

Predicting Internet Path Dynamics and Performance with Machine Learning

Zhenghui Wang
Shanghai Jiao Tong University
Shanghai, China
felixwzh@outlook.com

Yuheng Zhi
Shanghai Jiao Tong University
Shanghai, China
?@?.com

Hao Wang
Shanghai Jiao Tong University
Shanghai, China
?@?.com

Shukai Liu
Shanghai Jiao Tong University
Shanghai, China
?@?.com

ABSTRACT

We study the problem of predicting internet path dynamics and performance. We use `traceroute` measurement and machine learning models.

KEYWORDS

TODO

ACM Reference Format:

Zhenghui Wang, Hao Wang, Yuheng Zhi, and Shukai Liu. 2017. Predicting Internet Path Dynamics and Performance with Machine Learning. In *Proceedings of SJTU Computer Network Workshop (CNW)*. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

[[Zhenghui says “I think we should focus on the process not the final result, which is also important.”]]

1 INTRODUCTION

1. introduce the original paper: their task, method, and dataset
2. we find some drawbacks in their data. For instance, the route change times problem discussed in github. And the way they process the data.
3. we try some other models to get better performance like xgboost and LSTM.

2 RELATED WORKS

1. we introduce the original paper in details
2. we can introduce some other machine learning methods applied in computer network scenarios. [[Zhenghui says “I’ll take this part”]]
3. very briefly introduce xgboost and lstm

3 DATA ANALYSIS

[[Zhenghui says “I think this is an important part, who will take this part?”]] [[Yuheng says “Plots of temporally differentiated prediction targets are given here to show their behavior patterns in temporal domain, thus supporting the use of LSTMs.”]] With careful review of the data used in the original paper, here we (a) show the authors had used the data incorrectly in the context of making **predictions** and (b) reveal the temporal properties that the original method failed to employ but perfectly fits LSTMs (long short term memory neural networks), which we later use to improve the performance.

3.1 Justification of Data Processing

The original paper [?] made a major mistake on data employment: they implicitly used data from the future to predict events in the future, which is not possible. More specifically, all of their tests on the performance of *NetPerfTrace* depend on statistics calculated out of data of all time, including the data they were to predict. Features like {5, 15, 25...} percentage point of route life, for example, takes the route life of all routes during the whole week into account, including route r_i . However, these features still take part in predicting the residual life of r_i . This is just like guessing something when you knew it.

In fact, a principle of making predictions is to use the data **already observed** to guess the unobserved. Thus, in all the experiments of this paper, we made it a hard constraint that all the features used to predict any event at, say, time t , only come from the raw data collected **before** t . First, we re-validated the performance of classic machine learning methods on the 3 tasks according to our justified data processing manner, in the following 3 ways: (a) use the global features (max value, min value, distribution, etc.) of the former $k\%$ data to train the models and predict the targets of the latter $(1-k)\%$ data, (b) use the global features to train the models and update them during the predicting procedure and (c) use data in a certain recent time-slot to calculate the global features. Then we train LSTMs on these data in two different ways: (a) input only a scalar, the prediction target, to the network and (b) input the prediction target and the global features as a vector.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
CNW, December 2017, Shanghai, China
© 2017 Copyright held by the owner/author(s).
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

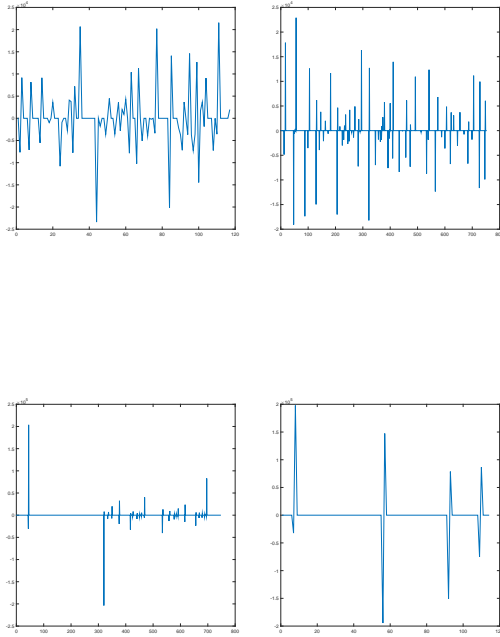


Figure 1: Temporal patterns of route life among 4 different paths.

3.2 Temporal Properties of the Data

We believe temporally adjacent routes and **traceroute** records of a path share some common network nodes, thus their behavior contain implicit dependencies in temporal domain. While the data are organized in the order of time, the original method ignored this property but regard them as independent instances, i.e., the prediction for the next time step only depends on the features of the present, but does not use information given by more previous time steps. This manner fails the original method to capture even the most naive temporal dependencies. Here we do not provide strict proofs of the temporal dependencies that lie in these routes but simply throw some light on the possibility of their existence.

Applying simple differential analysis to the raw data, we found the prediction targets indeed show local temporal patterns which we can hopefully capture with LSTMs. Fig. 1 to Fig. 3 exhibit differentiated results of the three prediction targets in different paths. We can see that these paths behave in somehow consistent temporal patterns along their own ways but differently from each other.

No more solid evidence could be provided to prove the periodicity of these patterns because **pattern** itself is hard to define. However, these results are already enough to make

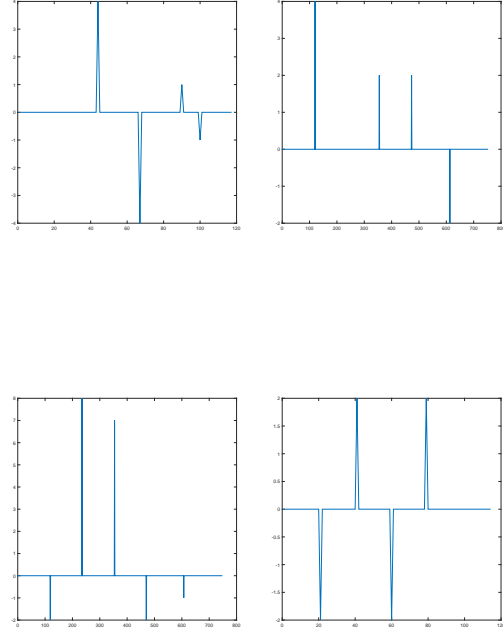


Figure 2: Temporal patterns of the number of route changes among 4 different paths.

us hopeful about the performance of LSTMs, which are expert in capturing temporal dependencies, no-matter long-term ones or short-term ones.

1. We can plot some figures of the statistics of data, like the distribution of the route duration and avgRTT.
2. We can further discuss the relation between routes in one path or in different paths.
3. Then we could discuss why the authors of the paper process the data in a wrong way
4. We show our solution for data process. three ways for random forest models.

4 EXPERIMENT

4.1 Classic Models

We show the experiment results of the 3 different data we obtained, namely **K&fix**, **K&update**, **timeslot&update**. We need to find some difference between our 3 data processing methods and the authors', i.e., **origin**.

The experiments we need to conduct are as follows:

- 1. **K&fix**+RF
- 2. **K&update**+RF
- 3. **timeslot&update**+RF
- 4. **origin**+RF
- 5. **K&fix**+xgBoost

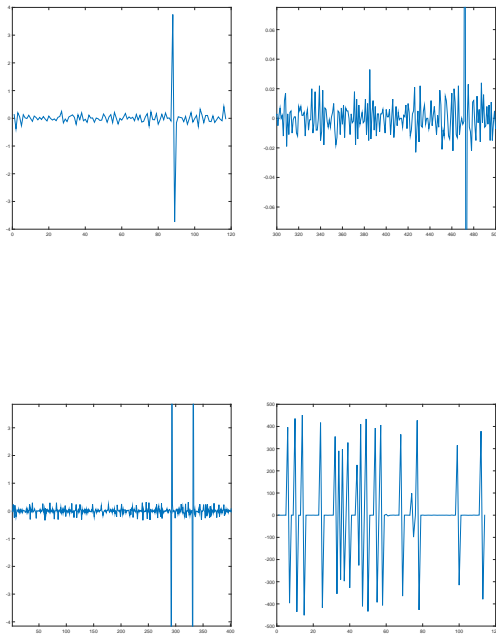


Figure 3: Temporal patterns of round trip time among 4 different paths.

- 6. **K&update**+xgBoost
- 7. **timeslot&update**+xgBoost
- 8. **origin**+xgBoost

[[Zhenghui says “Note”]] We first predict the route duration instead of resLife. Further study on the difference between these two predicting objective could be conduct if we have time. Currently, we predict the route duration for task 1, because we can only predict this when we use LSTM.

4.2 Deep Models

There are two kinds of input for the LSTM at each timestep, (i) one simple scalar (ii) a vector. The experiments we need to conduct are as follows:

- scalar input + LSTM
- vector input + LSTM

5 CONCLUSIONS

TODO

ACKNOWLEDGMENTS

TODO