An Efficient Framework for Online Traffic Time Series Forecasting

Miao Lu*, Lin-Yu Tai, Jian Yang
Yahoo/Oath Research

Abstract

This paper describes our ongoing research on the time series forecasting framework: an efficient machine learning system for time series forecasting of online traffic. The framework is an ensemble-model based time series/machine learning forecasting, with MySQL database, backend/frontend dashboard, and Hadoop streaming. The novel framework involves data aggregation and grouping, holiday handling, change point/anomaly detection, and model ensemble, for better time series forecasting. Numerical experiments show the advantageous performance of proposed method over the existing baselines.

1 Introduction

Time series forecasting is of essential importance in the business operations. The main purpose of time series forecasting is to carefully collect and rigorously study the historical observations to develop an appropriate model for future values. For example, internet companies might be interested in the daily active users (DAU), say, what is DAU after certain period of time.

There exist many models about time series, among which, the most popular and frequently used time series model is the Autoregressive Integrated Moving Average (ARIMA) [1, 2, 3, 4]. Taking seasonality into consideration, [1] proposed the Seasonal ARIMA. The Holt–Winters method [5] is an alternative by using exponential smoothing. State space model [6, 7, 8] also attracts much attention, which is a linear function of an underlying Markov process plus additive noise. Exponential Smoothing State Space Model (ETS) [9] decomposes times series into error, trend, seasonality that change over time. Recently, deep learning is applied for time-series trend learning using LSTM [10], bidirectional dynamic Boltzmann machine [11] is used for time-series long-term dependency learning, and coherent probabilistic forecast [12] is proposed for a hierarchy or an aggregation-level comprising a set of time series. In Internet industry, Google develops the Bayesian structure time series (BSTS) model [8, 7] to capture the trend, seasonality, and similar components of the target series, with its robust time series forecasting at scale [14]. Facebook proposes the Prophet approach [15] based on a decomposable model with interpretable parameters that can be intuitively adjusted by analyst. However, as in the DAU example, some special events like Christmas Holiday or President Election, newly launched apps or features, may cause short period or long-term change of DAU, leading to weird forecasting of these traditional models. The aforementioned cases are well known as

(1) Anomaly points. The events or observations that don't conform to an expected pattern or other items in the dataset, leading to a sudden spike or decrease.

(2) Change points. A market intervention, such as a new product launch or the onset of an ads campaign, may lead to the level change of the original series.

(3) Holiday points. Different countries have various culture and holidays, leading to different periodical holiday impact.

Time series without change point, anomaly detection and holiday handling may lead to bizarre forecasting since these models might learn the abrupt changes in the past. There are literatures on detecting anomaly or change points separately, examples can be found in [16, 17, 18, 19, 20]. However, the aforementioned change point detection models could not support detection in the presence of seasonality, while the presence of trend/change point is not handled by the anomaly detection models. Most importantly, there doesn't exist a framework that incorporate data aggregation/adjustment, change point, anomaly detection, holiday handling as well as time series forecasting.

Strongly motivated by overcoming the limitations of the most (if not all) current models that the anomaly and change points are not properly considered, we propose an efficient framework for time series forecasting, including the change point and anomaly detection, holiday smooth and restore, ensemble modeling, etc. The developed framework is applied to the online traffic forecasting for daily traffic across different properties, regions and devices.

2 Framework

It is known to all that growing audience traffic is the key driver of long term revenue growth. Thus, accu-

*Email: mlu@oath.com.

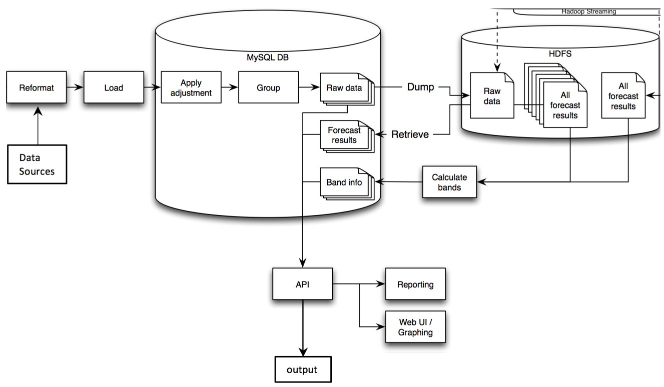

Figure 1: Online Traffic Forecasting Framework

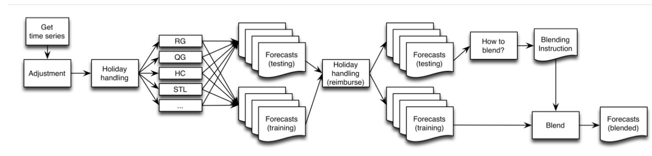

Figure 2: Model Pipeline via Hadoop Streaming for Online Traffic Forecasting

rate and frequent projection framework of traffic forecasting help guide the business decisions. The heart of online traffic forecasting framework is an ensemble model based time series/machine learning forecasting, with MySQL database, shell php backend/frontend, and Hadoop streaming. The Figure 2 sketches the framework of online traffic forecasting. After reformatting and loading data from different sources, a MySQL database is employed to perform multiple adjustment and grouping. The data is dumped to Hadoop Distributed File System (HDFS) with Hadoop streaming running models. The forecasting results are retrieved back to production server, displayed as dashboard visualization and output reporting.

Data Structure
The data structure involves the interaction with multiple servers. The data is downloaded from HDFS server to the production server, where multiple steps are processed such as importing and aggregating into MySQL, and generating time series data from MySQL as model input. The grid server is running Hadoop streaming jobs, interacting with production server for model inputs and outputs. A website user interface (UI) on production server is for graphs visualization of time series and ad-hoc analyses.

Data Preparation
Data preparation has several parts. First, different sources of data have to be reformatted before loading. Second, level and discrepancy adjustments are used in the database. Third, grouping is conducted based on different products, devices and regions mappings, and filling time series gaps. Fourth, a forecast plan with all parameters should be accompanied with the raw time series before dumping to HDFS and running Hadoop streaming on the top right of Figure 2.

After preparing the raw time series, an efficient model pipeline is running via Hadoop steaming. In Figure 2, the input time series is performed procedures such as adjustment (change point, anomaly detection), holiday handling, ensemble modeling, blending, etc, in order to obtain the forecasting results.

Change Point and Anomaly Detection
As stated in the introduction, anomaly points depicts sudden change of series. They might be caused by some unknown interventions or simply model noises. Similarly, change point indicates long-term level change of original series. Without change point and anomaly detection, as well as their adjustment, the future forecasting may occur weird results.

Holiday Handling
Holiday handling is separately into two parts. Firstly, after inputting the holiday lists, we could identify the holidays and smooth those points by interpolations before feeding into models. Secondly, a reimburse is performed to restore the future holidays to their volumes based on the previous smoothing ratios.

Model Ensemble
Time series data is split into training and testing datasets, and the ensemble model includes statistical time series models (e.g., seasonal trend with loess (STL), Prophet), machine learning models (e.g., Support Vector Machine (SVM), Random Forest (RF), Gradient Boosting (GB)), and deep learning models (e.g., Artificial Neural Network (ANN)). The blending weight depends on the model performance on the testing data, i.e., more weight is imposed on better performed model.

3 Experiment

We ran the online traffic forecasting for different properties, regions and devices. The Table 1 is one example of traffic forecasting performance for homepage on website in united states. The performance metrics mean absolute percentage error (MAPE) and root mean square error (RMSE) are defined as

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|x_i - \hat{x}_i|}{x_i}, RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}.$$

Also, the listed baselines are time series models: STL, BSTS from Google, Prophet from Facebook, machine learning models: SVM, RF, and GB, as well as ANN model. In Table 1, the proposed framework achieves the lowest daily/weekly MAPE and RMSE, compared with all other methods.

Table 1: Testing Errors for Different Methods.

Methods	Daily MAPE	Weekly MAPE	RMSE
STL	0.24	0.24	17.94
BSTS	0.25	0.06	15.71
Prophet	0.08	0.06	6.62
SVM	0.18	0.16	12.48
GB	0.16	0.13	11.00
RF	0.22	0.19	13.94
ANN	0.23	0.23	17.29
Proposed	**0.05**	**0.03**	**5.32**

4 Conclusion

In this paper, we propose an efficient framework to forecast online traffic across different regions, properties and devices. This new framework could achieve Hadoop streaming, series forecasting parallelization, and dashboard visualization. The developed data/model pipelines incorporate multiple steps such as data aggregation and grouping, holiday handling, change point/anomaly detection, and model ensemble. Besides the machine automation, human could also feed in intelligence of holidays or interventions via forecast plans. This framework is also easily to adopt more forecasting models and data processing modules.

References

[1] Box, George EP and Jenkins, Gwilym M and Reinsel, Gregory C and Ljung, Greta M, *Time series analysis: forecasting and control*, John Wiley & Sons, 2015.

[2] Zhang, G Peter, *Time series forecasting using a hybrid ARIMA and neural network model*, Neurocomputing, 50: 159-175, 2003.

[3] Cochrane, John H, *Time series for macroeconomics and finance*, 2015.

[4] Hipel, Keith W and McLeod, A Ian, *Time series modeling of water resources and environmental systems*, volume 45. Elsevier, 1994.

[5] Winters, Peter R, *Forecasting sales by exponentially weighted moving averages*, Management science, 6(3): 324-342, 1960.

[6] Durbin, James and Koopman, Siem Jan, *Time series analysis by state space methods*, volume 38. OUP Oxford, 2012.

[7] Scott, Steven L and Varian, Hal R, *Predicting the present with bayesian structural time series*, International Journal of Mathematical Modeling and Numerical Optimization, 5(1-2):4-23, 2014.

[8] Brodersen, Kay H and Gallusser, Fabian and Koehler, Jim and Remy, Nicolas and Scott, Steven L and others, *Inferring causal impact using Bayesian structural time-series models*, The Annals of Applied Statistics, 9(1):247-274, 2015.

[9] Hyndman, Rob and Koehler, Anne B and Ord, J Keith and Snyder, Ralph D, *Forecasting with exponential smoothing: the state space approach*, Springer Science & Business Media, 2008.

[10] Tao, Lin and Tian, Guo and Karl, Aberer, *Hybrid Neural Networks for Learning the Trend in Time Series*, In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, pp. 2273-2279, 2017.

[11] Takayuki Osogami and Hiroshi Kajino and Taro Sekiyama, *Bidirectional Learning for Time-series Models with Hidden Units*, In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017.

[12] Souhaib Ben Taieb and James W. Taylor and Rob J. Hyndman, *Coherent Probabilistic Forecasts for Hierarchical Time Series*, In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017.

[13] Tassone, Eric and Rahani, Farzan, *Our quest for robust time series forecasting at scale*, http://www.unofficialgoogledatascience.com/2017/04/our-quest-for-robust-time-series.html, 2017.

[14] Tassone, Eric and Rahani, Farzan, *Our quest for robust time series forecasting at scale*, http://www.unofficialgoogledatascience.com/2017/04/our-quest-for-robust-time-series.html, 2017.

[15] Taylor, S. J. and Letham, *Prophet: forecasting at scale*, https://research.fb.com/prophet-forecasting-at-scale/, 2017.

[16] Twitter, *AnomalyDetection: Anomaly Detection with R*, https://github.com/twitter/AnomalyDetection, 2017a.

[17] Netflix, *RAD: Time Series Anomaly Detection*, https://github.com/Netflix/Surus/, 2017.

[18] Barry, Daniel and Hartigan, John A, *A Bayesian analysis for change point problems*, Journal of the American Statistical Association, 88(421):309-319, 1993.

[19] Killick, Rebecca and Eckley, Idris, *changepoint: An R package for changepoint analysis*, Journal of Statistical Software, 58(3):1-19, 2014.

[20] Twitter, *Breakout Detection via Robust E-Statistics*, https://github.com/twitter/BreakoutDetection, 2017b.