Forecasting Street Speed using Alternative Data

Lingbo Ji
New York University

Levente Szabo
New York University

Yaowei Zong
New York University

***Abstract*—Forecasting of street speed is a pivotal component of many traffic analysis and route prediction algorithms. Street speed over a geographic region is highly non-linear being effected by both seasonal and environmental factors. To test our hypothesis of using alternative data sources to supplement forecasting street speed we train a baseline model against one with additional weather data and traffic collision counts. We train and evaluate our model over several New York City streets. Our results indicate significant improvement over the baseline particularly due to the pandemic lockdown in early 2020.**

***Index Terms*—street speed, forecasting, alternative data**

I. Introduction

New York City is the 14^{th} most traffic congested city in the world and the 4^{th} worst city in the United States. New York City also has two (second and third) of the top 10 most congested corridors in the country [1]. Traffic analysis and route prediction systems are used by drivers and city planners alike to decrease travel times. Early results for traffic forecasting include ARIMA modelling [2] which show promising results through capturing trend and seasonality. Such models are relatively lightweight but require rigorous fine tuning additionally adding features to a baseline model remains difficult. Contemporary traffic models include approaches at using deep learning [3] to model the spatio-temporal relationships underlying traffic patterns. The computational costs of implementing such models however limits their practicality in some situations. In order to find a middle ground between simple time series forecasting and deep learning we propose a multivariate time series forecasting approach where external environmental data is utilized to improve traffic forecasting accuracy. To test the effectiveness of our approach we predict street speeds with additional weather data and motor vehicle collision counts by utilizing Fbprophet [4], a lightweight time series forecasting toolkit. Our models are trained on New York City street speed data from 2019 and are used to forecast speeds for 2020.

II. Motivation

Congestion can cost time and money. Overall, the average hours lost in congestion for American drivers are 99 hours a year in 2019 (140 hours for NYC), with a $1,377 cost ($2,072 for NYC) [1]. Additionally the 5 most congested cities in the U.S. are also the cities that have higher scores in utilizing public transit to replace driving. Understanding traffic speed patterns can help city planers and daily commuters to save unnecessary speed due to congestion. Utilizing alternative data sources for the traffic forecasting problem could assist both more accurate predictions on local levels (street scale) as well as on macro levels (neighborhood traffic patterns). Furthermore, with the advent of autonomous vehicles we see fine tuned traffic forecasting that is both flexible and computationally feasible as a critical research direction.

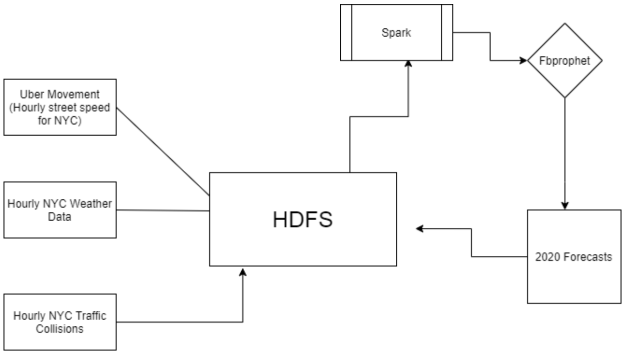

Spark
Fbprophet
Uber Movement (Hourly street speed for NYC)
Hourly NYC Weather Data
HDFS
2020 Forecasts
Hourly NYC Traffic Collisions

Fig. 1. System Architecture

III. Related Work

Some previous studies on traffic speed forecasting mainly focused on a single road segment, where statistical techniques were widely used, since the traffic conditions and transportation data set they used were relatively small. For example, Yang used Kalman Filtering [5] and Williams and Hoel used ARIMA [2] in their studies, etc. In the recent years, the widely deployed traffic sensors quickly increase data availability, size and coverage. Therefore, to deal with complex traffic conditions and capture non-linear relationships, many machine learning methods have been employed to forecast traffic speed. In [6] a novel hybrid model, S-GCN-GRU-NN is proposed, in which a spatiotemporal graph convolutional network model was used for acquiring the complex spatiotemporal dependencies and a gated recurrent unites neural network model was used for short-term traffic speed forecasting. However, due to external factors such as hardware equipment, the data set' s time span is very short and the volume of street segments is very limited. So that the prediction results have great limitations and the predictive accuracy cannot be further improved.

Another prediction model using generic algorithm-support vector machine (GA-SVM) by H. Niu, et al [7], shows that parallel genetic optimized SVM on cloud computing has higher prediction accuracy, shorter running time, and higher speedup.

IV. Design and Implementation

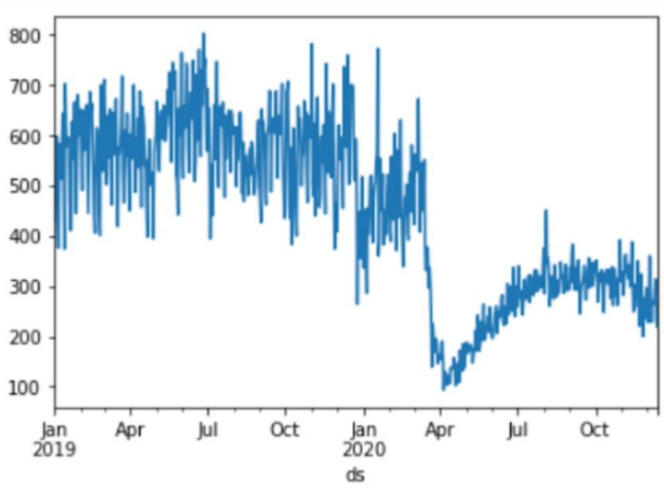


Fig. 2. Daily Traffic Collisions: NYC

A. Design Details

Our system architecture is displayed in Fiugre 1. Initially the raw data is taken from three different sources. The hourly street speed data, the weather data, and motor vehicle crashes. After the data sources are placed into HDFS we apply data cleaning and pre-processing using MapReduce to construct currated data sets that are ready for analysis.

Then using Spark we construct DataFrames over the data sources and apply additional steps to fit the structure for forecasting.

Finally using our time series forecasting tool, Fbprophet in conjunction with Pyspark we construct forecasts over our testing period for a fixed number of streets. These can then be stored in HDFS or otherwise the average RMSE is calculated against the true street speed values.

V. Datasets

There are three datasets that we used in this project. The Uber street speed data served as our main focus and we used local weather reports and vehicle collision reports as our external data source to perform prediction on street speed. In order to construct forecasting on our data we eventually aggregate everything to average daily values and place data into the data structure required for Fbprophet forecasting models. The format requires two specific columns, *ds* and *y*, in our case the day and speed respectively. Additionally we append the most useful features to our data structure.

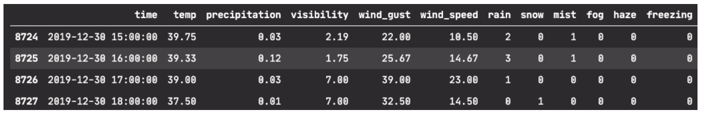
A. NYC Uber Movement Street Speed Data

The NYC street speed data is obtained from Uber Movement [8], where the speed values are derived from average speed readings from on-trip Uber data across the street segments. For our interests in this project, we downloaded the monthly data file which provides the average speed on a given road segments for each hour of each day for New York City, ranging from January 2019 to March 2020. The data set is 72GB in total, 58GB for fitting model and 14GB to verify the forecast accuracy of the model. To pre-process the original data, the following three steps were to be taken. First, we selected the speed and time columns, transformed them into a time series format by using MapReduce. The key of the result is composed of a street's start node ID and end node ID while value is composed of 8760 speed data points which is in a time series format. Second, for each rode, the hourly average speed of a day was calculated and it was used to fill in the null value according to the corresponding hour. Last, if any roads still have null values, we use the total average speed of this road segment to replace it.

B. NYC Local Climatological Data

Local Climatological Data is pulled from National Centers for Environmental Information (NCEI, formarly the National Climatic Data Center, NCDC) contains historical weather reports for NYC collected at central park weather station [9]. There are 15099 rows for the tine range from January 2019 to March 2020. Each row is one report for a specific time, usually on an hourly basis, consists 124 comma separated data fields. For this project, we extract only the fields we need, including DATE(timestamp), HourlyDryBulbTemperature, HourlyPrecipitation, HourlyPresentWeatherType, HourlyVisibility, HourlyWindGustSpeed and HourlyWindSpeed. A detailed schema can be found in LCD Dataset Documentation [10].

We utilized MapReduce to extract those fields of interests. During data cleaning, we noticed that there might be multiple entries per hour, thus we calculated the average value for numeric field such as temperate and visibility, and use indicators range from 0-3 for different levels of weather conditions such as rain, snow, fog and mist. An example of weather data output is shown in Figure 3.



	time	temp	precipitation	visibility	wind_gust	wind_speed	rain	snow	mist	fog	haze	freezing
8724	2019-12-30 15:00:00	39.75	0.03	2.19	22.00	10.50	2	0	1	0	0	0
8725	2019-12-30 16:00:00	39.33	0.12	1.75	25.67	14.67	3	0	1	0	0	0
8726	2019-12-30 17:00:00	39.00	0.03	7.00	39.00	23.00	1	0	0	0	0	0
8727	2019-12-30 18:00:00	37.50	0.01	7.00	32.50	14.50	0	1	0	0	0	0

Fig. 3. Weather Output Schema

C. NYC Motor Vehicle Collisions Data

To encapsulate additional traffic congestion indicators for NYC the Motor Vehicle Collision -Crashes data is provided by the NYPD through NYC Open Data [11]. Every data element represents a crash event and contains publicly available information included in the police report regarding a given crash.

To provide usable results for our time series forecasting model we transform data by selecting the number of crashes for each day over our desired time frame (2019-2020). The number of traffic collisions over this period can be seen in Figure 2. We believe that traffic collisions across the city can serve as a proxy for local congestion on specific streets. Additionally we note that the large drop in collisions during March 2020 is due to the COVID-19 pandemic lockdown. We posit that including this data will improve 2020 forecasting results by accounting for the pandemic slowdown.

VI. Results

A. Models

A Fbprophet time series forecasting model is decomposable as noted in (1). The trend $g(t)$ refers to the overall behavior of the time series and can be linear or logisitic. The seasonality $s(t)$ refers to the effect of time related components such as time of day, week,year. Holiday effects refer to effects of specific days on our prediction and is depicted as $h(t)$, this concept can be extended to include additional external regressors such as our alternative environmental data. Finally ϵ denotes the additional components not accounted for by the model components, this is assumed to be a normal distribution.

$$y(t) = g(t) + s(t) + h(t) + \epsilon \tag{1}$$

Hyperparameters Fbprophet is an out of the box time series forecasting model in the sense that it does not require users to calculate features or apply rigorous preprocessing. To conduct forecasting the time series just needs to be put into the format usable by the model. There are however several hyperparameters regarding the components noted in (1). Due to the nature of vehicle traffic trend can only be *linear* while both seasonality and our added regressors can both be either *additive* or *multiplicative*. When seasonality or regressors are additive it means that the model learns the fluctuations from the trend line through addding or subtracting. In contrast multiplicative regressors measure fluctuations from the trend line through positive or negative dilation.

Model 1 Our baseline model utilizes only trend and seasonality with no added weather features or traffic collision counts. We construct a Fbprophet model trained on daily speed data from 1/1/2019 to 12/31/2019 and forecast street speeds over 1/1/2020 to 3/31/2020. The error for a single street is the Root Mean Squared Error (RMSE) and the error over dataset is the average RMSE.

Model 2 Our second model utilizes both trend and seasonality but also incorporates the weather features *rain*, *snow*, *freezing*, *visiblity*, *crashes*. These additional features are added as regressors and street speed is forecast along an identical time horizon as Model 1. We utilize the average RMSE to evaluate the efficacy of using the additional features.

B. Experiment

Our validation set consists of a randomly selected interconnected road network of 15 road segments. We utilize this validation set to tune our hyperparameters. The topological structure of the 15 road segments and their neighbors is shown in Figure 4.

First, to gain insight into the problem we used the baseline model to predict the speed of these 15 roads, and compared the predicted value with their true value for fitting. The fitting curve of train set and part of the test set showed that we did get a decent model, However the overall error score of the test set performed poorly overall. All of the 15 road segments had an obvious fork in the tail, as the red line and green line show on the left side of the Figure 5. Then, we chose March 1^{st} as the threshold to split the testing set into two parts and calculated the error score separately. By comparison, we found that the forecasting results when excluding March were significantly better than when it was included.

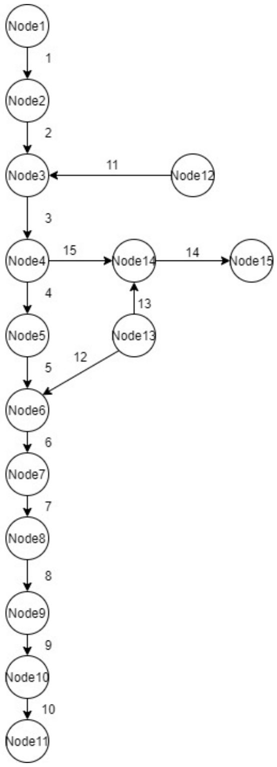


Fig. 4. Roads Sketch

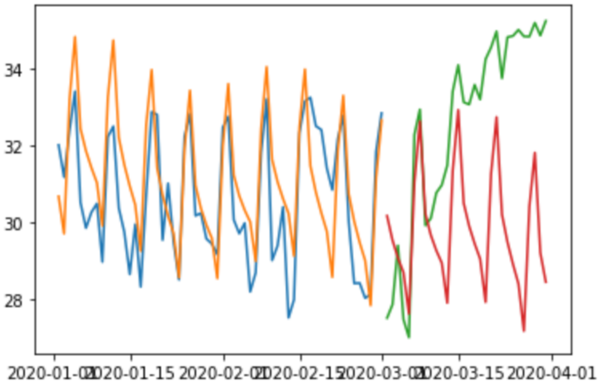
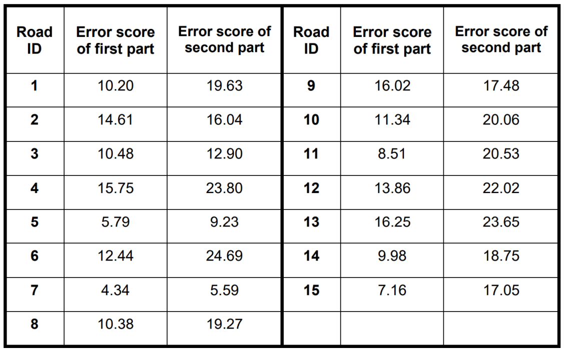


Fig. 5. Error Score result Of Road 1

As shown in Figure 1., taking road 1 as an example, the orange line and blue line represent the actual and predicted values before March while the green line and red line represent the actual values and predicted values in March.

Fig.6 is a error score table, it shows the two-part error score result of the 15 road segments. It can be seen that the difference between the results of the two parts is very large in all segments.



Road ID	Error score of first part	Error score of second part	Road ID	Error score of first part	Error score of second part
1	10.20	19.63	9	16.02	17.48
2	14.61	16.04	10	11.34	20.06
3	10.48	12.90	11	8.51	20.53
4	15.75	23.80	12	13.86	22.02
5	5.79	9.23	13	16.25	23.65
6	12.44	24.69	14	9.98	18.75
7	4.34	5.59	15	7.16	17.05
8	10.38	19.27			

Fig. 6. Error Score result of two-part testing set

After discovering the discrepancy over the testing period, we tried to improve our initial model and added the environmental features to it. Then, we used the improved model to predict the speeds of these 15 road segments again. As shown in the line chart on the right side of Figure 7., for these specific street segments the improved model prediction results corrected the previous problem.

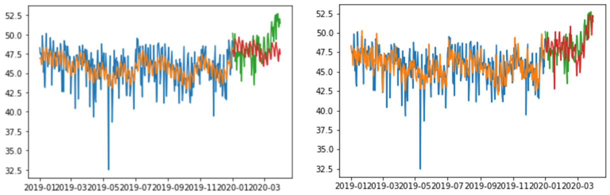


Fig. 7. Error Score result of two models

In order to support the conclusion that the improved model has indeed increased the model's predictive power, we calculated the error scores of the same road segments using the baseline and the improved model for comparison. The error comparison is shown in Figure 8., the average RMSE over the road network decreased from 21.38 to 14.73, a 31% reduction.

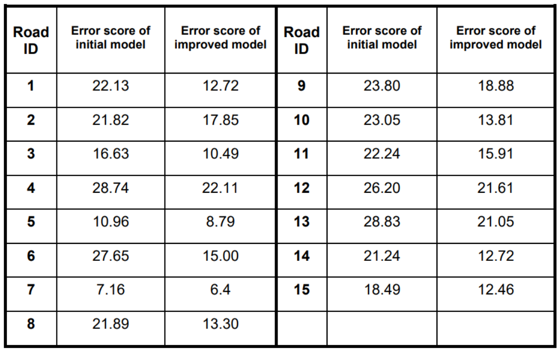
Next we tuned the trend, seasonality and additional regressor hyperparameters on the validation set. The average RMSE for each set of parameters is shown for both models in Figure 8 . The model using linear trend, multiplicative seasonality and multiplicative added regressors achieved the lowest average RMSE. Finally the model was assessed on a testing set of $N = 295$ street time series with the baseline model achieving an average RMSE of 19.55 and the model with added regressors attained a RMSE of 16.55. The strong effect of the pandemic lockdown in the second half of our testing period appears to be accounted for when using the additional external regressors.

Hyperparameters	Model 1	Model 2
Linear Additive Additive	21.38	15.35
Linear Additive Multiplicative	21.38	15.07
Linear Multiplicative Multiplicative	**21.97**	**14.88**
Linear Multiplicative Additive	21.97	14.90

Fig. 8. Hyperparameter Errors

VII. Future Work

In the future, we will consider more environmental factors, such as traffic flow, proxies for COVID-19 outbreak, construction and additional road conditions to improve the forecasting accuracy of the proposed model. Moreover, we will try to use deep learning method to further improve the model by learning the spatio-temporal relationships over road networks. Ideally we would like to leverage our models to build a traffic surge forecasting application capable of providing drivers with geographic congestion predictions in real time.



Road ID	Error score of initial model	Error score of improved model	Road ID	Error score of initial model	Error score of improved model
1	22.13	12.72	9	23.80	18.88
2	21.82	17.85	10	23.05	13.81
3	16.63	10.49	11	22.24	15.91
4	28.74	22.11	12	26.20	21.61
5	10.96	8.79	13	28.83	21.05
6	27.65	15.00	14	21.24	12.72
7	7.16	6.4	15	18.49	12.46
8	21.89	13.30			

Fig. 9. Error Score result of two models

VIII. Conclusion

Our hypothesis was correct in that utilizing external environmental features like weather and traffic collisions counts is an improvement over the baseline time series forecasts. We believe the COVID-19 lockdown is responsible for larger error on the baseline and that our model anticipates that. Fbprophet is a lightweight time series forecasting tool that has allowed us to test our hypothesis regarding alternative data use for the traffic forecasting problem. Despite limiting model complexity, utilizing Fbprophet for forecasting along with Apache Spark for data infrastructure has allowed us to quickly prototype and test our ideas.

Acknowledgment

We would like to thank NYU HPC team for providing access to the Dumbo Computing Cluster, where this project was developed and tested on. We would like to especially thank Dr. Lakshminarayanan Subramanian for his guidance and feedback through out the project. We would also like to express our deepest sympathy for the loss of Dr. Suzanne K. McIntosh, a great teacher and mentor.

References

[1] T. Reed, “Inrix global traffic scorecard(2019),” INRIX Research, Mar. 2020.

[2] B. Williams and L. Hoel, “Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results,” *Journal of Transportation Engineering*, vol. 129, pp. 664–672, 11 2003.

[3] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Wang, “Traffic flow prediction with big data: A deep learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.

[4] S. J. Taylor and B. Letham, “Forecasting at scale,” *PeerJ Preprints*, vol. 5, p. e3190v2, Sep. 2017. [Online]. Available: https://doi.org/10.7287/peerj.preprints.3190v2

[5] L. H. R. B. Yang F, Yin Z, “Online recursive algorithm for short-term traffic prediction.” *Transportation Research Record Journal of the Transportation Research Board*, vol. 1879, pp. 1–8, 2004.

[6] X. L. Manrui Jiang, Wei Chen, “S-gcn-gru-nn: A novel hybrid model by combining a spatiotemporal graph convolutional network and a gated recurrent units neural network for short-term traffic speed forecasting,” vol. 35, pp. 3–25, 2020.

[7] H. Niu, Z. Yang, D. Mei, Q. Yang, H. Zhou, and X. Li, “Traffic flow prediction model for large-scale road network based on cloud computing.” *Mathematical Problems in Engineering*, vol. 2014, p. 926251, 2014.

[8] “Uber historical speeds, hourly time series,” Uber Movement, Dec. 2020. [Online]. Available: https://movement.uber.com/cities/new$_{y}ork/downloads/speed$

[9] “Nyc local climatological data,” National Centers for Environmental Information, Dec. 2020. [Online]. Available: https://www.ncdc.noaa.gov/cdo-web/datasets/LCD/stations/WBAN:94728/detail

[10] “Local climatological data (lcd) dataset documentation,” National Centers for Environmental Information, 2020. [Online]. Available: https://www.ncdc.noaa.gov/cdo-web/datatools/lcd

[11] “Motor vehicle collisions - crashes,” NYC Open Data, Dec. 2020. [Online]. Available: https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95