**Recent Advancements in Deep Learning in GIS – Spatiotemporal data mining and forecasting**

**Prepared by**

**Supasin Wuthikulphakdi**

**ID# 608785068**

**Tamkang University**

**September 3, 2021**

**A proposal submitted in partial fulfillment of the requirements for the Master Degree of Science in Computer Science**

**Proposed Supervisor: Isaac Tsai Yihjia, Faculty of Engineering**

**Proposed Reader(s) Faculty of Engineering – Department of Computer Science**

**Date of Submission**

**September 3, 2021**

**TITLE**

Recent Advancements in Deep Learning in GIS – Spatiotemporal data mining and forecasting

**INTRODUCTION & BACKGROUND**

Nowadays, data mining is a powerful analytic tool. As some forms of static datasets (i.e. images in the form of matrices) and sequential ones (i.e. time series, Natural Language Processing (NLP), etc.) are being mined to train Machine Learning (ML) analytic and predictive models.

In Geoinformatics (GIS) [16], spatiotemporal data is a dataset relate to both space and time. It is collected in the space domain (a.k.a. a (heat)map over a location in a form of raster and vector data matrices), and the time domain (a.k.a. the time series - a heatmap average for each timestep in the form of timestep-D vectors, also includes NLP datasets). Overall, when concatenated, a spatiotemporal dataset is usually a 3D+ tensor.

Given the popularity of GPS-supporting devices, including smartphones, the need of S.T.-data mining rises with the emerging use of intelligent transport system (ITS, i.e. automatic guidance, self-driving cars, traffic prediction, etc.), and disaster prediction (i.e. weather, flood, earthquake, storms, clouds, smog, global warming etc.).

To manually collect, process, and forecast the ST data is a labourious task. Several models based on both machine learning have been deployed, but ML models need a human to extract the feature representations. Deep learning models, capable of self-feature extraction, usually outperform regular ML ones. Fundamentally, a ST-Deep Learning model works as follows:

* A CNN learns the spatial input (i.e. heatmap, graph data (GCN), & data coordinates)
* While and LSTM/RNN learns the temporal domain of the input data.
* With both DNN working together, sometimes with an intermediate submodel (i.e. a dense submodel in [14]), we can correlate and forecast a spatiotemporal data (such as traffic, train ridership, etc.) not only the time series (time domain), but also the heatmap (space domain).

However, DL models have drawbacks, a CNN/GCN/FCN can only extract and learn only the space domain, while an RNN/LSTM/seq2seq can only do these on the temporal domain of the S.T. data. Proven by several papers, RNN, when processing a long sequence, has a problem of vanishing/exploding gradient. Thus, LSTM is preferred in later works

As the result, some work referred in the LR section overcame the problems by adding such two neural networks into the same framework, allowing them to extract and learn the ST-data of both domains simultaneously. More technical details can be read in “Outline\_Draft3.docx”

My thesis is focused on paper survey of deep learning algorithms and models in ST-data mining & forecasting. With the AI, there are some uses in Spatiotemporal data mining:

* CNN-GCN-FCNs are used to extract features in the spatial domain (i.e. heatmap).
* LSTM-RNN-seq2seq are used to extract features in the temporal domain (i.e. sequence type data).
* In survey engineering, some researchers use CNN observe changes in land use, crop growth, and construction progresses.
* In ITS, ST-Data mining is extremely useful in traffic prediction, guidance systems, route planning, and self-driving cars.
* In Meteorology, the STDM is used for weather and disaster forecast.
* To make GPS based pathfinding more accurate, Spatiotemporal Data Mining is often utilized in several papers to train the pathfinding ML/DL-architectures.
* In transportation, deep learning methods learn highly intricate ST-correlations among the traffic data – useful in some tasks such as traffic flow prediction, traffic incident detection, and traffic congestion prediction – such as in [13] and [14].
* In On-demand service, such as Uber [10], to perform pathfinding
* All the papers to be referred below, may be added & updated in the future, involve GIS, Computer Vision (CV), time series prediction, and spatiotemporal data mining.

**LITERATURE REVIEW & RELATED WORKS**

According to Doshi et al, 2018 [1], a CNN is employed to identify disaster-impacted areas by comparing the change in man-made features extracted from satellite imagery. Using a pre-trained semantic segmentation model from [2] they extracted man-made features, the pre- and post-event images on the “before, during and after” imagery of the event-affected area.

According to Amit et al. [3], CNN is a sequence of layers, the convolution layer (who detects features from a data image), the pooling layer (downsamples the input), and the FC layer (who classifies the features detected earlier). with ReLU as the main activation function of the network. Further explained in the section 2 of [3].

Iglovikov et al. 2017 [4], used an FC-CNN named U-NET, along with an embedded multispectral sensor, which detects frequency reflection by the objects, to detect geo-features in satellite images and yielded satisfying results.

According to Bochkovskiy et al. [5], with the help of YOLO V.4, and TensorFlow Keras, CNN’s performance in image recognition is improved. So it is possible for us to deploy such a CNN model in this thesis. We hope that our deep learning models, written in Python, will work as the goals above.

In Terms of video and sequence type photos (such as slideshow), however, the use of LSTM-RNN is needed. According to Fang et al. [6], LSTM is excellent at predicting flood because it could process time series data.

According to Li et al, 2018. [7], spatiotemporal forecasting is a crucial task for a learning system that operates in a dynamic environment. It can be useful in pathfinding, autonomous vehicles, logistics, city planning etc. They used a Diffusion Convolutional Recurrent Neural Network (DCRNN) model to forecast the road traffic within a specific space and timeframe (The dataset was METR-LA, 2014). Diffusion convolution extracts the traffic features, and the RNN processes the traffic volumes in sequence.

Yu et al, 2018. [8] proposed a Spatiotemporal Graph Convolutional Networks (STGCN), to tackle the time series prediction problem in traffic domain. They formulated the problem on graphs and build the model with complete convolutional structures, enabling much faster training speed with fewer parameters. Compared with existing models, STGCN more effectively captured comprehensive spatiotemporal correlations through modeling multi-scale traffic networks and consistently outperforms state-of-the-art baselines on various real-world traffic datasets.

Correa et al, 2017 [9] performed a spatiotemporal data mining of Taxi vs Uber ridership in NYC, 2014+15. According to the heatmap inside the paper, the ridership for both taxi systems depended on several factors – such as personal income, education, jobs, car ownership etc. With 3 spatial models for ridership prediction – linear, spatial error and spatial lag models, the last one outperformed not only the first 2 algorithms, but also yielded a considerable accuracy and performance.

Amato et al. [10] designed a deep learning-based architecture called “Empirical Orthogonal Functions principal component analysis” EOF-PCA in which the EOF framework decomposes the spatiotemporal input data, in terms of a sum of products of temporally referenced basis functions and of stochastic spatial coefficients which can be spatially modelled and mapped on a regular grid. Then, the input layer spatial covariates are processed by a “Fully Connected Neural Network” (FCNN) to obtain predictive coefficient to be recomposed altogether with the decomposed data stream, obtaining a spatiotemporal signal reconstruction.

Tang et al. [11] designed an LSTM based framework to learn and forecast the rail traffic. The short-term forecast of rail transit is an issue in intelligent transportation system (ITS). Accurate forecast can forewarn travel outburst, helping the passengers with their travel plans. Even though the LSTM is notably effective in temporal data, it cannot correlate the time domain with the space domain. That is why we propose ST-LSTM. Compared with other conventional models, ST-LSTM network can achieve a better performance in experiments.

Lu et al. [12], designed a spatial-temporal deep learning network, termed ST-TrafficNet, for traffic flow forecasting, whose architecture works as follows. 1. The Spatial Aware Multi-Diffusion Convolution Bloc (ADC-Block – who introduces Graph Attention Mechanism (GAM) into the MDC) uncovers unseen spatial dependencies from traffic graph signals automatically 2. From the data stream, the multi-diffusion convolution (MDC) block harvests ST-features of the spatial domain. 3. The ST-TrafficNet, an LSTM based framework, harvests the features of the temporal domain. 4. The output from both ANN are summed up to achieve convolutional results. And 5. The ST-TrafficNet is evaluated on two benchmark datasets and compare it with various baseline methods for traffic forecasting.

Pan et al. [13] designed a deep learning framework for traffic flow prediction called “ST-Metanet” According to him, Traffic prediction is to enhance traffic safety and make the transportation system intelligent. However, it has to face to challenges: 1) complex spatio-temporal correlations of urban traffic and 2) diversity of such spatio-temporal correlations. To tackle these challenges, they proposed a deep-meta-learning based traffic model, entitled ST-MetaNet, to collectively predict urban traffic in all location at once. ST-MetaNet employs a seq2seq architecture, consisting of an encoder to learn historical traffic information and a decoder to make predictions step by step. More specifically, the encoder and decoder have the same network structure, which contains a recurrent neural network (RNN) to encode the urban traffic, a meta graph attention network (Meta-GAT) to capture diverse spatial correlations, and a meta recurrent neural network (Meta-RNN) to consider diverse temporal correlations. Extensive experiments were conducted based on two real-world datasets to illustrate the effectiveness of ST-MetaNet against several state-of-the-art methods.

De Medrano et al. [14] designed A Spatio-Temporal Spot-Forecasting Framework for Urban Traffic Prediction, named CRANN (Convo-Recurrent Attentional Neural Network). It is highly adaptable in several ST conditions, easy to understand and interpret, and better & more stable than state-of-the-art alternatives.

To cope the nonlinearity of the traffic flow data during the holidays, Luo et al, 2019 [15], designed a discrete Fourier transform (DFT) and support vector regression (SVR) based machine learning model to predict the road traffic flow during the holidays in Jiangsu Province, China, on Tomb-sweeping Day and National Day from 2011 to 2015. With proper training, the model outperformed other ML models – like ARIMA, SVR and EMD-SVR. The model is described in the paper itself [15].

Shih et al. [17] designed an LSTM capable of processing multiple time series at the same time called “Temporal Pattern Attention LTSM” (TPA-LSTM). The architecture is designed to process complex and non-linear interdependencies between time steps of multivariate time series data. To obtain accurate prediction, an RNN with attention mechanism is designed and deployed to learn long-term dependency in time series data. The typical attention mechanism reviews the information at each previous time step and selects relevant information to help generate the outputs; however, it fails to capture temporal patterns across multiple time steps. The model uses a set of filters to extract time-invariant temporal patterns, similar to transforming time series data into its “frequency domain”. The attention mechanism to select relevant time series, and use its frequency domain information for multivariate forecasting. Surprisingly, regardless of the cases, the model achieved a comparable performance with other state-of-the-art models and architectures.

**Potential Methods**

From each referred/surveyed paper, on GoogleTM-Colab environment, we will run the Python codes written for each model and dataset. Then, we will run the models in the reference with their own datasets, and compare them in the form of MSE & RMSE metrics – the less of them for each model, the better it performs in the term of learning and forecasting. Finally, we will conclude which model does the best learning/forecast. As this is a survey thesis, we will not design a custom model.

The Deep Learning models and ANNs are written in Python with Pytorch, Mxnet & Tensorflow-Keras as our preferred modules. Be advised, not every model is runnable on the stock COLAB environment, and thus must be modified to meet their requirements.

**Experiment Results:**

The full table can be seen in this week’s .xlsx file of the author. Usually for each model, the learning rate is 0.01, and the optimiser is Adam’s, and they are trained for 200 epochs.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Architecture** | **Epochs** | **learning rate** |  | **Key Metrics** | | **Other Metrics** | |
|  |  |  | **MSE (either)** | **RMSE(train)** | **RMSE(test)** | **Type** | **Value** |
| Taxi-Simple-LSTM-pytorch | 200 | 0.01 | 7.6E+07 | 8691.08 | N/A |  |  |
| Uber-Simple-LSTM-pytorch | 200 | 0.01 | 0.03257 | 0.18049 | N/A |  |  |
| Taxi-Simple-LSTM-Keras | 200 | 0.01 | 17995921 | 4242.16 | 10723.6 |  |  |
| Uber-Simple-LSTM-Keras | 200 | 0.01 | 33913502 | 5823.53 | 10244.9 |  |  |
| CRANN-Temporal | 200 | 0.01 | 6954.62 | - | 83.3944 | Rel.Err.% | 9.2181 |
| CRANN-Spatial | 200 | 0.01 | 58546.7 | - | 241.964 | Rel.Err.% | 21.7391 |
| CRANN-Dense | 200 | 0.01 | 67154.6 | - | 259.142 | Rel.Err.% | 24.7697 |
| seq2seq(flow) | 200 | 0.01 | 1814.76 | 40.335922 | 42.6 | MAE | 21.3 |
| GAT-seq2seq(flow) | 200 | 0.01 | 1267.36 | 33.136707 | 35.6 | MAE | 18.3 |
| ST-Metanet(flow) | 200 | 0.01 | 1156 | 28.51646 | 34 | MAE | 16.9 |
| seq2seq(speed) | 200 | 0.01 | 52.8529 | 6.6689725 | 7.27 | MAE | 3.55 |
| GAT-seq2seq(speed) | 200 | 0.01 | 44.3556 | 6.0764656 | 6.66 | MAE | 3.28 |
| ST-Metanet(speed) | 200 | 0.01 | 39.0625 | 5.7992749 | 6.25 | MAE | 3.05 |
| TPA-LSTM | 40 | 1.00E-05 | RAE=0.3118 | RSE=0.4765 | CORR=0.9850 | Precision | 0.58851 |
|  |  |  |  |  |  | Recall | 0.68889 |
|  |  |  |  |  |  | F1 | 0.76333 |
| Lotto-seq2seq | 200 | 1.00E-04 | - | - | - | Sparse Top K | 0.22571 |

The yellow cells indicate the models that need significant architectural changes to calculate the MSE due to unequal input-output tensor dimensions. The blue numbers indicate possible Metric values as if whose models were actually run (can be turned black later if a suitable environment is found). And the blue model names indicated planned, but not yet run, models. Despite environmental restrictions preventing such model from running on COLAB, it is obvious that the model performed well, at least for its architecture. To see the detailed issues and possible solutions, see the issue subsection below.

It is also apparent that complex state-of-the-art models (such as CRANN), when learning correct datasets, greatly outperformed simple models (like keras-LSTM). As we can see on the table above, the ST-Metanet model, to date, is the best model for spatiotemporal traffic prediction.

**Issues:**

The ST-Metanet model by Pan et al. [7] has a set of stringent environmental and modular requirements

* Which must be 100% met, otherwise the model will not run.
* Mxnet version worked: 1.4.0
  + But the notebook crashed due to insufficient RAM (required at least 25GB)
* Pymal has a ‘six’ dependency version conflict with pandas
* A Cloud VM seems to be the most promising solution

The TRA-LSTM model by Shih et al. [17], despite suitable environment set on COLAB, takes forever to train.

* + It is better to run it on a physical machine, otherwise run it under COLAB PRO+ Environment.
    - The author needs to wait until the first week of semester 110/1, to relocate the lab and acquire some stronger hardware and computing resources.

**To do next**

* Study Attention Mechanisms
* Find suitable environments for
  + ST-Metanet model of Mxnet-cu90 V.1.4.0
  + TPA-LSTM
* Run the Lotto-LSTM+Att model (of MSE)
* Upgrade the author’s COLAB to pro+ version (costs him NT$1500/month)
* Run a spatial model for Taxiuber
  + Choose & implement a suitable model for Taxiuber spatial module (Likely a CNN based arch.)

**Project Phases:**

* August/September
  + Run Models & Environmental Solutions
* October
  + Final Validation Runs, statistical collection & analysis of model performance
* November
  + Result & Analysis + Writing a paper.
* December
  + Finish the paper.
* January
  + Defence oral exam

**Progresses:**

Current Phase: Run Models & Environmental Solutions

**Works done:**

* Read 4 reference papers
* Run the Lotto-LSTM+Att model (of the original metric)
* Run the taxi-uber LSTM time series forecasting model written in Keras
  + Model & coding are to be improved to extrapolate the future trends.
  + To implement a CNN/FCN-model for the spatial module is our another task
* Write python notebooks for the ST-Metanet and the TPA-LSTM networks on colab
* CRANN – model deployment has been completed with satisfying results

**References:**

[1] Doshi, J., Basu, S. and Pang, G., 2018. From satellite imagery to disaster insights. *arXiv preprint arXiv:1812.07033*.

[2] Doshi, Jigar. 2018. Residual Inception Skip Network for Binary Segmentation. Pages 216–219 of:Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.  
  
[3] Amit, S.N.K.B. and Aoki, Y., 2017, September. Disaster detection from aerial imagery with convolutional neural network. In 2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC) (pp. 239-245). IEEE.

[4] V. Iglovikov, S. Mushinskiy, and V. Osin, “Satellite Imagery Feature Detection using Deep Convolutional Neural Network: A Kaggle Competition,” vol. June, 2017.

[5] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: Optimal speed and accuracy of object detection,” ArXiv, vol. abs/2004.10934, 2020.

[6] Fang, Z., Wang, Y., Peng, L. and Hong, H., 2020. Predicting flood susceptibility using LSTM neural networks. *Journal of Hydrology*, [online] p.125734. Available at: <https://doi.org/10.1016/j.jhydrol.2020.125734> [Accessed 18 April 2021].

[7] Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

[8] Yu, B., Yin, H., & Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*.

[9] Correa, D., Xie, K., & Ozbay, K. (2017). Exploring the taxi and Uber demand in New York City: An empirical analysis and spatial modeling. In *96th Annual Meeting of the Transportation Research Board, Washington, DC*.

[10] Amato, F., Guignard, F., Robert, S., & Kanevski, M. (2020). A novel framework for spatio‑temporal prediction of environmental data using deep learning. Scientific Reports, 2020(10), 22243. doi:

[11] Tang, Q., Yang, M., & Yang, Y. (2019). ST-LSTM: A Deep Learning Approach Combined Spatio-Temporal Features for Short-Term Forecast in Rail Transit. Journal of Advanced Transportation, 2019, Article ID 8392592. doi:<https://doi.org/10.1155/2019/8392592>

[12] Lu, H., Huang, D., Song, Y., Jiang, D., Zhou, T., & Qin, J. (2020). ST-TrafficNet: A Spatial-Temporal Deep Learning Network for Traffic Forecasting. Electronics, 2020(9), 1474.

[13] Pan, Z., Liang, Y., Wang, W., Yu, Y., Zheng, Y., & Zhang, J. (2019, July 25). Urban Traffic Prediction from Spatio-Temporal Data Using Deep Meta Learning. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. KDD ’19: The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. https://doi.org/10.1145/3292500.3330884

[14] de Medrano, R., & Aznarte, J. L. (2020). A spatio-temporal attention-based spot-forecasting framework for urban traffic prediction. *Applied Soft Computing*, *96*, 106615.

[15] Luo, X., Li, D., & Zhang, S. (2019). Traffic flow prediction during the holidays based on DFT and SVR. *Journal of Sensors*, *2019*.

[16] Jiawei Han, Micheline Kamber, Jian Pei, 13 - Data Mining Trends and Research Frontiers, Morgan Kaufmann, 2012, Pages 585-631.

**[17]** Shih, S. Y., Sun, F. K., & Lee, H. Y. (2019). Temporal pattern attention for multivariate time series forecasting. *Machine Learning*, *108*(8), 1421-1441.