Multivariate Time-Series Imagification with Time Embedding in Constrained Environments (Student Abstract)

Seung Woo Kang, Ohyun Jo*

Chungbuk National University, Department of Computer Science, 28644, Cheongju, Republic of Korea
swkang@chungbuk.ac.kr, ohyunjo@chungbuk.ac.kr

Abstract

We present an imagification approach for multivariate time-series data tailored to constrained NN-based forecasting model training environments. Our imagification process consists of two key steps: Re-stacking and time embedding. In the Re-stacking stage, time-series data are arranged based on high correlation, forming the first image channel using a sliding window technique. The time embedding stage adds two additional image channels by incorporating real-time information. We evaluate our method by comparing it with three benchmark imagification techniques using a simple CNN-based model. Additionally, we conduct a comparison with LSTM, a conventional time-series forecasting model. Experimental results demonstrate that our proposed approach achieves three times faster model training termination while maintaining forecasting accuracy.

Introduction

These days, the proliferation of IoT devices has resulted in a wealth of multivariate time-series data. Analyzing and forecasting trends within this data has garnered significant attention in recent years, leading to the active exploration of various NN (Neural Network)-based models, including DNN (Deep Neural Networks) and RNN (Recurrent Neural Networks), for time-series forecasting. While these NNs can perform effectively once trained, their direct training on small IoT devices is often impractical. However, CNN (Convolutional Neural Networks) offer a compelling solution. CNNs excel at model training by identifying significant features within images while requiring fewer parameters compared to other NN-based models.

There have been some challenges for time-series data to be applied to CNN by converting them into images. Several methods, such as Recurrence Plots (RP) and Gramian Angular Field (GAF), have been devised to convert time-series data into images, often by concatenating them vertically in a multivariate context (Rajabi and Estebsari 2019; Wang, Oates et al. 2015; Yang, Chen, and Yang 2019). However, these traditional imagification techniques generate images with a quadratic space complexity of $O(n^2)$, making them unsuitable for resource-constrained devices.

*Corresponding author: Ohyun Jo

In this paper, we introduce a novel imagification method focused on enhancing the training efficiency of multivariate time-series forecasting models by achieving linear space complexity, $O(n)$, in image production. Our approach incorporates two key components: the Re-stacking method, which enhances associations between adjacent pixels in the image, and the time embedding method, which injects temporal information as channels into the image representation. To validate our proposed method, we conduct a comprehensive performance comparison against RP, GASF (Gramian Angular Summation Field), and GADF (Gramian Angular Difference Field) within the same CNN model framework. Additionally, we include a comparison with LSTM (Long Short-Term Memory), a conventional choice for time-series forecasting, to assess the effectiveness of our approach.

Proposed Imagification Method

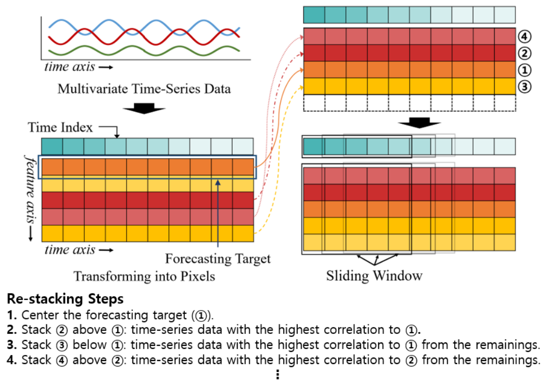
Re-stacking Time Series Features

Re-stacking time-series features enhances the effectiveness of CNNs in capturing significant features within adjacent pixels. This approach draws inspiration from the observation that adjacent pixels in conventional images are typically contiguous. Moreover, this technique allows models to associate features of related time-series. To implement features Re-stacking, as shown in Figure 1 (a), we place the forecasting target time-series in the middle and stack time-series data above and below it based on their Pearson Correlation Coefficient (PCC). After applying the feature Re-stacking technique, we utilized a sliding window approach to capture them as the initial channel in each image.

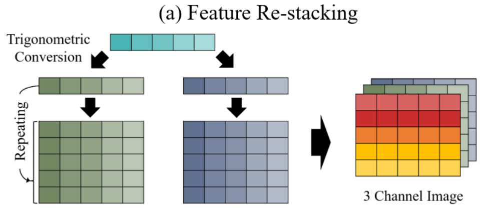
Time Embedding

Time embedding in our imagification method introduces two additional channels containing real-time information, incorporating time cycle details of time-series. Many time-series datasets include time indices with intervals, such as minutes or hours. To ensure the continuity of the time index, we employ two trigonometric functions for its conversion. The parameter C represents the cycle, for instance, representing the hourly time index repeating every 24 hours. Using C, with time index t, the trigonometric conversion of the time index is expressed as follows:

$$\sin(2\pi t/C), \cos(2\pi t/C) \quad (1)$$



(a) Feature Re-stacking



(b) Time Embedding

Figure 1: Illustration of the proposed imagification process. (a) Time-series feature Re-stacking and sliding window (b) Time embedding technique

The two converted time index sequences are repeated as many times as the length of the feature axis, as demonstrated in Figure 1 (b). Subsequently, these repeated time index sequences are added as the second and third image channels, completing a 3-channel image.

Performance Evaluation

Experimental Setting

We utilized traffic speed data collected from 105 road sections on the Seoul highway in South Korea, with measurements taken every hour. Our objective was to forecast the traffic speed for a specific road section one hour into the future, which served as our forecasting target time-series. This dataset was generated in real-life environments with support from the Ministry of Land, Infrastructure, and Transport, Korean Government.

For comparison of our proposed imagification method with other benchmark techniques, we employed a straightforward CNN-based model consisting of two convolutional layers and FC (Fully Connected) layers. Additionally, for a benchmark comparison with conventional NN-based models for time-series forecasting, we also employed an LSTM-based model integrated with FC layers.

Experimental Results

In terms of model training time, the CNN-based model with our proposed method achieved the fastest convergence to the minimum error, as shown in Figure 2. While our proposed approach reached the minimum error within 25 seconds, the other models required more than 76 seconds. In the worst case, it exceeded 130 seconds. Therefore, our proposed imagification method has the potential to significantly reduce training time.

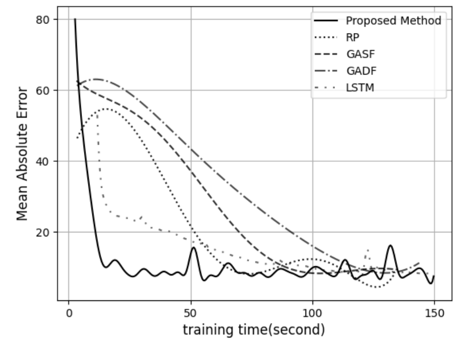


Figure 2: A plot illustrating how the MAE of each model changes over training time

Furthermore, our proposed approach demonstrates comparable performance to other models in terms of forecasting accuracy. During the verification using the test dataset, the model employing the proposed method achieved a MAE (Mean Absolute Error) of 7.96, while the RP, GASF, GADF, and LSTM approaches resulted in MAE values of 9.37, 8.85, 8.52, and 8.87, respectively. When considering forecasting accuracy across all methods, our approach exhibited slightly superior performance.

Conclusion

In this paper, we introduce an imagification method for multivariate time-series data tailored to constrained NN-based training environments. Through a series of experiments, we have demonstrated the potential of this method to reduce training time while maintaining reasonable forecasting accuracy. In our future endeavors, we aim to develop a lightweight CNN-based model capable of leveraging our proposed method.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT). (No. 2021R1A2C2095289)

References

Rajabi, R.; and Estebsari, A. 2019. Deep learning based forecasting of individual residential loads using recurrence plots. In *2019 IEEE Milan PowerTech*, 1–5. IEEE.

Wang, Z.; Oates, T.; et al. 2015. Encoding time series as images for visual inspection and classification using tiled convolutional neural networks. In *Workshops at the twenty-ninth AAAI conference on artificial intelligence*, volume 1. AAAI Menlo Park, CA, USA.

Yang, C.-L.; Chen, Z.-X.; and Yang, C.-Y. 2019. Sensor classification using convolutional neural network by encoding multivariate time series as two-dimensional colored images. *Sensors*, 20(1): 168.