Traffic speed prediction in the Lyon area using DCRNN

Fabio Mensi[1,2,3], Angelo Furno[2][0000-0001-9658-9179], and Rémy Cazabet[3][0000-0002-9429-3865]

[1] ENS de Lyon, CNRS, IXXI, Lyon, France
[2] Univ. Lyon, Univ. Gustave Eiffel, LICIT UMR_T9401, Lyon, France
[3] Univ de Lyon, CNRS, Université Lyon 1, LIRIS, UMR5205 Villeurbanne, France

Keywords: Traffic prediction · Machine learning on graphs · Graph neural networks

1 Introduction

In this paper, we deal with the problem of traffic speed prediction, i.e., forecasting, based on relevant historical data (e.g., previous speed values, topological features, etc.) up to a certain time t, what will be the average speed on a given set of road segments at time $t+T$, where T is the **prediction horizon**.

More specifically, we focus on **short term prediction** (i.e., a maximum T of one hour) for a subset of road segments in the road network of Lyon, France, by exploiting the past history of observed traffic speeds from a real dataset.

1.1 Lyon road network

Our dataset consists in floating car data, reconstructed from GPS trajectories collected from 20,000 cars on an average working day. Data is heterogeneously distributed and most of the signal relates to a minor subset of road segments during daytime. To address this issue we have developed a filtering pipeline to select a subset of road segments that meet several criteria, including high data availability, non-stationarity and noise.

From the initial road network, consisting in 317,693 nodes, we extract a subgraph of 180 road segments, as highlighted in Fig. 1. This subset is interested by several **congestion phenomena**, making the traffic speed prediction task particularly challenging and the adoption of **deep learning techniques**, such as **Graph Neural Networks** (GNN), well justified.

2 Graph Neural Networks, the idea

In many domains, data is naturally represented in the form of graphs. In chemistry, molecules are made of atoms linked via chemical bonds, in e-commerce, costumers and products are linked via their consumer relationship. Novel deep learning methods have been introduced to address those problems where data

Fig. 1. Lyon subgraph, blue markers represent nodes

lies on graph by exploiting their underlying network structure [3]. These methods are generally referred as GNNs and they are typically implemented to solve problems such as node classification, graph classification or link prediction.

We focus our attention on a specific kind of GNN, namely Diffusion Convolutional Recurrent Neural Network (DCRNN), that has already shown promising results in traffic speed forecasting settings.

3 DCRNN

The DCRNN is a type of GNN specifically aimed at tackling spatiotemporal forecasting tasks, first introduced in an article by Y. Li et al. [2]. In the original paper it was described and evaluated on a traffic speed prediction problem, reporting state-of-the-art performances.

Speed data can be represented as graph signals (time series) X_t laying on the road network, modelled as a directed graph G, where road segments are represented as nodes and intersections as edges.

3.1 Diffusion Convolution operation

Spatial dependency is modelled following an analogy with a diffusion process.

Given a graph signal $\boldsymbol{X} \in R^{N \times P}$, with N being the number of nodes in the graph, and the diffusion transition matrices $D_O^{-1}W$, $D_I^{-1}W^T$, with D_O representing the out-degree diagonal matrix, D_I the in-degree diagonal matrix, and W the adjacency matrix of the directed graph G, the convolution operation is defined as:

$$\boldsymbol{X}_{:,p} \star f_\theta = \sum_{k=0}^{K-1} (\theta_{k,1}(D_O^{-1}W)^k + \theta_{k,2}(D_I^{-1}W^T)^k)\boldsymbol{X}_{:,p} \text{ for } p \in \{1, ..., P\} \quad (1)$$

The filtered signal of a node i $(1 \leq i \leq N)$ is the result of combinations of the signals of the neighbors in the graph up to a distance of K steps from the node, weighted by the parameter $\theta \in R^{K \times 2}$.

3.2 DCGRU cell

The diffusion convolution operation accounts for the spatial dynamics component of the problem and its implementation is directly connected with the temporal dynamics part.

The temporal dependency is modelled through a recurrent neural network variant, the gated recurrent unit (GRU)[1]. To combine spatial and temporal modelling each matrix multiplication operation is replaced by the the diffusion convolution operation, described in the previous equation. The resulting modified GRU cell, called DCGRU, can be defined by the following equations:

$$\begin{aligned} r^{(t)} &= \sigma(\Theta_{r\star G}[X^{(t)}, H^{(t-1)}] + b_r) \\ u^{(t)} &= \sigma(\Theta_{u\star G}[X^{(t)}, H^{(t-1)}] + b_u) \\ C^{(t)} &= \tanh(\Theta_{C\star G}[X^{(t)}, (r^{(t)} \odot H^{(t-1)})] + b_c) \\ H^{(t)} &= u^{(t)} \odot H^{(t-1)} + (1 - u^{(t)}) \odot C^{(t)} \end{aligned} \tag{2}$$

where $X^{(t)}$, $H^{(t)}$ are the input and output (or activation) at time t, $r^{(t)}$, $u^{(t)}$ are the reset gate and update gate, and $C^{(t)}$ is the candidate output, which contributes to the new output based on the value of the update gate $u^{(t)}$.

3.3 Results in the Lyon area

We have applied the DCRNN method to our case study, using data from October-November 2017, and compared it with simple baselines, such as historical average and naive forecasting, and other more traditional methods, such as ARIMA (AutoRegressive Integrated Moving Average) and VAR (Vector AutoRegression).

In Table 1, the performances of these techniques are reported according to commonly used error metrics. Performance improvement can be observed when using DCRNN, especially with shorter horizons and the Mean Average Error (MAE) metric (the metric the neural network optimizes during training). An additional study has been performed on two subareas, where we compare DCRNN to other classical deep learning methods obtaining similar results, with DCRNN exhibiting better performance overall.

4 Reusable DCGRU

In order to allow future model developments, the DCGRU cell should be readily usable, independently from its original context. Starting from the original

Table 1. Performance comparison for different forecasting approaches, applied on the Lyon area subgraph

T	Metric	Hist Avg.	Naive	ARIMA	VAR	DCRNN
15min	MAE	6.27	4.01	4.36	4.13	**3.23**
	MSE	152.83	79.83	85.84	69.33	**59.80**
	MAPE	9.70%	8.37%	9.63%	9.23%	**7.39%**
30min	MAE	6.27	5.43	5.78	5.05	**4.07**
	MSE	152.83	151.48	147.53	131.04	**101.82**
	MAPE	**9.70%**	12.11%	13.42%	12.07%	10.07%
60min	MAE	6.27	6.81	7.01	5.76	**4.74**
	MSE	**152.83**	295.93	256.92	232.44	167.13
	MAPE	**9.70%**	19.00%	19.35%	16.32%	14.54%

implementation[2], we have built the DCGRU cell using the latest version of Tensorflow, a popular machine learning library, in such a way that is compatible with Keras, a deep learning API, and does not rely on deprecated libraries.

Our goal is to provide code that allows for quick model upgrading and can be integrated with minimal effort[4]. We have tested the newly implemented DCGRU layer on the prediction of a graph-based synthetic signal and compared it with other commonly used deep learning architectures. The DCGRU layer obtained the best performance, showing potential for employment in similar tasks.

5 Future perspectives

We aim at upgrading the current DCGRU cell to make it able to exploit information other than traffic speed or graph-based signals. For example non-graph related temporal data (in a traffic prediction context this could mean weather, holidays, weekday, etc...), static graph-related features (e.g. road structure) or the adoption of attention mechanisms[4] in the diffusion convolution operation.

Other efforts could be dedicated to redefine the traffic speed prediction problem in order to meaningfully consider those nodes that have been excluded in the filtering step (mostly urban road segments).

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv preprint arXiv:1707.01926 (2017)
3. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE Transactions on Neural Networks **20**(1), 61–80 (2008)
4. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)

[4] Code available at github.com/mensif/DCGRU_Tensorflow2