Methodology for real-time congestion forecasting based on Feature Engineering

Yana Barsky[1], Ayelet Gal-Tzur[2], Shlomo Bekhor[3]

Motivation

One of the goals of local and metropolitan transportation authorities is to efficiently manage traffic flow using their signalized road network. Traffic Management (TM) aims to maximize the use of existing infrastructure, improve service level and ensure the safety of all road users, reduce environmental pollution and minimize the expose of citizens and visitors to excessive levels of noise.

Congestion on the approaches to signalized intersections is a common phenomenon for which traffic management actions are intended to provide a fast and efficient solution. TM actions are carried out through municipal urban traffic management systems in accordance with their urban transportation policy.

Early detection of congestion formation on approaches to signalized intersections enables the activation of an alternative Traffic Signal Plan (TSP), which changes the signal settings by prolonging green duration of specific approaches, attempting to mitigate the impending congestion or minimizing its negative effect. This strategy, known as Proactive Traffic Management (PTM), can be implemented by using congestion prediction model forecasting outputs, trained on historical traffic data measured from traffic sensors.

The implementation of a proactive strategy reduces the capacity in "competing" approaches to a congested approach, and may interrupt the free flow movement in them. Hence, a high degree of prediction accuracy is a necessary condition for the effective implementation of such measure. Signalized intersections in urban arterials typically exhibit rapid-intensive fluctuations in short periods of time, which poses a challenge to differentiate between momentary congestion, and Stable Congestion (SC).

With the evolution of computational intelligence, the challenge of reliable traffic forecast modelling on signalized urban arterials has led to increasing use of data driven methods. Data driven methods are based on machine learning algorithms, offering a self-learning pattern recognition techniques, in which a prediction model is created by training the algorithm with historical dataset.

Feature Engineering

A crucial factor affecting the prediction quality of a model is the way that historical data is used as input to the algorithm in the training phase. Despite the recognition of the importance of the process of building input variables incorporating domain expert knowledge (a process called Feature Engineering) (Domingos, 2012), to the best of our

[1] Transportation Research Institute, Technion, Israel

[2] School of Engineering, Ruppin Academic Center

[3] Faculty of Civil and Environmental Engineering, Technion, Israel

knowledge there is no structured methodology for implementing this process for short-term traffic forecasting.

By using domain knowledge, Feature Engineering is the process of constructing features that better represent the underlying problem to a machine learning algorithm, resulting in a stable prediction model, i.e. robust to changes in the training dataset. Another advantage of engineered features is that it allows using less complex models, which improves running times and are easier to understand and maintain.

This paper presents a Feature Engineering based methodology for real-time congestion forecasting on approaches to urban signalized intersections. The proposed methodology is an iterative process aiming to find at each iteration a set of features, based on incorporation of data mining techniques (Random Forest) and traffic knowledge in building and selecting the most promising features, that result in an improved congestion prediction accuracy and higher robustness to changes in dataset. In the paper we will describe in detail the algorithm and the literature on the subject. In this abstract, we present results of a test case using a real dataset.

Test Case

A signalized sub-network (Figure 1), comprised of the congested approach (Link 171172) on a major arterial in Tel Aviv-Yafo and five surrounding road sections, was selected for testing the developed methodology. Loop detectors are located in all lanes of the links within the sub-network.

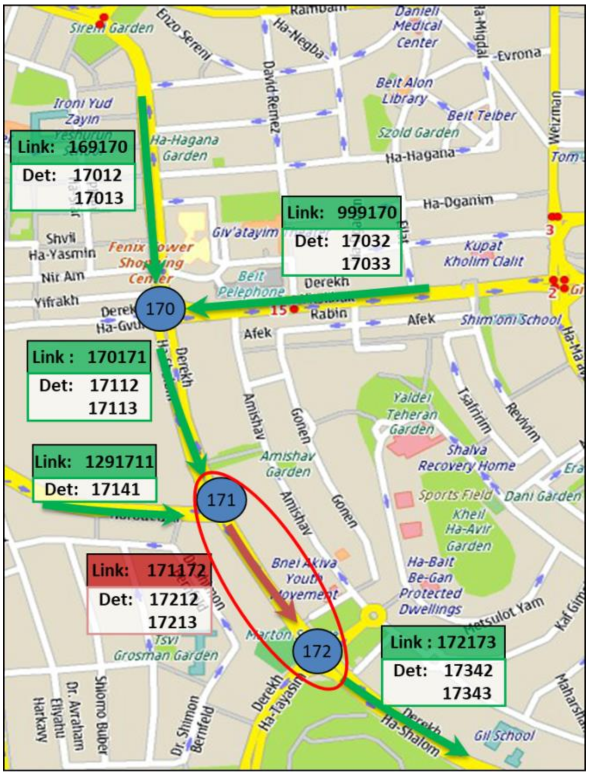


Figure 1 The sub-network

Historic traffic data were collected for a three and a half months, on weekdays during the peak hours 15:00-19:00 (for every two-minute interval), for each of the four upstream links, the congested link, and one downstream link detectors. The data include volume and occupancy values for each detector.

The implementation of PTM strategies requires a quantifiable and clear definition of congestion. Specifically, we need to differentiate between free flow (FF), momentary congestion, and Stable Congestion (SC). We are interested to identify SC, which justifies intervention in routine traffic management, i.e. switching to alternative TSP. The definitions are based on the level of service (LOS) calculated for the approach under interest.

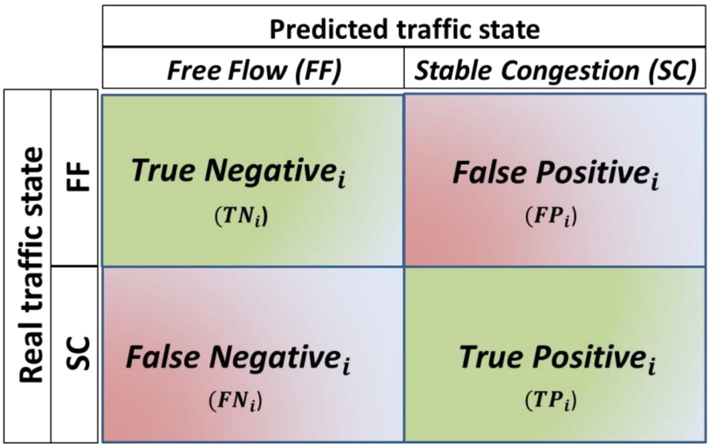
The 3.5 months of raw data were processed in order to construct a database of historic traffic states that included examples of events reflecting SC and FF. Analysing historic LOS data from an approach under interest revealed a total of 155 instances of SC. To ensure an effective machine learning process, reasonable balance had to be maintained between the two traffic states in the database. In order to address the challenge of reliable SC prediction, the database has to contain various different examples of FF traffic states (e.g. immediately before and/or immediately after SC states). Based on those considerations, 465 instances of FF were added to the database.

In order to represent the evolving nature of traffic states, for each instance of SC and FF, loop data from three time intervals (six minutes) from all lanes in a sub-network before the beginning of traffic state were used to construct each historic example. Hence the historical examples database contains 66 input variables for each example: volume and occupancy from each of the 11 loop detectors in three time intervals before the beginning of traffic state.

For the machine learning technique, a decision tree method was selected. The reason for choosing decision trees results is because they provide a set of logical If-Then rules, based on systematic reasoning which may be validated by human inspection, easily interpretable by the staff operating and maintaining the urban network control system.

Results

In order to assess the quality of SC prediction and the contribution of Feature Engineering, at each iteration of proposed methodology the database was partitioned 100 times to different training and testing sets. For each partition, decision tree was trained based on $Training\ Set_i$ and the trained model was tested on $Testing\ Set_i$, $i \in \{1,2,3,..,100\}$, resulting in a confusion matrix for each partition i , Figure 2.



		Predicted traffic state	
		Free Flow (FF)	*Stable Congestion (SC)*
Real traffic state	FF	*True Negative*$_i$ (TN_i)	*False Positive*$_i$ (FP_i)
	SC	*False Negative*$_i$ (FN_i)	*True Positive*$_i$ (TP_i)

Figure 2 Confusion Matrix for partition i

Based on a general structure of a confusion matrix, we rely on two conventional Performance Indicators (PI) to assess the quality of each decision tree:

$$False\ Positive\ Rate = \frac{False\ Positive}{False\ Positive + True\ Negative}$$

$$True\ Positive\ Rate = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

For the purpose of this paper, we developed four types of Performance Indicators to assess the quality of SC prediction at each iteration:

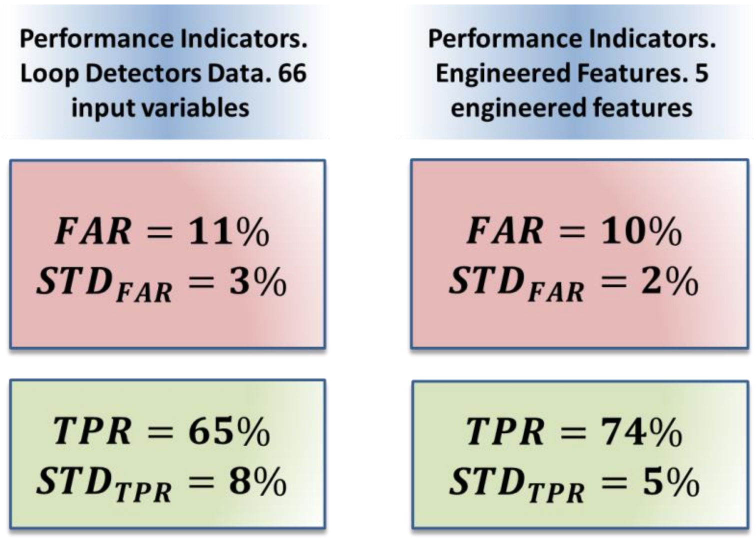
$$False\ Alarm\ Rate(FAR) = Avg_{i\in\{1,2,\ldots,100\}}\,(\frac{FP_i}{FP_i + TN_i})$$

$$Standard\ Deviation\ of\ False\ Alarm\ Rate(STD_{FAR}) = Std_{i\in\{1,2,\ldots,100\}}\,(\frac{FP_i}{FP_i + TN_i})$$

$$True\ SC\ Rate(TSCR) = Avg_{i\in\{1,2,\ldots,100\}}\,(\frac{TP_i}{TP_i + FN_i})$$

$$Standard\ Deviation\ of\ True\ SC\ Rate(STD_{TSCR}) = Std_{i\in\{1,2,\ldots,100\}}\,(\frac{TP_i}{TP_i + FN_i})$$

The final results of applying a developed methodology in comparison with performance indicators based on initial set of variables are presented in Figure 3.



Performance Indicators. Loop Detectors Data. 66 input variables	Performance Indicators. Engineered Features. 5 engineered features
$FAR = 11\%$ $STD_{FAR} = 3\%$	$FAR = 10\%$ $STD_{FAR} = 2\%$
$TPR = 65\%$ $STD_{TPR} = 8\%$	$TPR = 74\%$ $STD_{TPR} = 5\%$

Figure 3 Comparison of Performance Indicators

Applying a proposed methodology results in reduction of input variables from 66 to 5, along with significant (statistically tested) improvement in all PI values, yielding a higher classification accuracy and stability of prediction model.

Reference

P. Domingos, "A few useful things to know about machine learning", Communications of the ACM, 2012.