Performance Prediction of Large Data Transfers

Jason Cheung
Stony Brook University
jason.m.cheung@stonybrook.edu

***Abstract*—Scientific facilities around the world transfer terabytes of data to Berkeley Lab's National Energy Research Scientific Computing Center (NERSC) for processing. These large data transfers can cause congestion on the computer network. To better manage these large transfers, we plan to predict their expected transfer time using machine learning techniques. Through a careful study of traffic logs (Tstat), we find an effective way of utilizing information from recently completed transfers to improve the prediction accuracy by up to 30%.**

I. Introduction

NERSC uses specialized Data Transfer Nodes (DTNs) for high speed transfers. Flows traveling to and from the DTNs are measured using Transfer Control Protocol Statistics (Tstat) [1]. Tstat is widely used for network performance prediction and anomaly detection [2].

Our Tstat logs contain over 300 million flows from 2019 and 2020. Using these records, we train machine learning models to predict the duration of large data transfers. Expected transfer times are useful for resource management and user applications. A user can use the information to plan ahead, download data from the fastest server, or dynamically reserve a high speed link for consistent performance [3].

This study extends previous work in [4]. Nakashima used logs from January and February 2019 to train time series models and predict network throughput. To obtain regular time intervals, he averaged flows over variable length windows. This approach equally weighs small control flows and large data flows. In this study, we pay special attention to large data flows because they can occupy a large share of resources and cause congestion.

II. Methods

We define a large data transfer as either a single file transfer or a batch transfer of smaller size whose total size is within the largest 5% of all flows (larger than 5 GB). The duration of these transfers range from a few minutes to a day.

The first challenge is that a single file transfer can use parallel streams, and a batch transfer can use concurrent streams for each file. To map flows to data transfers, flows with the same source and destination IP that overlap in time by 90% are aggregated.

The second challenge is that many Tstat features are recorded after the transfer is complete. When predicting the duration of a new transfer, only features known before the transfer begins can be used. These base features include the start time, size, duration, and pinged round-trip time (RTT). A limitation of these features is that they do not adequately reflect current network conditions.

Meanwhile, network conditions are captured by the initially unknown features. These features include the standard deviation of the RTT, throughput, and congestion window size. To use these extended features for prediction, their values can be retrieved from the most recently completed transfer to the same site. By also recording the lag time since this transfer, a model can learn whether the historical information is relevant or obsolete.

III. Results

Tstat distinguishes between outgoing versus incoming connections and client-to-server (C2S) versus server-to-client (S2C) transfers, for a total of four transfer directions. This summary presents results for outgoing S2C transfers.

We first train random forest and support vector machine (SVM) with radial basis function kernel using only the base feature set. Then, we extend the feature set using the historical information and retrain the models for comparison.

Figure 1 shows the mean absolute error (MAE) of the models trained on the two feature sets. The extended features improve model performance by up to 30%. Support vector machine is outperformed by random forest. Random forest trained on the extended feature set has an MAE of 300 seconds, which is only 3% of the longest transfer times. Measured versus predicted plots are shown in Figure 2.

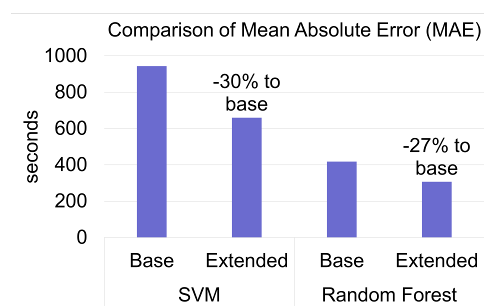
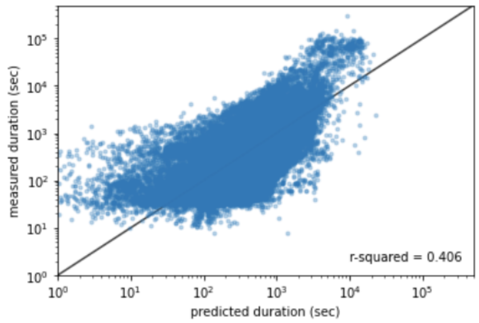


Fig. 1: Comparison of mean absolute error

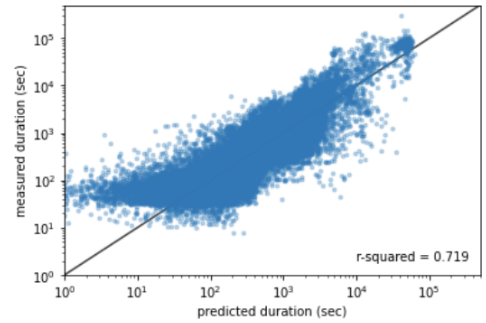
IV. Conclusion

We present three recommendations for preprocessing Tstat data:

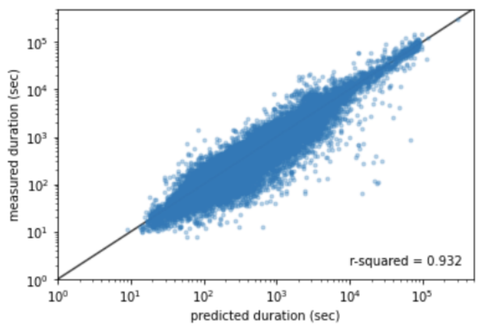
1) Separate models should be developed for each transfer direction



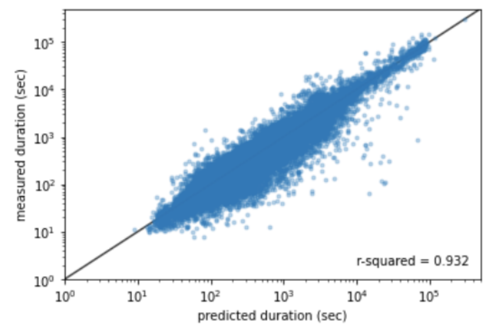
(a) SVM trained on base features



(b) SVM trained on extended features



(c) Random forest trained on base features



(d) Random forest trained on base features

Fig. 2: Predicted versus measured duration

2) Parallel and concurrent streams can be aggregated for large transfers
3) Unknown features can be used for prediction by referencing recent transfers

Referencing recently completed transfers improves model performance. The extended features add awareness to current network conditions and anomalies. In the future, we plan to train deep learning using the two feature sets and classify low performance flows.

Acknowledgment

This work was supported in part by the Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231, and also used resources of the National Energy Research Scientific Computing Center (NERSC). I would like to thank my mentors, Jinoh Kim, Alex Sim and John Wu, at Scientific Data Management (SDM) Group, Lawrence Berkeley National Laboratory. This work was also supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internship (SULI) program.

References

[1] M. Mellia, R. L. Cigno, and F. Neri, "Measuring ip and tcp behavior on edge nodes with tstat," *Computer Networks*, vol. 47, no. 1, pp. 1–21, 2005.

[2] M. Kiran, C. Wang, G. Papadimitriou, A. Mandal, and E. Deelman, "Detecting anomalous packets in network transfers: investigations using pca, autoencoder and isolation forest in tcp," *Machine Learning*, vol. 109, no. 5, 2020.

[3] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "Rsvp: A new resource reservation protocol," *IEEE network*, vol. 7, no. 5, pp. 8–18, 1993.

[4] M. Nakashima, A. Sim, and J. Kim, "Evaluation of deep learning models for network performance prediction for scientific facilities," in *Proceedings of the 3rd International Workshop on Systems and Network Telemetry and Analytics*, 2020, pp. 53–56.