Traffic Forecasting In Complex Urban Networks: Leveraging Big Data and Machine Learning

Florin Schimbinschi, Xuan Vinh Nguyen, James Bailey, Chris Leckie, Hai Vu, Rao Kotagiri
Department of Computing and Information Systems, The University of Melbourne
Email: florinsch@student.unimelb.edu.au, vinh.nguyen@unimelb.edu.au, baileyj@unimelb.edu.au, caleckie@unimelb.edu.au, hvu@swin.edu.au, kotagiri@unimelb.edu.au

***Abstract*—Accurate network-wide real time traffic forecasting is essential for next generation smart cities. In this context, we study a novel and complex traffic data set and explore the potential to apply big data and machine learning analysis. We evaluate several hypotheses and find that the availability of big data is able to facilitate more accurate predictions. Furthermore, we find that spatial aspects have more influence than temporal ones and that careful choice of thresholding parameters is crucial for high performance classification.**

***Keywords*-big data; traffic forecasting; time series prediction;**

I. Introduction

The availability of detailed data streams on road networks offers great promise for intelligent transport in the context of smart cities. Applications include personalized copilots with real time route suggestions based on user preferences and traffic conditions, economical parking metering, agile car pooling services and self driving cars. At the core of these systems, a proactive and accurate, network-wide, real-time traffic prediction system is paramount.

Current systems are largely reactive since much of the existing work has been performed on simple freeway datasets that do not entirely capture the complex spatiotemporal characteristics of a city's traffic. Conversely, we introduce an intricate network dataset and leverage big data and machine learning for traffic forecasting in complex urban networks.

Our contributions include:

- Investigation of a novel big traffic data set, including an exploratory analysis and benchmarking of state of the art machine learning algorithms.
- Evaluation of the following hypotheses: Is a larger sampling bandwidth beneficial? If so, is it because it captures the large variance between weekends and weekdays? Does augmenting missing data with contextual average trends increase accuracy? Is recent data more useful on its own? Does big data help even if old data is used? What is the effect of including proximity data? Is it more important than sampling bandwidth?

Our evaluations reveal that: i) the spatiotemporal representation is one of the central issues, ii) predicting only on weekdays is easier and separate predictors can be deployed separately for weekends or each day of the week, iii) adjusting sampling bandwidth and proximity data increases performance, iv) class label thresholds should be set dynamically or avoided altogether.

II. Dataset

The VicRoads dataset was collected in the City of Melbourne over six years. A special feature of this dataset is its volume and variety, covering the Central Business District (CBD) and suburban areas, including freeways as depicted in the figure below. A quantitative comparison with existing datasets is given in Table I. To the best of our knowledge, it is the first dataset of its kind studied by the community. Vehicle volume count data is recorded using loop detectors at a frequency of 1 minute for the raw data set, which measures about 700 GB per year. VicRoads recently released* all their fine-grain traffic volume data. In this article we use a subset.

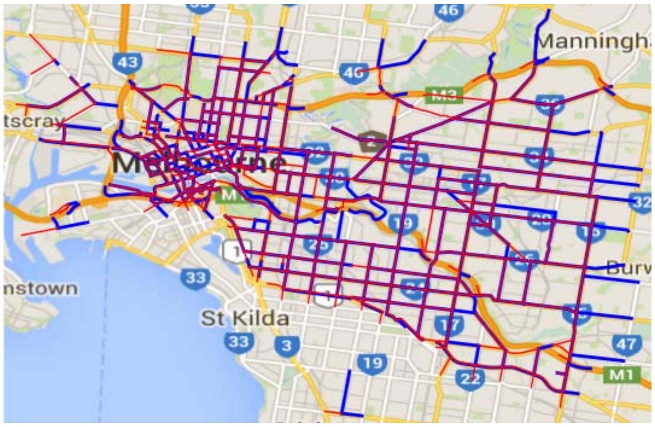

Figure 1: Melbourne roads with available traffic data are highlighted. Each physical road typically has 2 traffic directions, colored red and blue. Latitude and longitude coordinates for each sensor are also included.

The data stream is aggregated into 96 bins for each day, taken at 15 minute intervals for each sensor. Measurements are aggregated for all lanes in the same direction, aggregating into road segment level statistics. These tensors thus contain traffic information for 2033 days × 1084 sensors × 96 time points per day.

III. Exploratory data analysis

A random subset of days was gathered for one sensor and the 96 dimensional traffic volume vectors projected onto a 2D space using a randomly generated orthonormal matrix. Random embeddings onto lower dimensions preserve Euclidean geometry, thus three clusters are visible in Figure 2.

*https://vicroads-public.sharepoint.com/InformationAccess/SitePages/Home.aspx

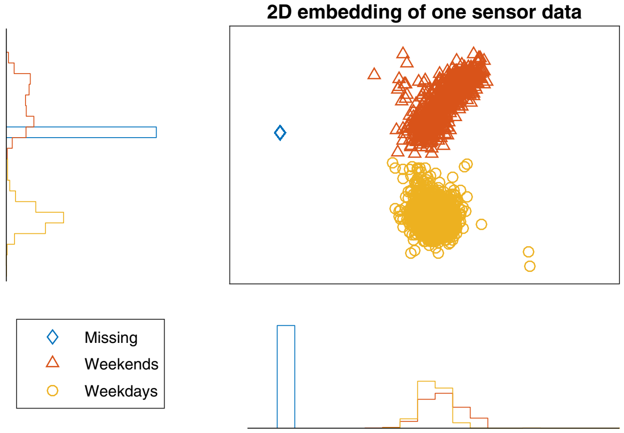

Figure 2: 2D embedding of a subset of data for one sensor. Large clusters are weekdays or weekends. The high density diamond cluster corresponds to days where the traffic volume is zero for an entire day (missing data).

Examining several samples from each cluster suggests the largest variance between days corresponds to whether the day is a working day or part of the weekend. The same can be observed in Figure 4 where the daily average traffic volume was computed for each day of the week, confirming the weekday – weekend assumption. The high density cluster (diamond) in Fig. 2 corresponds to days where the volume is zero. This may be due to sensor failure, maintenance operations or human processing errors. The number of days having such events were counted for each sensor and sorted:

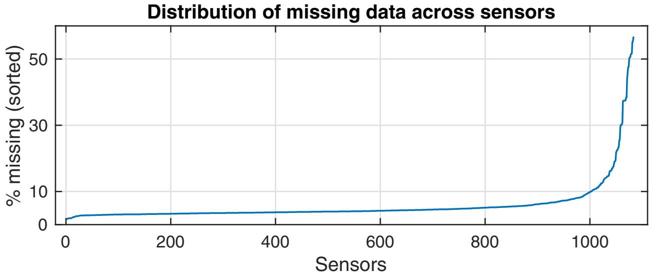

Figure 3: 92% of sensors have less than 10% missing data, while the rest can reach up to 56%. There are only 10 sensors without missing data.

In total there are 128,014 zero-valued days out of 2,203,772 (5.8%). The distribution differs across sensors. Approximately 92% of sensors have less than 10% missing data, while the rest of 8% can reach up to 56%. The cumulated averages per each day of the week show that for working days, the largest variance between sensors is the difference between noon and evening rush hour peaks. For some sensors the traffic peaks are higher at noon and lower in the evening while for others, the peaks might have the same amplitude. It is interesting to observe in Figure 4 that for Fridays and Saturdays there is a another volume peak at approximately 23:00. However, this pattern is not observable for all sensors. Another source of variance between sensors is a slight time shift between peaks (e.g. morning traffic peaks between 07:30 and 08:30). These shifts are likely to be a function of the sensor's proximity to the CBD. During weekends, the traffic pattern is not consistent across sensors, though typically there is a noon peak (12:00 – 13:00).

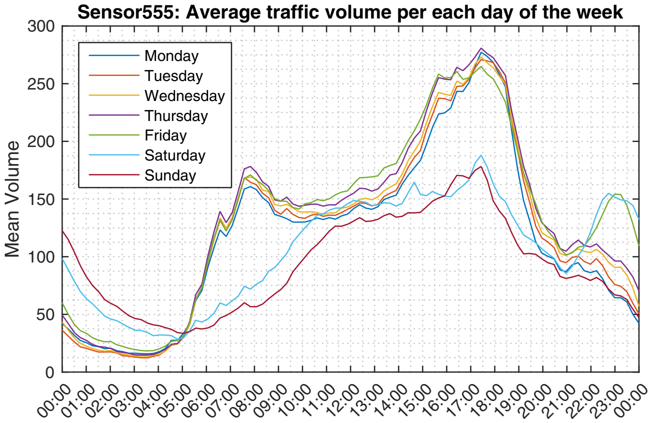

Figure 4: Average traffic over 6 years accumulated per day of the week. An outbound road. Evening peak is higher, when commuters depart.

Sporadically, weekend evening peaks are similar to a typical weekday peak. Another interesting observation is that even though some sensors have the same peak traffic profile across working days, the actual *volume* can differ across days. Each day could thus be modeled individually. A similar analysis was performed in [1], where the daily characteristics show similar patterns. The authors conclude that seasonal variance is mainly governed by school holidays.

Table I: Comparison of the VicRoads dataset with ones from literature.

Dataset	# Sensors	Timespan	Granularity	Total timepoints
VicRoads	1084	6 years	15 Min	211,562,112
[2]	837	3 months	5 Min	20,248,704
[3]	502	1 week	5 Min	1,012,032
[4]	52	31 days	5 Min	464,256
[5]	50	16 days	5 Min	230,400
[6]	22	24 days	5 Min	152,064
[7]	4	28 days	5 Min	32,256
[8]	4	10 hours	20 Min	>5,000
[9]	12	6 days?	5 Min	1,600
[10]	4	??	1 Min	--

IV. Problem Setting and Related Work

For prediction, the dataset was reshaped into a single continuous series per sensor. Each column S is a sensor, scaled to $\mathcal{D} \in [0, 1)$. This yielded a dataset $\mathcal{D} \in \mathbb{R}_+^{T \times S}$ where $S = 1,084$ streams and $T = 195,168$ time points. The first 70% time points were used for training while the last 30% was kept unaltered for testing in all experiments.

A. Spatiotemporal considerations

The term *n-step-ahead* refers to the number of points into the future for which predictions are made. *Prediction horizon* refers to the difference between the current time and the start of the prediction time and can be one-step-ahead or n-step-ahead. In our work, we only consider one-step-ahead predictions in the immediate step into the future. It is trivial to adapt to a larger prediction horizon. Simultaneous network-wide prediction can be modeled either as multiple individual learners or holistically as a multivariate learner that makes predictions on all measurement points simultaneously. We initially model each sensor individually. For one-step-ahead prediction, the dimension of the response variable (target)

$y_s(t) \subset \mathcal{D}^s(t)$ where $y_s \in \{0, 1\}$ is always one $|y_s| = 1$. The sliding window is moved forward one step at a time through the training set for all sensors simultaneously. The training data $x_s(t) \subset \mathcal{D}^s(t-1-w, t-1)$ with $x_s \in [0, 1)$ has a length of $|x_s| = w$ observations and is sampled for a particular sensor and a specific time frame, while the window is moved. f_s is the decision boundary, ε_s is the irreducible error and λ_s is the regularization term for sensor s. One possible interpretation is that we aim to solve the general least squares problem thus finding the optimal decision boundary $f_s \in \mathcal{H}_s$ (all sensors independently) that best separates the classes y_s using the data from x_s.

$$\underset{f_s \in \mathcal{H}}{\arg\min} \{ \| f_s(x_s + \varepsilon_s) - y_s \|_2^2 + \lambda \| f_s \|_2^2 \} \qquad (1)$$

Thus, there are S such equations that are solved simultaneously although *independently* during the training phase. In subsection VI-B we also share data between predictors.

B. Related research and datasets

Predominant methods in the literature are Autoregressive Integrated Moving Average models (ARIMA), Kalman filters, spectral methods and neural networks. A study [11] on short term traffic forecasting suggests that compared to neural networks, the other algorithms are less robust when congestion increases. The work in [12] suggests that this might be due to the smoothing of input data, which obscures the spatiotemporal correlations. In [13], the authors conclude that Big Data is paramount for increased performance.

ARIMA [14] are parametric models commonly used in time series prediction. SARIMA models are used to cope with seasonal effects. VARIMA models generalize univariate to multivariate and capture *linear* correlations among multiple time series. A VARIMA inspired [3] makes predictions as a function of both location and time of day. They report an average accuracy of 91.15 over a network of 500 sensors.

A study on autocorrelation on spatiotemporal data [6] concludes that ARIMA based models assume a globally stationary space-time autocorrelation structure and are thus incapable of capturing complex dynamics. Another ARIMA inspired algorithm [2] uses a parametric, space-time autoregressive threshold algorithm for forecasting velocity. The equations are independent and incorporate the MA (moving average) and a neighborhood component that adds information from sensors in close proximity. Lasso [15] is used for simultaneous prediction and regularization. The authors motivate their approach as a means of coping with computational intractability in the case where the number of sensors is larger than 300. In the next sections we show it is possible to tractably make accurate network-wide forecasts on 1084 sensors simultaneously.

Particle filter methods have been used for traffic state estimation on freeways [7], [10], in combination with other methods such as discrete wavelet transforms. Similar to [8], such datasets are quite different to ours: the focus is on high resolution time-series on short intervals. Freeway data is less complex and furthermore these algorithms are challenging to fine-tune [10]. As pointed out in [16] such methods are largely reactive. Moreover, particle filters are difficult to scale to large nonlinear road networks. A nonparametric (kNN) multivariate regression technique is evaluated in [16] for one-step-ahead forecasting. The term multivariate refers to the modeling of three types of measurements, namely velocity, volume and flow. The authors show that using data from multiple types of measurements increases performance.

Neural networks have been used extensively for short-term real-time traffic forecasting [4], [8], [9], [11], [12], [17], [18] where the focus is to predict on larger prediction horizons. However, the employed datasets are far too simple. In [4] a neural network is used for simultaneous forecasting at multiple points along a commuter's route (the route is set and prediction is done before the traveling starts), with an error averaging to 5 mph for a 30 minute route. Multiple univariate neural networks are used in [9] for prediction. Data from the past week, neighboring traffic and the day of the week is added as input in order to further improve performance. Recurrent neural networks have demonstrated better forecasting performance [8] at larger prediction horizons compared to feed-forward networks. Hybrid ARIMA and neural networks [19] have also been applied successfully.

V. Peak traffic volume prediction

Peak traffic prediction is modeled as binomial classification. A threshold was set $\omega = 0.85$ for each sensor and high volume $y_s(t) > \omega = 1$ was labeled as positive examples. This procedure resulted in an imbalanced set, since peak traffic is less frequent. Accuracy was used as a performance measure. It is intuitive to expect an accuracy of 85% as a lower bound. Several baselines were evaluated: daily average traffic patterns (Fig 4); means from the previous week and ARMA models. The highest accuracy 89.24 was recorded with the cumulated daily average method.

A. Initial algorithm selection

A subset of 15% of the sensors was selected randomly and eight algorithms were evaluated. The results are displayed in Table II along with average running times in seconds. There were no efforts made towards fine-tuning.

Table II: Network wide classification accuracy and average running time on a random subset (15%) of sensors. One independent predictor per sensor.

Algorithm	$w = 1$	$w = 5$	Seconds
Baseline	89.34		N/A
LogReg	92.54	**92.95**	2.5
FFNN	92.49	**92.86**	7.2
RUSBoost	90.19	***92.21***	*430.9*
LDA	91.99	***92.00***	0.1
Tree	***92.47***	*90.66*	*0.8*
SvmRBF	91.90	85.34	140.7
NB	91.61	84.00	13.4
kNN	85.06	81.18	42.2

The table above is sorted in descending order on the third column. With $w = 1$ the best accuracies are recorded for the same algorithms that do better with a larger window size, suggesting that these algorithms are more appropriate for the current task. Table II suggests which algorithms to eliminate from further consideration. There is only a marginal improvement for the first four algorithms, while the last three show a sudden decrease in performance, some even falling under the baseline. Increasing window size w should not dramatically decrease accuracy. A failure to effectively use the information gathered using a larger time sample suggests that the algorithm is less suitable for detecting, learning and predicting complex events.

B. The effect of increasing window size

Since some algorithms in Table II are slower with the same or worse accuracy, in the follow up experiments only logistic regression, Feed Forward Neural Networks (FFNNs) and classification trees (as a counter example) are considered. Results are shown in Figure 5 where the window size is increased even further with $w \in \{1, 5, 10, 20\}$.

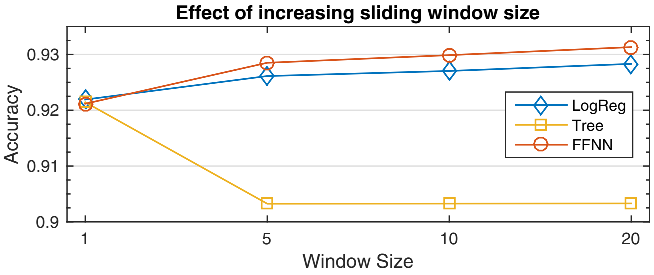

Figure 5: Increasing window size w results in better accuracy. Results for $w = 20$: neural network 93.13, logistic regression 92.83.

The accuracy increases linearly for $w > 1$. The behavior for logistic regression is similar to the FFNN. However, the neural network always outperforms the simpler logistic regression since it finds the decision boundary in a smaller nonlinear space. This implies that there must be an intrinsic lower dimensional space where the classes are better separable. The maximum accuracy for the neural network is 93.13 while for the linear algorithm it is 92.83, with $w = 20$ equivalent to looking back 5 hours. For a $w = 10$, the accuracies are 92.99 for the neural network and 92.70 for logistic regression. Elastic Net and Lasso [15] (L_1, $L2$ and $L1 + L2$) were succinctly evaluated for both a linear and quadratic combination of time points. Regularization can decrease variance at the expense of increasing bias. The contribution of each time point in either form is almost equal and thus regularization is not useful in the original space.

C. Exclusive Monday to Friday traffic prediction

From previous experiments it is clear that using more time steps from the past provides a more robust temporal context and thus results in better accuracy. Here, we ask whether this also holds for simpler periodic data, by considering prediction only on working days. This experiment is a follow up on the observed clusters in Fig. 2. It is possible that a larger window size captures more accurately the difference between work days and weekends, hence the better predictions. Towards evaluating this hypothesis, the weekends were removed from both the training and the testing set and the previous experiment was repeated. These results are not directly comparable to those in Figure 5. However, the majority of the literature is focused on Mon–Fri data, captured from less complex traffic networks.

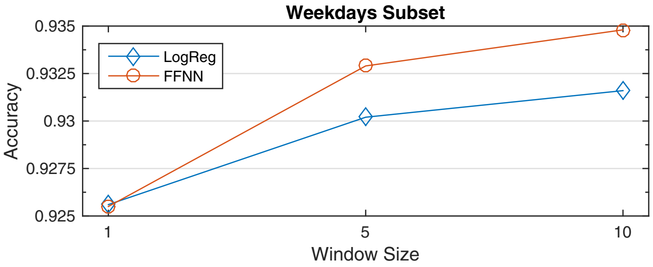

Figure 6: Weekends are removed. Larger window size w still results in better performance. Top accuracy (93.48, $w = 10$, FFNN) is better than if weekends are included (92.99).

The above figure shows that the hypothesis was false and indeed increasing the window size still has a great impact on prediction accuracy. Despite the fact that the greatest variance is observed between workdays and weekends (see Figure 2), simply capturing more temporal context is still beneficial, regardless if the largest source of variance is not present. For all following experiments the window size is set to 10, unless otherwise stated. We chose not to take a larger window size for computational reasons.

D. Augmenting missing data with context average trends

Missing data is one of the frequent problems of big data applications. This is also a significant characteristic of the current data set. In order to observe the impact of missing data on prediction accuracy, the zero values were replaced with the corresponding hourly sensor trend, cumulated per each day of the week. The results for logistic regression and the neural network are presented in Table III, along with the difference in accuracy Δ from the previous experiment (see Fig. 5). The augmented dataset is denoted by $\mathcal{D}_\mu$.

Table III: Adding mean trend values for missing data increases accuracy.

	Logistic Regression			FF Neural Network		
	$\mathcal{D}_\mu$	$\mathcal{D}$	Δ	$\mathcal{D}_\mu$	$\mathcal{D}$	Δ
$w = 1$	92.26	92.19	0.07	92.31	92.12	0.19
$w = 5$	92.68	92.61	0.07	92.91	92.85	0.06
$w = 10$	92.78	92.70	0.08	93.05	92.99	0.06

VI. Big Data versus Small Data

Collection of Big Data is essential, as it is not possible to know what questions will be asked in the future. What does Big Data mean for the current setting? Characteristic to our dataset, there are two dimensions to consider when addressing this question, namely time and space.

How much data is needed in order to generalize well? As one might guess, the traffic patterns are quite cyclical. Towards answering these questions, we repeat the previous experiments and: i) use less temporal data; ii) share data between classifiers, based on sensor proximity.

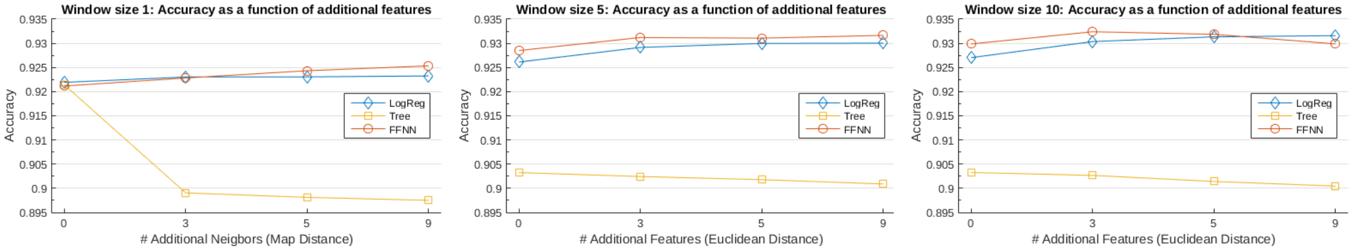

Figure 7: Additional data from 3, 5 and 9 closest sensors is added to each classifier. Best result thus far: 93.24% FFNN $w = 10$ $k = 3$.

A. Leveraging the temporal dimension of big data

In this section we examine the implications of using less data from the temporal dimension. Table IV shows accuracy as a function of the size of the training set for the logistic regression algorithm and the neural network. The first 10% (old data), last 10% and 25% (recent data) of the training set is selected for training, while the test set is kept the same.

Roughly, 100% of the training data accounts for traffic volume recorded over approximately 4 years, while for the validation set, it accounts for approximately 2 years. Then, 10% of the training data amounts to half a year, while 25% from the entire training set corresponds to data for one year.

Table IV: Big Data is relevant on the temporal dimension: accuracy decreases as the variety and volume of the full dataset is reduced.

	100%	1st Half Year	Last Half Year	Last Year
LogReg	92.70	89.86 (-2.84)	92.00 (-0.7)	92.22 (-0.48)
FFNN	92.99	89.90 (-3.09)	92.19 (-0.8)	92.41 (-0.58)

Using less data results in a decrease in performance and the drop is more abrupt as data become increasingly outdated. If data only from the first year is used for training (Table IV column 2), the accuracy decreases almost to the level of the baseline. Complex models are more likely to overfit. For the neural network, these effects are thus stronger since less data contains less variety.

B. Leveraging Big Data through sensor proximity

Thus far the network-wide prediction was modeled naively: one predictor per sensor was trained using data only from its own history. Traffic on a particular road is influenced by traffic in its proximity, hence the predictors should model this accordingly. We therefore proceed by including data from neighboring traffic for each predictor (still one predictor per sensor), based on the Euclidean distance between sensors, computed from geographical coordinates. This does not correspond to the actual city block distance it takes to navigate between sensors / roads. Some sensors have multiple coordinates, in which case the location is approximated to the average – a potential source of error.

The data from $k \in \{3, 5, 10\}$ neighboring sensors is added by simply concatenating it to the input data, resulting in a training vector of length $|x_s| = w \times k$ for each predictor. This is not necessarily the most efficient or best method of performing feature selection or simultaneous multivariate prediction. A better means of determining the correlations between traffic at each sensor is to perform spatial partitioning [20] and leverage the volume traffic data itself, instead of using map coordinates. However, this results in a fixed graph representation, while correlations between roads are likely to change throughout the day (e.g. peaks in Figure 4). This has been observed in [21], where univariate and multivariate methods are compared, also based on map coordinates. They note that the parameters for the multivariate modeling of traffic flow are not stationary. The same observation is made in a study [6] on traffic spatiotemporal correlations where the authors conclude that the autocorrelation structure changes both spatially and temporally, according to the traffic peaks, thus a non-stationary approach is preferable. We leave this for future work and experiment with the window size and number of additional feature vectors.

The results are shown in Figure 7, where the main observation is that adding data from proximity results in better predictions. The accuracy increases to 93.24% for the neural network, in the case where the window size ($w = 10$) and data from three neighboring sensors is used ($k = 3$), the highest accuracy recorded thus far. This result is also better than the case where $w = 20$ and $k = 0$ (for the neural network the accuracy was 93.13%, see Fig. 5).

This implies that proximity data and feature selection have more impact than simply increasing the window size, although both are beneficial. The pattern in Figure 7 is clear. As more time-points and more data from neighboring sensors are added, the performance increases. However, the result obtained using neural networks with a $w > 5$ and $k > 5$, is lower although very close to the best result obtained ($w = 10$ and $k = 3$). The effects of the curse of dimensionality thus become more evident as the length of the feature vector is increased to more than 30 time points. For logistic regression the performance still increases, although still less accurate than the neural network (a matter of regularization).

C. Performance as a function of location

The resulting test accuracies from the best logistic regression results are clustered (over all sensors) and the corresponding color coded prototypes are overlaid on the map in Figure 8. From the distribution of accuracy we can observe that prediction is more challenging in the extremities and the CBD. The geometry of road segments does not appear to have an impact on accuracy. Furthermore, there seems to be no connection between the direction of traffic (outbound vs inbound) and accuracy, since the clusters are distributed evenly. There are 150 sensors (7.2%) where the accuracy is one standard deviation below the mean (taken

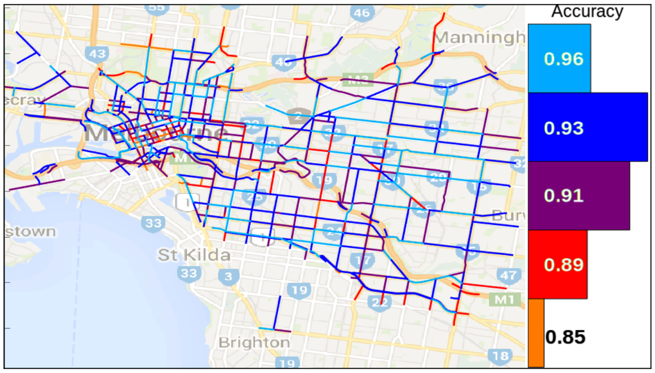

Figure 8: Accuracy distribution over the traffic network (logistic regression). The histogram on the right shows the relative size of each accuracy cluster.

over all sensors). Out of these, we examined the volume data for one sensor and observed that after the first three years the traffic volume is doubled. Thus, almost all traffic after the 3rd year is labeled as high volume. Permanent changes to the traffic rules can have a drastic impact on the distribution of the volume data for one sensor. Therefore, a threshold over the entire time series, introduces additional error.

VII. Discussion and Conclusion

In this paper, we have explored a novel big traffic data set for the application of traffic forecasting. The classification experiments show that increasing temporal context is beneficial, provided an appropriate representation and that including neighboring data further improves performance.

This is also consistent with the work in [4], [9] where however, the volume and complexity of the dataset is lower. Additionally, it was shown that using outdated or smaller volumes of data causes the prediction performance to drop, suggesting that the volume and variance of data is critical for Big Data applications. An additional increase in accuracy can also be obtained if the missing data is augmented with the average contextual traffic trend. The accuracy can be further increased if the goal is to forecast Mondays to Fridays only, or other subsets of data.

We suggest that for useful real-time predictions, the threshold that separates high and low traffic should be adapted dynamically according to the query point (sensor) and the time of the day. Setting an appropriate threshold can be considered a separate problem in itself. We believe that the thresholding procedure introduces more problems than it solves. As future work, we aim to focus on prediction of continuous values, the spatiotemporal correlations and feature selection, perform n-step-ahead prediction and further increase the prediction horizon. Additionally, since all the algorithms have been benchmarked in batch mode, not all can be deployed successfully or be as effective in online settings, where learning and prediction is simultaneous.

References

[1] R. Chrobok *et al.*, “Three categories of traffic data: Historical, current, and predictive,” in *Control in Transp. Syst., Proc. of the 9th IFAC Symp.*, 2000, pp. 250–255.

[2] Y. Kamarianakis *et al.*, “Real-time road traffic forecasting using regime-switching space-time models and adaptive lasso,” *Applied Stochastic Models in Business and Industry*, vol. 28, no. 4, pp. 297–315, 2012.

[3] W. Min *et al.*, “Real-time road traffic prediction with spatio-temporal correlations,” *Transp. Res. Part C: Emerging Tech.*, vol. 19, no. 4, pp. 606–616, 2011.

[4] J. Park *et al.*, “Real time vehicle speed prediction using a neural network traffic model,” in *Neural Networks (IJCNN), Int. Joint Conf. on.* IEEE, 2011, pp. 2991–2996.

[5] X. Min *et al.*, “Urban traffic network modeling and short-term traffic flow forecasting based on gstarima model,” in *Intell. Transp. Syst. (ITSC), 13th Int. Conf. on.* IEEE, 2010, pp. 1535–1540.

[6] T. Cheng et.al, “Spatio-temporal autocorrelation of road network data,” *Jrnl of Geographical Syst.*, vol. 14, no. 4, pp. 389–413, 2012.

[7] Y. Xie *et al.*, “Short-term traffic volume forecasting using kalman filter with discrete wavelet decomposition,” *Computer-Aided Civil and Infrastructure Eng.*, vol. 22, no. 5, pp. 326–334, 2007.

[8] H. Dia, “An object-oriented neural network approach to short-term traffic forecasting,” *European Jrnl of Oprn Res.*, vol. 131, no. 2, pp. 253–261, 2001.

[9] E.-M. Lee *et al.*, “Traffic speed prediction under weekday, time, and neighboring links‘ speed: back propagation neural network approach,” in *Advanced Intell. Comp. Th. and Apps.* Springer, 2007, pp. 626–635.

[10] Y. Wang *et al.*, “Real-time freeway traffic state estimation based on extended kalman filter: Adaptive capabilities and real data testing,” *Transp. Res. Part A: Policy and Practice*, vol. 42, no. 10, pp. 1340–1358, 2008.

[11] S. Clark *et al.*, “The use of neural networks and time series models for short term traffic forecasting: a comparative study,” in *European Transp., Highways And Planning 21st Annual Meeting*, vol. P363, 1993.

[12] V. Blue *et al.*, “Neural network freeway travel time estimation,” in *Intell. Eng. Syst. Artificial Neural Nets (Saint Louis, Mo.) Proc. of*, vol. 4, 1994.

[13] V. Arem *et al.*, “Recent advances and applications in the field of short-term traffic forecasting,” *Int. Jrnl of Forecasting*, vol. 13, no. 1, pp. 1–12, 1997.

[14] D. Asteriou *et al.*, *Applied econometrics*. Palgrave Macmillan, 2011, pp. 265–286.

[15] H. Zou *et al.*, “Regularization and variable selection via the elastic net,” *Jrnl of the Royal Stat. Soc.: Series B (Stat. Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[16] S. Clark, “Traffic prediction using multivariate nonparametric regression,” *Jrnl of Transp. Eng.*, vol. 129, no. 2, pp. 161–168, 2003.

[17] B. Smith *et al.*, “Short-term traffic flow prediction: neural network approach,” *Transp. Res. Record*, no. 1453, 1994.

[18] M. S. Dougherty *et al.*, “Short-term inter-urban traffic forecasts using neural networks,” *Int. Jrnl of forecasting*, vol. 13, no. 1, pp. 21–31, 1997.

[19] G. P. Zhang, “Time series forecasting using a hybrid arima and neural network model,” *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[20] T. Anwar *et al.*, “Spatial partitioning of large urban road networks,” in *EDBT*, 2014, pp. 343–354.

[21] Y. Kamarianakis *et al.*, “Forecasting traffic flow conditions in an urban network: comparison of multivariate and univariate approaches,” *Transp. Res. Record*, no. 1857, pp. 74–84, 2003.