Traffic Flow Prediction With Big Data: A Deep Learning Approach

Yisheng Lv, Yanjie Duan, Wenwen Kang,
Zhengxi Li, and Fei-Yue Wang, Fellow, IEEE

Presented by: Bharath Kumar N B, June 2, 2017

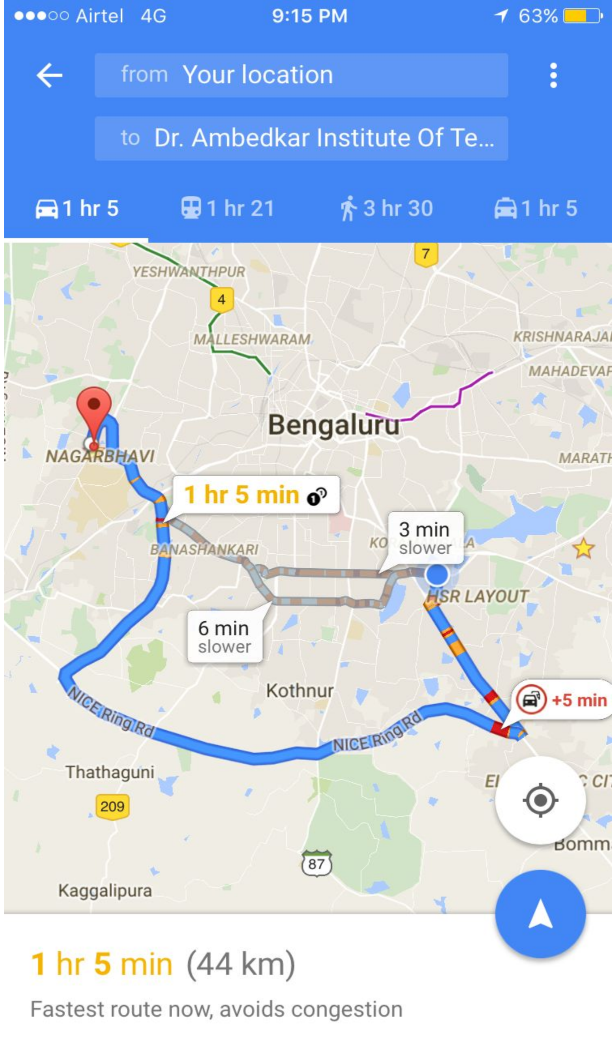
Outline

- Overview and Motivation
- Literature Survey
- Methodology
  - Autoencoder
  - Stacked Autoencoder(SAE)
  - Training Algorithm
- Experiments
  - Data description
  - Performance
- Conclusion

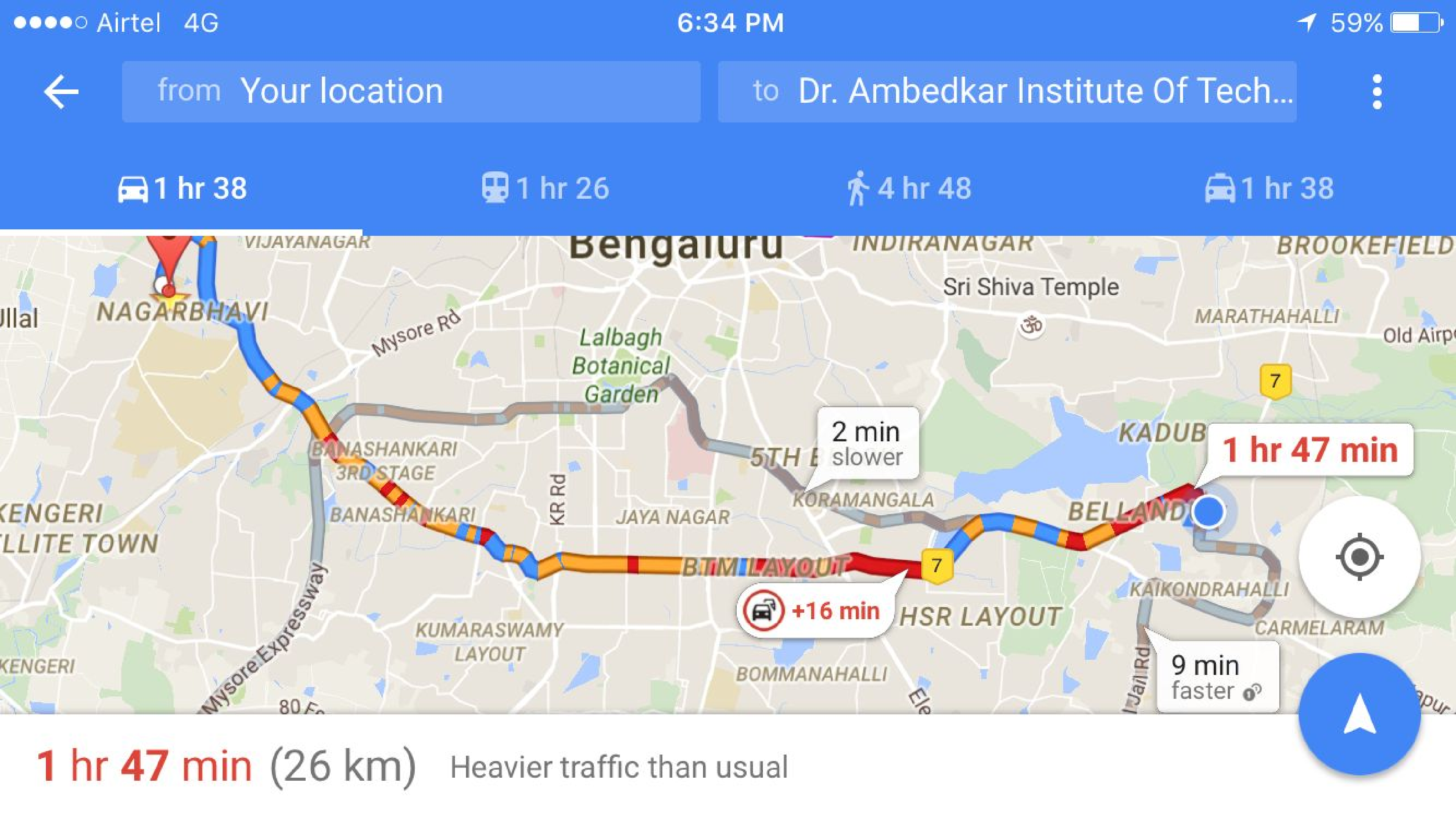
Overview and Motivation

Motivation

- Accurate and timely traffic flow information needed for
    - individual travelers
    - business sectors
    - government agencies.
- It has the potential to help
    - road users make better travel decisions
    - alleviate traffic congestion
    - reduce carbon emissions
    - improve traffic operation efficiency.



Airtel 4G
9:15 PM
63%
from Your location
to Dr. Ambedkar Institute Of Te...
1 hr 5
1 hr 21
3 hr 30
1 hr 5
YESHWANTHPUR
MALLESHWARAM
KRISHNARAJA
MAHADEVA
Bengaluru
NAGARBHAVI
MARATH
1 hr 5 min
3 min
slower
BANASHANKARI
HSR LAYOUT
6 min
slower
Kothnur
+5 min
NICE Ring Rd
NICE Ring Rd
Thathaguni
209
87
Kaggalipura
1 hr 5 min (44 km)
Fastest route now, avoids congestion


Airtel 4G
6:34 PM
59%
from Your location
to Dr. Ambedkar Institute Of Tech...
1 hr 38
1 hr 26
4 hr 48
1 hr 38
Bengaluru
VIJAYANAGAR
INDIRANAGAR
BROOKEFIELD
Sri Shiva Temple
NAGARBHAVI
Mysore Rd
Lalbagh Botanical Garden
MARATHAHALLI
2 min slower
KADUB
1 hr 47 min
BANASHANKARI 3RD STAGE
KORAMANGALA
BELLANDU
KENGERI
LLITE TOWN
BANASHANKARI
KR Rd
JAYA NAGAR
BTM LAYOUT
+16 min
HSR LAYOUT
KAIKONDRAHALLI
CARMELARAM
Mysore Expressway
KUMARASWAMY LAYOUT
BOMMANAHALLI
9 min faster
1 hr 47 min (26 km) Heavier traffic than usual

Overview and Motivation

Motivation

- Existing traffic flow prediction methods
    - use shallow traffic prediction models
    - still unsatisfying for many real-world applications.

Overview and Motivation

- Real-time traffic data collected from
  - Inductive loops
  - Radars
  - Cameras
  - Mobile Global Positioning System
  - Crowdsourcing
  - Social media, etc
- With the widespread traditional traffic sensors and new emerging traffic sensor technologies, traffic data are exploding and we have entered the era of **big data transportation**.

Overview and Motivation

A new approach

- Deep-learning based
  - As a traffic flow process is complicated in nature, deep learning algorithms can represent traffic features without prior knowledge, which has good performance for traffic flow prediction.
- A stacked Autoencoder model is used to learn features
  - Trained in greedy layerwise fashion.
  - Considers the spatial and temporal correlations inherently.

Literature Review

The traffic flow prediction problem can be stated as:

Let X_i^t denote the observed traffic flow quantity during the t^{th} time interval at the i^{th} observation location in a transportation network. Given a sequence $\{X_i^t\}$ of observed traffic flow data, $i = 1, 2, \ldots, m$, $t = 1, 2, \ldots, T$, the problem is to predict the traffic flow at time interval $(t+\Delta)$ for some prediction horizon Δ.

Literature Review

- Previous prediction approaches can be grouped into three categories
    - Parametric techniques
        - time-series models, Kalman filtering models, etc
    - Nonparametric methods
        - k-nearest neighbor (k-NN) methods, artificial neural networks (ANNs), etc
    - Simulations
        - use traffic simulation tools to predict traffic flow.

Literature Review

Parametric

- A widely used technique to the problem of traffic flow prediction is based on time-series methods.
- ARIMA - Autoregressive Integrated Moving Average
    - Kohonen- ARIMA (KARIMA)
    - subset ARIMA
    - ARIMA with explanatory variables (ARIMAX)
    - vector autoregressive moving average (ARMA)
    - space–time ARIMA
    - seasonal ARIMA (SARIMA)
- ARIMA and SARIMA algorithms perform reasonably well during normal operating conditions but do not respond well to external system changes.

Literature Review

Non-Parametric

- k-NN method performed comparably with but not better than the linear time-series approach.
- Bayesian network approach
- Online learning weighted Support Vector Regression (SVR)
- Various ANN models

Literature Review

Hybrid Methods

- Moving average (MA), Exponential smoothing (ES), ARIMA, and Neural Network (NN) models.
  - The MA, ES, ARIMA models were used to obtain three relevant time series that were the basis of the NN in the aggregation stage
- Different linear genetic programming, multilayer perceptron, and fuzzy logic (FL) models
- ARIMA model with the expectation—maximization and cumulative sum algorithms
- Adaptive hybrid fuzzy rule-based system approach

Literature Review

Conclusion

- It is difficult to say that one method is clearly superior over other methods in any situation.
- Reason
  - the proposed models are developed with a small amount of separate specific traffic data
  - the accuracy of traffic flow prediction methods is dependent on the traffic flow features embedded in the collected spatiotemporal traffic data.
- In general literature shows promising results when using NNs, which have good prediction power and robustness.

Methodology

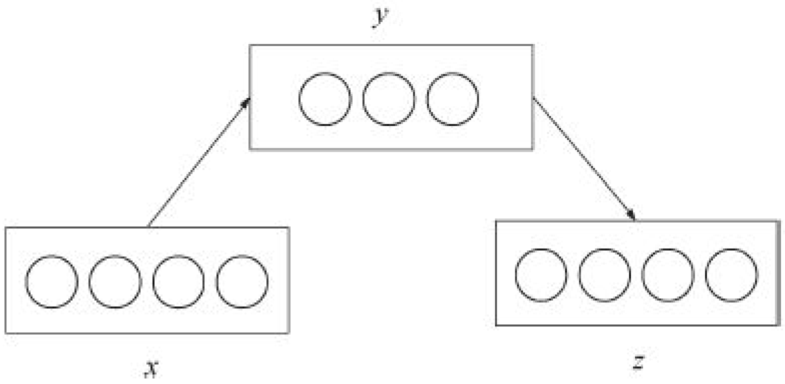
Autoencoder

- An autoencoder is a Neural Network that attempts to reproduce its input, i.e., the target output is the input of the model.
- The aim of an autoencoder is to learn a representation (encoding) for a set of data, typically for the purpose of dimensionality reduction.[1]

[1] Source: https://en.wikipedia.org/wiki/Autoencoder

Methodology

Autoencoder



Methodology

Autoencoder

- An autoencoder always consists of two parts, the encoder and the decoder, which can be defined as transitions

$$y(x) = f(W_1 x + b)$$
$$z(x) = g(W_2 y(x) + c)$$

- minimizing reconstruction error L(X, Z), we can obtain the model parameters, which are here denoted as θ

$$\theta = \arg\min_{\theta} L(X, Z) = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} \left\| x^{(i)} - z\left(x^{(i)}\right) \right\|^2.$$

Methodology

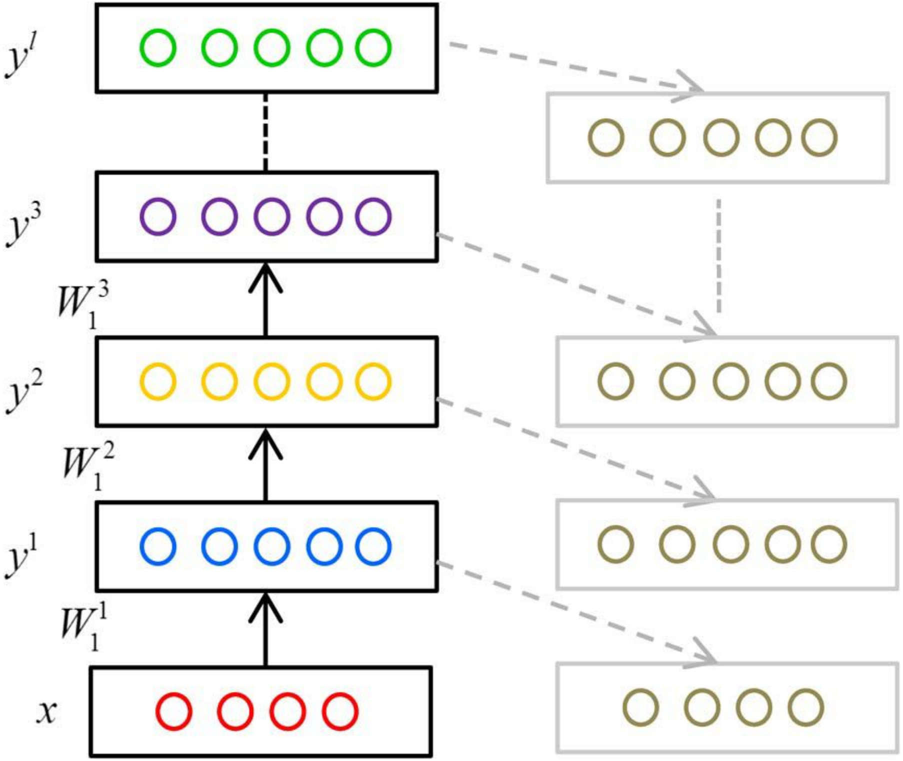
Stacked Autoencoders

- A SAE model is created by stacking autoencoders to form a deep network by taking the output of the autoencoder found on the layer below as the input of the current layer.
- A standard predictor has to be added on the top layer
    - In this paper, a logistic regression layer on top of the network is used for supervised traffic flow prediction.
- The SAEs plus the predictor comprise the whole deep architecture model for traffic flow prediction.

Methodology

Stacked Autoencoders

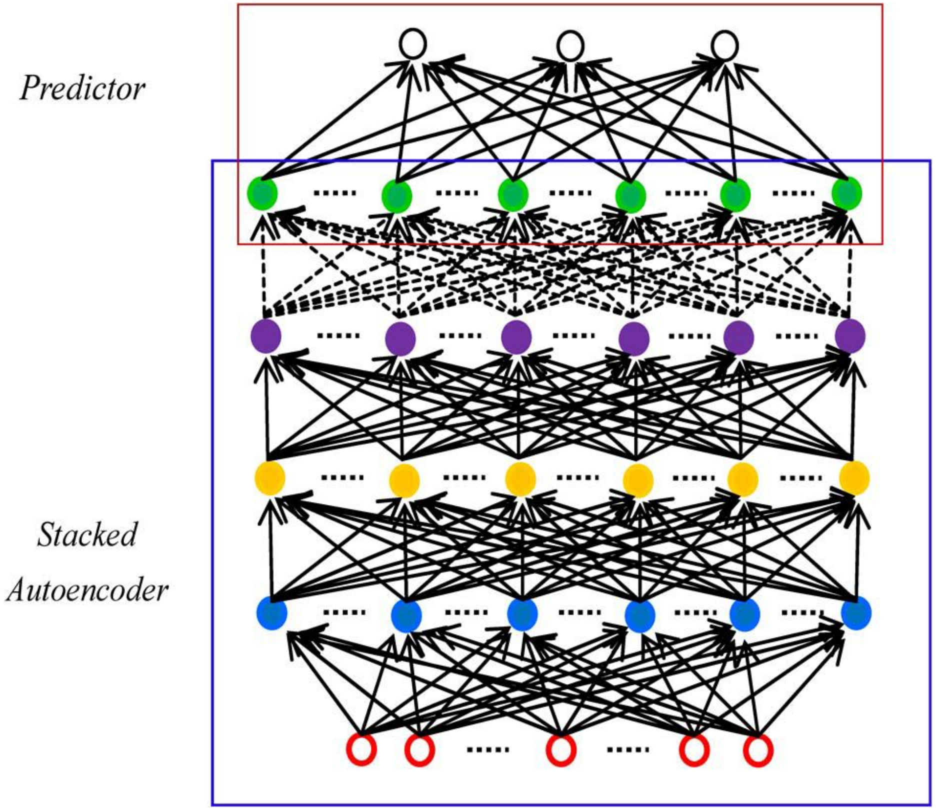
- SAEs with *l* layers
- the first layer is trained as an autoencoder, with the training set as inputs.
- After obtaining the first hidden layer, the output of the k^{th} hidden layer is used as the input of the (k + 1)th hidden layer.



Methodology

Deep architecture model for traffic flow prediction.

- A SAE model is used to extract traffic flow features, and a logistic regression layer is applied for prediction.



Methodology

Training Procedure

- The key point to using the greedy layer wise unsupervised learning algorithm is to pretrain the deep network layer by layer in a bottom–up way
- After the pretraining phase, fine-tuning using BP can be applied to tune the model’s parameters in a top–down direction to obtain better results at the same time.

Methodology

Training Procedure

1. Train the first layer as an autoencoder by minimizing the objective function with the training sets as the input.
2. Train the second layer as an autoencoder taking the first layer's output as the input.
3. Iterate as in 2) for the desired number of layers.
4. Use the output of the last layer as the input for the prediction layer, and initialize its parameters randomly or by supervised training.
5. Fine-tune the parameters of all layers with the BP method in a supervised way.

Methodology

Training Algorithm

Algorithm 1. Training SAEs

Given training samples X and the desired number of hidden layers l,

Step 1) Pretrain the SAE

- Set the weight of sparsity γ, sparsity parameter ρ, initialize weight matrices and bias vectors randomly.
- Greedy layerwise training hidden layers.
- Use the output of the kth hidden layer as the input of the $(k+1)$th hidden layer. For the first hidden layer, the input is the training set.
- Find encoding parameters $\{W_1^{k+1}, b_1^{k+1}\}_{k=0}^{l-1}$ for the $(k+1)$th hidden layer by minimizing the objective function.

Methodology

Training Algorithm

Step 2) Fine-tuning the whole network

— Initialize $\{W_1^{l+1}, b_1^{l+1}\}$ randomly or by supervised training.

— Use the BP method with the gradient-based optimization technique to change the whole network's parameters in a top–down fashion.

Experiments

Dataset Description

- Caltrans Performance Measurement System (PeMS) database as a numerical example.
- The traffic data are collected every 30s (aggregated to 5min) from over 15000 individual detectors, which are deployed statewide in freeway systems across California.
- The data of the first two months were selected as the training set.
- The remaining one month's data were selected as the testing set.

Experiments

Index of Performance

- Mean Absolute Error (MAE)
- Mean Relative Error (MRE)
- Root Mean Square Error (RMSE)

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |f_i - \hat{f}_i|$$

$$\mathrm{MRE} = \frac{1}{n}\sum_{i=1}^{n} \frac{|f_i - \hat{f}_i|}{f_i}$$

$$\mathrm{RMSE} = \left[\frac{1}{n}\sum_{i=1}^{n} \left(|f_i - \hat{f}_i|\right)^2\right]^{\frac{1}{2}}$$

where f_i is the observed traffic flow, and f_i^ is the predicted traffic flow

Experiments

Determination of the Structure of a SAE Model

- Input layer used the data collected from all freeways as the input
- Considering the temporal relationship of traffic flow, to predict the traffic flow at time interval t, use the traffic flow data at previous time intervals i.e. $X^{t-1}, X^{t-2}, ..X^{t-r}$.
- This accounts for the spatial and temporal correlations of traffic flow inherently.
- The dimension of the input space is mr, whereas the dimension of the output is m, where m is the number of freeways.

Experiments

Determination of the Structure of a SAE Model

TABLE I
STRUCTURE OF SAEs FOR TRAFFIC FLOW PREDICTION

Task	r	Hidden Layers	Hidden Units (bottom-top)
15-min traffic flow prediction	3	3	[400, 400, 400]
30-min traffic flow prediction	3	3	[200, 200, 200]
45-min traffic flow prediction	4	2	[500, 500]
60-min traffic flow prediction	3	4	[300, 300, 300, 300]

Experiments

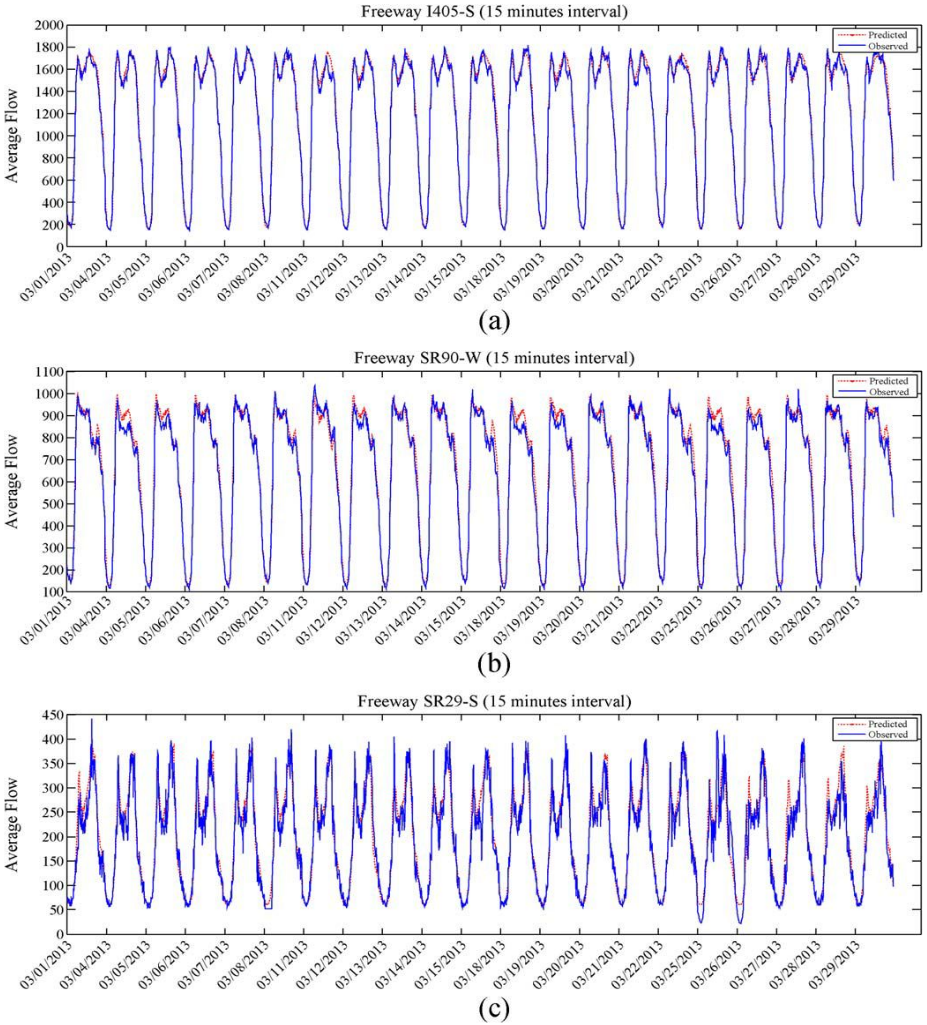
Results

- Predicted traffic flow has similar traffic patterns with the observed traffic flow.
- Matches well in heavy and medium traffic flow conditions.
- However, the proposed model does not perform well in low traffic flow conditions, which is the same as existing traffic flow prediction methods.
- The reason for this phenomenon is that small differences between the observed flow and the predicted flow can cause a bigger relative error when the traffic flow rate is small.

Experiments

Results

Fig. 5. Traffic flow prediction of roads with different traffic volume. (a) Road with heavy traffic flow. (b) Road with medium traffic flow. (c) Road with low traffic flow.



Experiments

Results

BP NN - Back propagation Neural Network
RW - Random Walk
SVM - Support Vector Machine
RBF - Radial Basis Function Neural Network

TABLE II
PERFORMANCE COMPARISON OF THE MAE, THE MRE, AND THE RMSE FOR SAEs, THE BP NN, THE RW, THE SVM, AND THE RBF NN

Task	Stacked Autoencoders			BP Neural Network			RW			SVM			RBF		
	MAE	MRE (%)	RMSE	MAE	MRE (%)	RMSE	MAE	MRE (%)	RMSE	MAE	MRE (%)	RMSE	MAE	MRE (%)	RMSE
15-min traffic flow prediction	34.1	6.75	50.0	60.8	10.9	94.1	38.3	7.8	56.7	38.7	8.0	62.3	38.3	7.4	55.9
30-min traffic flow prediction	64.1	6.48	95.2	114.3	11.3	173.3	125.0	12.1	182.6	115.5	10.3	188.3	120.0	13.0	177.3
45-min traffic flow prediction	92.0	6.17	138.1	151.2	10.2	237.0	260.0	17.1	374.7	220.0	15.8	350.4	228.6	16.4	335.6
60-min traffic flow prediction	122.8	6.21	183.9	202.8	9.8	321.5	445.0	22.3	633.4	372.9	22.1	607.5	443.4	26.4	652.6

Conclusions

- The proposed method can successfully discover the latent traffic flow feature representation
  - nonlinear spatial and temporal correlations
- Applied the greedy layer wise unsupervised learning algorithm to pre-train the deep network
- Fine-tuning process to update the model's parameters to improve the prediction performance
- Evaluated the performance of the proposed method on a PeMS data set and compared it with the BP NN, the RW, the SVM, and the RBF NN model

Questions?