Collaborative Analytics for Predicting Expressway-Traffic Congestion

Chee Seng Chong[α]
chongcs@ihpc.a-star.edu.sg

William-Chandra Tjhi[α]
tjhiwc@ihpc.a-star.edu.sg

Mingguang Li, Reuben[β]
reubenli@nus.edu.sg

Bong Zoebir[ψ]
zbbong@ntu.edu.sg

Tianyou Zhang[α]
zhangty@ihpc.a-star.edu.sg

Whye Loon Tung[θ]
wltung@hp.com

Yu Shyang Tan, Alan[θ]
yu-shyang.tan@hp.com

Kee Khoon Lee[α]
leekk@ihpc.a-star.edu.sg

Bu-Sung Lee, Francis[ψ]
ebslee@ntu.edu.sg

ABSTRACT
There are many ways to build a predictive model from data. Besides the numerous classification or regression algorithms to choose from, there are countless possibilities of useful data transformation prior to modeling. To assist in discovering good predictive analytics workflows, we introduced recently a collaborative analytics system that allows workflow sharing and reuse. We designed a recommendation engine for the system to enable matching of analytics needs with relevant workflows stored in repository. The engine relies on meta-predictive modeling of traffic-analysis workflow-characteristics. In this paper, we present a feasibility study of applying this collaborative analytics system to predict traffic congestion. Different ways to build predictive models from traffic dataset are pooled as shared workflows. We demonstrate that through dynamic recommendation of workflows that are suitable for the real-time varying traffic data, a reliable congestion prediction can be achieved. The promising results showcase that systematic collaboration among data scientists made possible by our system can be a powerful tool to produce very accurate prediction from data.

Categories and Subject Descriptors
H.2.8 [**Database Applications**]: Data mining
D.4.8 [**Performance**]: Modeling and prediction
J.7 [**COMPUTERS IN OTHER SYSTEMS**]: Real time

General Terms
Design, measurement, experimentation, verification

Keywords
Prediction/information markets, reputation and recommendation system

1. INTRODUCTION
A key issue to achieve accurate prediction is to find the most suitable modeling given a dataset. For example, in making product recommendation, Amazon.com could use different predictive models for new and regular customers. Off-the-shelf analytics suites offer numerous classification and regression operators for predictive modeling. In addition, the steps prior to applying classification/regression (e.g. data preprocessing) also affect the accuracy of prediction. Thus, the problem is of finding the right workflow, which is combinatorial. This problem becomes more critical in applications where data rapidly change. Predictive models then need to be regularly rebuilt so that prediction stays aligned with the current state of data. The application investigated in this paper is traffic congestion prediction for one of Singapore's main expressways, the Pan Island Expressway (PIE). The data involved are real-time (traffic and weather patterns streams) and dynamic (e.g. peak hour vs. normal hour), suggesting the need to rebuild predictive models to cater for the changing data characteristics. While it is possible to use the same predictive modeling despite data change, there is a merit in adapting to new data to get better accuracy [4], as supported by our experimental results. For this purpose, there is a need to pool predictive analytics workflows for selection. In [9], we proposed a collaborative analytics system for RapidMiner users to collectively share their workflows into a common repository. In this short paper, we present a feasibility study of applying this system for the PIE traffic congestion prediction.

Related works exist on leveraging collective analyses to address complex analytics tasks. Learning Experiment Database [21] publicly shares classifiers' accuracies on benchmark datasets, while myExperiments and Galaxy allow sharing of workflows developed from suites like Taverna and RapidMiner [3]. Users of these systems have to manually select workflows that are deemed suitable. Meta-mining builds models based on data characteristics to find accurate classifier [4]. Our work uses meta-miner as one of its components, but focuses on building a complete system for sharing and reuse of workflows. Automatic on-the-fly workflow construction [10] has the benefit of not limiting the search space to existing workflows in a repository. However, this search space can be very large. In contrast, in our collaborative analytics approach, the search space is limited to workflows contributed to the repository and recommendation is done rapidly by workflow selection. Future technology could see combination of automated and collaborative workflow construction/selection. On traffic prediction, ARIMA [5] can be efficient and accurate but limited to linear patterns and not sensitive to rapid variations in traffic

ICEC '12, August 07 – 08 2012, Singapore, Singapore

α Institute of High Performance Computing, Singapore
β Department of Geography, National University of Singapore
ψ School of Computer Engineering, Nanyang Technological University, Singapore
θ HP Labs Singapore

flow. Related techniques include exponential smoothing [2], regression [1, 20], state-space method [16, 19], maximum likelihood [13]. Artificial neural network (ANN) [14] and support vector regression (SVR) [6] are applied to capture non-linear patterns. Hybrid ANN models [7, 18] generally perform very well. However, most ANN and SVR-based models have high computational complexity. Our collaborative analytics can leverage these individual solutions and select the most suitable one for a given dataset.

We contribute a new workflow recommendation framework by matching analytics needs with a collection of predictive models. Figure 1 shows the framework with traffic congestion prediction application. The aim is to predict the time length of traffic congestion in a road segment.

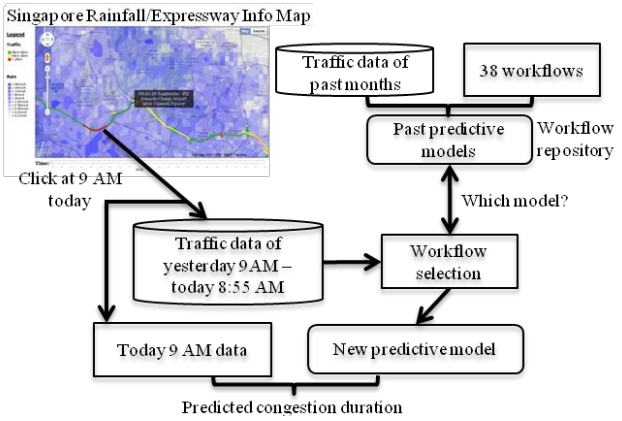

Figure 1. Process flow of proposed method

By demonstrating the feasibility of collaborative analytics on real-world traffic application, we pave the way for future researches that promote dynamic analytics system leveraging community-driven method contributions.

2. PROPOSED METHOD

The framework in Figure 1 uses traffic patterns from a day before the current time, *t*, until *t-5 minutes* as training data. Instead of training the data by a single classifier/regressor, the system consults a repository of past predictive modeling workflows to search for a reasonable workflow. A new predictive model is then built by applying the recommended workflow on the training data. Prediction is made by applying model on data instance at *t*. The repository is a collection of classifier-based workflows designed for traffic data. We now detail the key elements of the framework.

2.1 Data

Traffic data were obtained from the traffic.smart@OneMotoring portal [11] which serves its data through the onemap.sg map server. In addition, rainfall intensity on the road was obtained from the Singapore NEA marine services weather radar webpage [15]. Reverse data engineering was performed to synchronize the traffic and the rainfall data, which come in the form of images. The dataset features used are: 1)road segment, 2)traffic intensity (congested, slow, or clear), 3)east side neighboring segment traffic, 4)west side neighboring segment traffic, 5)rainfall intensity, 6)hour of day, and 7)congestion duration (i.e. the time it takes "congested" or "slow" to become "clear"). Congestion duration becomes the label of predictive models. For the purpose of classification, the originally continuous label was discretized into 3 categories with bin sizes automatically decided based on the frequency of the records. The repository of existing analyses was populated by workflows processing historical data from 13-17 May 2011. Each day contributed 2 datasets: AM and PM, for a total of 10 historical datasets.

2.2 Workflows

The repository of existing analyses is populated by 38 workflows representing the different ways predictive modeling can be performed to analyze traffic data. The 38 workflows fall into 5 main categories (as shown in Table 1).

Table 1. Five categories of workflow

Workflow template	Description
Per-segment model (batch)	One model per segment, use all data at once
Per-segment model (incremental)	One model per segment, use one data at a time
Global model (batch)	One model for all segment, all data at once
Global model (incremental)	One model for all segments, one data at a time
Time-series	Based on traffic trend per segment

For *per-segment* models, a predictive model is developed for each road segment based on the training data specific to that segment. For *global* models, one predictive model is built based on training instances regardless of road segments. In the *batch* models, classification is performed by considering all the training dataset simultaneously. An *incremental* model is initially built using a 50% sample of training dataset. The remaining 50% training instances are then randomly used individually to incrementally update the model. For time-series workflows, predictive models are built in a *per-segment* way, with training data in the form of univariate time-series on congestion duration. Prediction is made by performing regression with time-window of 10 data points. Finally, variations of workflow from each of the 5 categories are derived by substituting the classification/regression operators with alternative operators in RapidMiner. The non-time-series workflow categories use classification instead of regression so that the prediction output is in the form of time ranges. For the time-series, it is not natural to make prediction by classification, and therefore it is left as a regression problem.

These five workflow categories are not the only ways to build predictive model from the traffic data. Other variations, such as by replacing the classifiers in the non-time series workflow categories by a regressor, will be investigated in the future.

2.3 Recommendation engine

Given a training data, the recommendation engine needs to find from the repository, workflows that can produce good predictive models. To realize this, there is a need to define features that capture the workflow characteristics pertaining to historical data in the repository and the query training data. With these features, a meta-regressor can be built to predict a workflow's performance on the query training dataset, which in turn can be used to decide which workflow to recommend.

A natural choice to characterize a workflow pertaining to the historical data is by using its accuracies across the different historical datasets. Such an approach however cannot be used to characterize the query training data without actually executing the workflows on the query training data. If the latter is performed,

there is no longer need for the recommendation engine as the best workflow has been discovered by brute force.

To sidestep this issue, characterizing a workflow by using "proxy" accuracies is adopted. First each dataset (historical and query training dataset) is characterized by the landmarking approach. In a landmarking approach, simple classifiers are chosen as landmarkers and their classification accuracies on the query dataset become features. We follow the justification and approach taken by [17] and selected seven landmarker classifiers: 1K nearest neighbor, linear discriminant, worst node, decision node, average node, naïve bayes and randomly chosen node. Given a query training dataset, its top 3 neighboring historical datasets can then be found from the similarities measured based on the landmarking features. A workflow's characteristics pertaining to the query training dataset are then defined as the workflow's accuracies on the top 3 neighboring historical datasets. The same approach is used to derive the workflow characteristics pertaining to each individual historical datasets.

With the workflow characteristics defined, a support vector regression [8] is then used to predict the accuracy of a workflow when run on the query training dataset. Figure 2 shows in 2 stages the steps used by the recommendation engine.

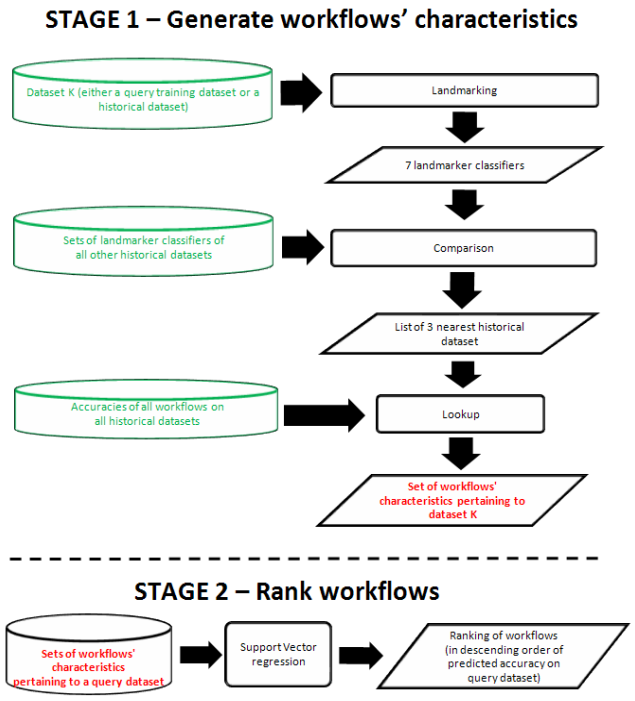

Figure 2. Steps used by the recommendation engine

3. RESULTS AND DISCUSSIONS

3.1 Experimentation Methodology

The East-bound direction of Pan Island Expressway was considered for our showcase. Three query training datasets were made available: for 1st and 2nd June, and 25th November 2011. RapidMiner version 5.1.011 was used for predictive model design, execution, and evaluation. Prediction accuracy was used as an evaluation measure (i.e. specifically for classification, F-measure was adopted).

Fine-tuning of predictive model parameters is done by looking up from the public Learning Experiment Database (LED) [12]. By comparing landmarking features, a dataset from LED that is closest to ours is found. Then, the most optimal parameters for all the different predictive models recorded in LED are extracted. There is however no guarantee that LED has captured the truly optimal parameters. Another issue is that there are only a few predictive models with optimal parameters recorded in LED. For those that are not, default parameters of RapidMiner were used.

The preparation step in the experiment is to have the 38 workflows evaluated against each of the historical datasets to obtain their accuracies. Then, all historical and query training datasets are passed through the landmarking module to generate their respective sets of 7 landmarker classifiers. The accuracies and landmarker classifiers information is used to generate the sets of workflow characteristics pertaining to a query training dataset, as shown in STAGE 1 of recommendation engine in Figure 2. In STAGE 2, a support vector regression returns top-5 ranked workflows in terms of predicted accuracy. STAGEs 1 & 2 are repeated for each query training dataset.

Finally, to judge the performance of the recommendation engine, all workflows are evaluated against each query training dataset to obtain the actual accuracies. The accuracy reflects classification/regression accuracy of the selected workflow using 5-fold cross-validation. With this, we examine how the recommended workflows stack against the others.

3.2 Results and Discussions

Table 2 lists the recommended workflows together with their actual accuracies when subsequently evaluated against their respective query training dataset. Figure 3 shows the accuracy of the top recommended workflow against the distribution of accuracies of all the 38 workflows. The recommendation engine returns an accuracy that is well within the vicinity of that of the best performer. This result shows that collaborative analytics can be a powerful tool to select from a pool of candidate workflows a reasonably effective workflow to yield very accurate prediction from data.

Table 2. Workflows returned by recommendation engine

Date	Workflow	Actual accuracy
1 June	**Time-Series Neural Net**	**0.845**
	Time-Series Opt SVM	0.723
	Batch Global SVM LIBSVM	0.455
	Batch Global Rule Induction	0.418
	Batch Global Opt SVM LIBSVM	0.455
2 June	**Time-Series Opt SVM**	**0.797**
	Time-Series SVM	0.754
	Time-Series Neural Net	0.797
	Batch Global k-NN	0.568
	Batch Global Rule Induction	0.552
25 Nov	**Time-Series Opt SVM**	**0.722**
	Time-Series Neural Net	0.770
	Time-Series SVM	0.782
	Batch Global Opt SVM LIBSVM	0.493
	Batch Global SVM LIBSVM	0.450

It is interesting to observe that time-series type of workflows feature prominently on top of the list of recommendations. This suggests that by focusing on the trend of the traffic on a particular road segment, time-series workflows seem to be most effective. This is in contrast to non-time-series workflows which also consider rain and neighboring segments' traffic condition. This extra information seems to be noise as it contributes negatively to the accuracies of the models.

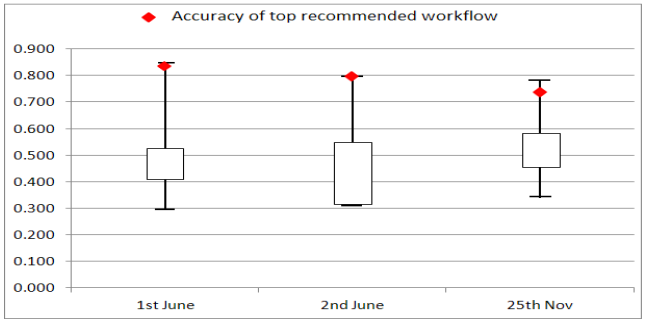

Figure 3. Accuracies of the 3 top recommended workflows compared to the accuracy distribution

4. CONCLUSIONS

We have presented a feasibility study on predicting traffic congestion by collaborative analytics. Instead of the conventional approach of building a fixed set of models to make the prediction, our approach leverages collectively shared predictive analytics workflows to dynamically apply the most reasonable models for given characteristics of data. To automate the workflow updating process, we have proposed a recommendation engine based on characterization of data and workflows. Our results show that our collaborative analytics can recommend workflows that give accurate prediction of traffic congestion. This signifies the potential of engaging systematically the community of data scientists to support very accurate data-backed prediction.

Extensive experiments to test the robustness of collaborative analytics are currently ongoing. Some interesting directions include adding semantic descriptions for data/workflow characterization and developing a sustainable model to involve the community in collective sharing of analytics workflows.

5. REFERENCES

[1] S. Clark. Traffic prediction using multivariate nonparametric regression, *Journal of transportation engineering*, vol. 129, p. 161, 2003.

[2] E. S. Gardner Jr. Exponential smoothing: The state of the art, *Journal of Forecasting*, vol. 4, no. 1, pp. 1–28, 1985.

[3] B. Giardine, C. Riemer, R.C. Hardison, R. Burhans, L. Elnitski, P. Shah, Y. Zhang, D. Blankenberg, I. Albert, J. Taylor, W. Miller, W.J. Kent, and A. Nekrutenko. Galaxy: A platform for interactive large-scale genome analysis,·*Genome Res*. 2005. 15: 1451-1455.

[4] C. Giraud-Carrier. Metalearning-a tutorial, *7th Int. Conf. on Machine Learning and Applications*,2008.

[5] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said. Short-term prediction of traffic volume in urban arterials, *Journal of Transportation Engineering*, vol. 121, p. 249, 1995.

[6] W. C. Hong. Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm, *Neurocomputing*, 2011.

[7] X. Jiang and H. Adeli. Wavelet Packet-Autocorrelation Function Method for Traffic Flow Pattern Analysis, *Computer-Aided Civil and Infrastructure Engineering*, vol. 19, no. 5, pp. 324–337, 2004.

[8] T. Joachims. Optimizing Search Engines Using Clickthrough Data, *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 133-142, 2002.

[9] H. Kasim, Terence Hung Gih Guang, Li Xiaorong, William Chandra Tjhi, Lu Sifei and Wang Long. A Cloud-Based Workflow Management Solution for Collaborative Analytics, *ICSOC Demo Track 2011*.

[10] J.U. Kietz, F. Serban, A. Bernstein, S. Fischer. Data Mining Workflow Templates for Intelligent Discovery Assistance and Auto-Experimentation, *European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases Workshop*, pp. 1-12, 2010.

[11] Land Transportation Authority of Singapore. traffic.smart portal, http://www.onemotoring.com.sg/publish/onemotoring/en/on_the_roads/traffic_smart.html.

[12] Learning Experiment Database. http://expdb.cs.kuleuven.be/expdb/index.php.

[13] W. H. Lin. A Gaussian maximum likelihood formulation for short-term forecasting of traffic flow, in *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pp. 150–155, 2001.

[14] A. Nagare and S. Bhatia. Traffic Flow Control using Neural Network, *Traffic*, vol. 1, no. 2, 2012.

[15] National Environmental Agency of Singapore. Weather Information Portal – Weather Radar Information, http://www.weather.gov.sg/wip/c/portal/layout?p_l_id=PUB.1023.5.

[16] I. Okutani and Y. J. Stephanedes. Dynamic prediction of traffic volume through Kalman filtering theory, *Transportation Research Part B: Methodological*, vol. 18, no. 1, pp. 1–11, 1984.

[17] F. Shafait, M. Reif, C. Kofler, T.M. Breuel. Pattern Recognition Engineering. *RapidMiner Community Meeting and Conference*, 2010.

[18] A. Stathopoulos, L. Dimitriou, and T. Tsekeris. Fuzzy modeling approach for combined forecasting of urban traffic flow, *Computer-Aided Civil and Infrastructure Engineering*, vol. 23, no. 7, pp. 521–535, 2008.

[19] A. Stathopoulos and M. G. Karlaftis. A multivariate state space approach for urban traffic flow modeling and prediction, *Transportation Research Part C: Emerging Technologies*, vol. 11, no. 2, pp. 121–135, 2003.

[20] H. Sun, H. X. Liu, H. Xiao, R. R. He, and B. Ran. Use of local linear regression model for short-term traffic forecasting, *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1836, no. -1, pp. 143–150, 2003.

[21] J. Vanschoren, H. Blockeel, B. Pfahringer, G. Holmes. Experiment databases - A new way to share, organize and learn from experiments. *Machine Learning*, in-press, 2011.