations and improvements have been made to previous models, recurrent networks would ultimately still be sequence based which these improvements could never address [4]. Although transformers are mainly known for their use in NLP, speech processing and computer vision. The transformer has seen large success in these areas but not much work has been done in time series data prediction. Transformers use an attention mechanism which will allow it to capture long and short term patterns regardless of the distance. This is something that LSTM struggles to capture especially with historical long term dependencies[5]. The ability for transformers to capture patterns regardless of the distance makes it suitable for predicting cloud workloads. Due to the how new transformers are, they are not commonly used for time series data but has been used in the past for forecasting but never cloud workload forecasting[5][11]. The results from the previous forecasting attempts will give us insight when creating the transformer model.

2 The Problem

The aim of the project is to create a transformer model that can accurately predict cloud workloads. The transformer model will have to be tuned and adapted to work for the input time series data.

3 Proposed Solution

In this project, a transformer model will be used to predict cloud workload. A transformer model has proven to be successful when implemented in other areas. Implementing a transformer model has the possibility have an improved accuracy compared to existing models such as ARIMA and LSTM.

3.1 Transformer Model

The transformer model works by encoding the input vector to an output vector. The decoder will then take this vector and generate an output vector for the model, each time, using the output from before to influence each value in the vector[8]. Transformers use attention which is different to regular recurrent networks.

3.2 Attention

Attention function is what separates transformer models to other models. This gives it the weighted sums of each output vector. The output is made up of key-value pairs for each output. The output is the weighted sum of all values which is how each output has able to relate to each output. This attention builds the relationship for all output values. Using attention will speed up computation time by splitting up sequential operations. According to the article "Attention is all you need", recurrent neural networks have a sequential operation of O(n) with a maximum path length of O(n). Self-attention on the other hand has a sequential operation of O(n) and a maximum path length of O(1). This difference will mean that computation can be done paralleled when using the transformer model. Scaled dot-product attention and multi-head attention are also components that are used by a transformer model.

- Scaled dot-product attention is used on all the keys, It will then go through a softmax function to output the weights if the values. The dot-product will be scaled by dividing by the square root of the query. This will prevent large values from having too great of an impact on the output.
- Multi-head attention is when single attention is projected linearly in the value, key and query dimension. Doing this will allow the information to take into account different dimensions which can not be done with single-attention.

3.3 Model Architecture

The transformer being made for the project will include the basic components such as encoding, decoding and self-attention. Alterations will have to be made such as the number of layers and other parameters. These alterations will be determined by the input data as well as which combination of parameters give the best prediction accuracy.

4 Evaluating your Solution

To test the effectiveness of the model, we will use some training data set to train the models then compare the accuracies to the LSTM model as it is currently one of the best predictors time series based workload.

We will use a variety of metrics and measures to measure the performance of both of these models. The aim for the transformer model is to obtain a higher prediction accuracy than the LSTM model. We will be using the metrics mean absolute error (MAE), root mean squared error (RMSE) and mean squared error (MSE) to measure the performance of the two models. These metrics were used to evaluate the performance in of the LSTM trained by Mahendra so we will use the same metrics to measure the accuracy of the transformer model [9]. We will also be measuring how long the computation takes as the prediction will not be effective if it takes too long to output regardless of the accuracy.

4.1 Dataset

Google cloud trace and Alibaba trace are both datasets that contain their annual cloud workload in the form of a dataset. Both of these datasets will be used for training and testing the models to determine the efficiency of the transformer model. Since both datasets have different inputs and behaviours, the models will have to be changed to adapt to each of these datasets.

4.2 Mean Absolute Error (MAE)

The mean absolute error is a measure of the magnitude of the difference between the observed and actual results. This is a measure of accuracy in a sequence of prediction.

4.3 Root Mean Squared Error (RMSE)

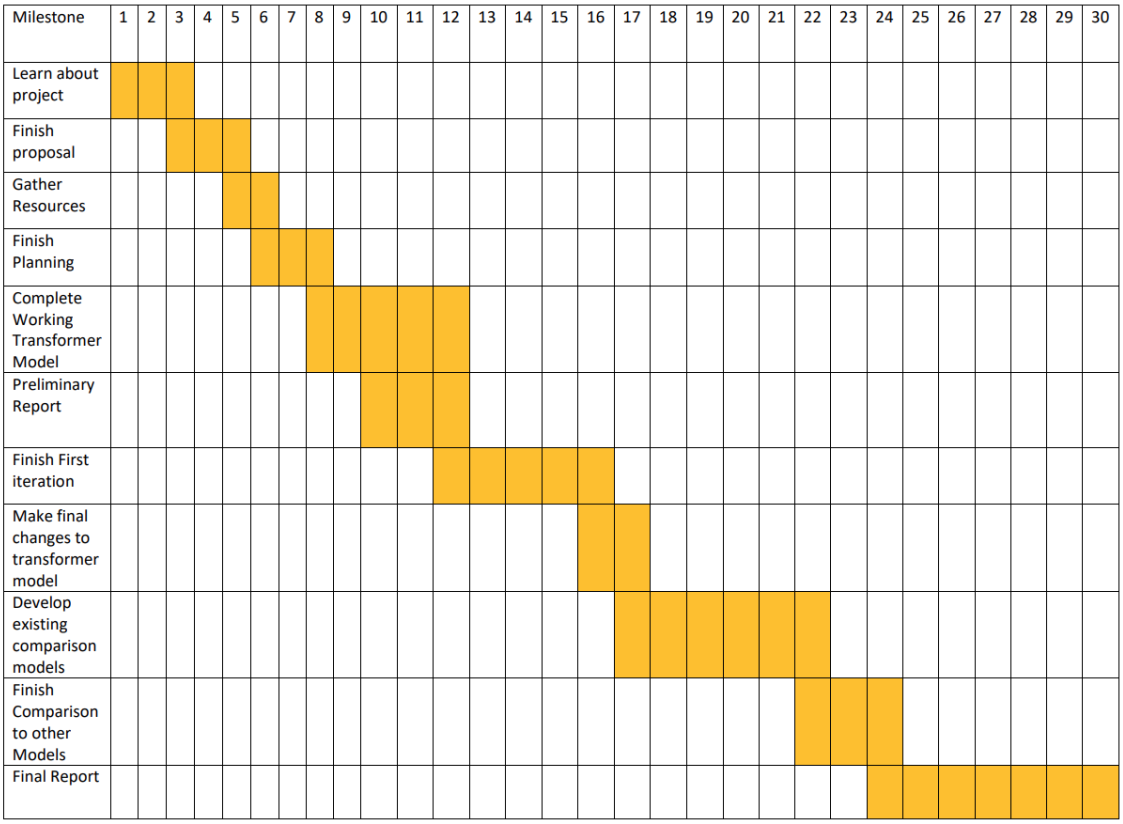
The root mean absolute error is a measure of the average difference between the observed and actual results. This is a measure of accuracy in a sequence of prediction.

4.4 Mean Squared Error (MSE)

The mean squared error is a measure of the abverage of the squares of the difference between the observed and predicted results.

5 Timeline

This timeline is a GAANT chart showing the main milestones of the project and how long each milestone is expected to take. The numbers at the top are in weeks. University breaks are included in the GAANT chart.



6 Resourcing and Ethics

All software I am using will be free and I will to use the software only for the intended use. All data sets used will be open source and free to access to the public. I will follow the correct practices and cite the data I am using when necessary.

6.1 Safety

In this project, I will only be using a computer to develop the model. Basic common sense around electronics will apply.

References

[1] Danilo Ardagna et al. "Quality-of-service in cloud computing: modeling techniques and their applications". In: *Journal of Internet Services and Applications* 5.1 (2014), pp. 1–17.

[2] Rodrigo N Calheiros et al. "Workload prediction using ARIMA model and its impact on cloud applications' QoS". In: *IEEE transactions on cloud computing* 3.4 (2014), pp. 449–458.

[3] Jiechao Gao, Haoyu Wang, and Haiying Shen. "Machine learning based workload prediction in cloud computing". In: *2020 29th international conference on computer communications and networks (ICCCN)*. IEEE. 2020, pp. 1–9.

[4] Oleksii Kuchaiev and Boris Ginsburg. "Factorization tricks for LSTM networks". In: *arXiv preprint arXiv:1703.10722* (2017).

[5] Shiyang Li et al. "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting". In: *Advances in neural information processing systems* 32 (2019).

[6] Qihang Ma. "Comparison of ARIMA, ANN and LSTM for stock price prediction". In: *E3S Web of Conferences*. Vol. 218. EDP Sciences. 2020, p. 01026.

[7] Aidan Meyler, Geoff Kenny, and Terry Quinn. "Forecasting Irish inflation using ARIMA models". In: (1998).

[8] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[9] Mahendra Pratap Yadav, Nisha Pal, and Dharmendar Kumar Yadav. "Workload Prediction over Cloud Server using Time Series Data". In: *2021 11th International Conference on Cloud Computing, Data Science Engineering*. 2021, pp. 267–272.

[10] Mahendra Pratap Yadav, Nisha Pal, and Dharmendar Kumar Yadav. "Workload prediction over cloud server using time series data". In: *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. IEEE. 2021, pp. 267–272.

[11] Yunhao Zhang and Junchi Yan. "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting". In: *International Conference on Learning Representations*. 2023.

[12] Qian Zhu and Gagan Agrawal. "Resource provisioning with budget constraints for adaptive applications in cloud environments". In: *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*. 2010, pp. 304–307.

[13] ZIPPA. *RIGHTSCALE 2019 STATE OF THE CLOUD REPORT FROM FLEXERA*. 2019. URL: `https://resources.flexera.com/web/media/documents/rightscale-2019-state-of-the-cloud-report-from-flexera.pdf` (visited on 03/30/2023).