Multi-Task Local-Global Graph Network for Flight Delay Prediction

Tianyi Wang
Knight Foundation School of Computing and Information Sciences
Florida International University
Miami, Florida 33199
wtian002@cs.fiu.edu

Shu-Ching Chen
Data Science and Analytics Innovation Center
University of Missouri-Kansas City
Kansas City, Missouri 64110
s.chen@umkc.edu

***Abstract*—Airline on-time performance has always been a key factor in evaluating the punctuality of the civil aviation industry and has a profound impact on airlines, airports, and passengers. As a result, there have been increasing demands for the systematic analysis of flight delays and the development of accurate and efficient tools for flight delay prediction. In this paper, a deep learning framework based on graph convolutional networks and multi-task learning is proposed for flight delay prediction. We first use graph convolutional networks to capture the local and global spatial dependencies among the airports. A multi-decoder sequence-to-sequence model is developed to extract the temporal correlation from the data. We further apply a hierarchical graph fusion approach to combine features at different levels of the network to exploit their cross-modality correlations. The model is trained using a dynamic multi-task learning strategy to predict flight arrival and departure delays at the same time to boost the model's generalization and performance. The proposed model is evaluated on a large-scale public flight record dataset against several state-of-the-art methods. The experimental results demonstrate that our model can outperform all baseline methods in predicting short to medium-term flight delays.**

***Index Terms*—flight delay prediction; deep learning; graph convolutional network; multi-task learning**

I. Introduction

According to the US Federal Aviation Administration (FAA), the total cost of delays in the US market has constantly risen in the past years and reached 33 billion dollars in 2019 [1]. As a result, the systematic analysis of flight delays and the development of accurate and efficient tools for flight delay prediction become important. Flight delay prediction problem has been extensively studied in the past decades. Conventional approaches construct mathematical or statistical models to capture the correlation between selected variables [2], [3]. Simulation tools are also developed to study the impact of certain elements on the flight network delay [4]–[6]. Recently, data-driven solutions such as machine learning and deep learning become popular due to the exponential growth of data availability and computing power [7]–[9]. Most methods on flight delay prediction focus on a single airport or airline [10], [11]. It is challenging to predict the flight delay at a network-wide level due to the complexity of capturing the correlation among a large number of airports. Existing studies on network-wide flight delay prediction struggled to fully exploit the spatial and temporal dependencies in the data [12], [13].

To address the aforementioned limitations and challenges, we propose a novel Graph Convolutional Network (GCN) based deep learning model using multi-task learning strategy (MTLG-Net) for network-wide flight delay prediction. MTLG-Net takes advantage of both local and global spatial correlations among airports in the graph network. The local correlated GCN captures the direct connectivity among nodes, whereas the global correlated GCN learns the hidden patterns modeled by the global node-wise similarity. A multi-decoder sequence-to-sequence (Seq2Seq) model is designed to capture the temporal dependency from the input data. A hierarchical graph fusion approach is adopted to combine features from different input modalities. Due to the close relationship between arrival and departure delays, we apply a dynamic multi-task learning strategy to train the model on arrival and departure delay prediction concurrently, which can substantially improve the model's performance. The main contribution of the proposed work can be summarized as follows:

- A novel deep graph network that learns both local and global spatial features. The local GCN focuses on the spatial dependency between nodes with direct connections. The global GCN captures the network-wide correlation among nodes sharing similar characteristics.
- An effective sparse matrix normalization approach to control the sparsity of the graph adjacency matrix that reduces redundant node-wise correlation.
- A multi-decoder Seq2Seq model with dynamic multi-task learning module that extracts the temporal correlation for both arrival and departure delay and improves the model learning efficiency.
- The proposed framework has been tested on a large-scale public flight record dataset. The Experiment results demonstrate the superior performance of our model compared to other baseline methods.

This paper is organized as follows. In Section II, we conduct literature review on existing works using deep learning for flight delay prediction. Section III introduces the proposed MTLG-Net and its main components. Section IV demonstrates the experimental setup and results. Last but not the least,

978-1-6654-6603-5/22/$31.00 ©2022 IEEE
DOI 10.1109/IRI54793.2022.00023

Section V summarizes the paper and suggests potential future work.

II. Related Work

Data-driven methods such as machine learning and deep learning have been extensively applied in studying the flight delay problem. Early methods in flight delay prediction mainly focus on the airport and airline levels. Yu *et al.* adopted an ensemble of deep neural networks, which include an improved Convolutional Neural Networks (CNNs) with skip connection and a combination of Densely Connected Convolutional Networks (DenseNet) [14] and Squeeze-and-Extraction Network (SENet) to predict the flight delay on a single domestic airport [15]. The authors also incorporated historical weather data to provide contextual information. In another work, Chakrabarty and Navoneel [10] utilized Gradient Boosting technique to predict flight arrival delays for American Airlines in 5 major air hubs. Our work further expands the problem domain by modeling the network-wide flight delays.

Guo *et al.* [16] applied a dual-attention network that contains graph-level and sequence-level attention mechanisms to model the spatial and temporal dependencies between airports in the network. The model is trained to classify the flight delay into a specific 30 minutes length window among the 48 intervals in a day. Bao *et al.* [17] utilized K-means clustering to divide the airports into four groups based on their average hourly delay pattern. A deep neural network based on GCN and Seq2Seq model is used to extract the spatial-temporal features. In another work, Zeng *et al.* [18] designed a weighted adjacency matrix based on the weighted sum of the spatial distance and flight frequency among the airports. In this paper, we utilize both local and global spatial correlations to comprehensively learn the spatial dependencies from data. Figure 1 demonstrates the architectural design of the MTLG-Net. We will discuss each component in details.

III. Methodology

For the flight delay prediction problem, a convolutional graph based on the flight network can be represented as $G = (X, y, E, A)$, where X represents the node features, y is the average hourly flight delay, $y \in \mathbb{R}^N$ where N is the number of airports, E is the set of edges in the graph, and A represents the graph adjacency matrix, $A \in \mathbb{R}^{N\times N}$. A forward pass in the GCN layer can be expressed as:

$$H^{[l+1]} = \sigma(H^l \widetilde{A} W^l + b^l) \tag{1}$$

where $H^{[l+1]}$ and H^l are the feature vectors at layer $l+1$ and layer l, σ represents any non-linear activation function, and W^l and b^l are the learnable weight parameter and bias term of layer l, respectively. The normalized graph Laplacian $\widetilde{A}$ is intended to prevent numerical instability and can be expressed as follows.

$$\widetilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} + I \tag{2}$$

where D is the degree matrix and I represents the identity matrix, which functions as a self-loop of the target node.

A. Local-Global Graph Convolutional Networks

Two types of GCNs are leveraged in MTLG-Net to learn the spatial correlations between airports from both local and global levels. The local correlated GCN is a directed graph based on each airport's flight route connectivity. On the other hand, the global correlated GCN uses an undirected graph to represent the node-wide similarity.

The local correlated GCN uses adjacency matrix $A_{local} \in \mathbb{R}^{N\times N}$ to represent the connectivity among airports. If there exists a flight route between airport i and airport j, then entry a_{ij} in A_{local} is set to one, otherwise zero. Therefore, the normazlied graph Laplacian for local correlated GCN A_{local} can be expressed as follows:

$$\widetilde{A}_{local} = D^{-\frac{1}{2}} A_{local} D^{-\frac{1}{2}} + I \tag{3}$$

and the graph $G_{local} = (X, y, E, \widetilde{A}_{local})$ is obtained for the local correlated GCN.

The adjacency matrix of the global correlated GCN is designed to contain the pair-wise airport similarity score, which is calculated based on the airport's annual average flight and passenger volumes. More specifically, the similarity score between two airports can be expressed as the cosine similarity between their corresponding vector representation of the two variables:

$$S_{ij} = \frac{v_i \cdot v_j}{\|v_i\| \, \|v_j\|} \tag{4}$$

where v_i and v_j are the similarity vectors for airport i and j.

The matrix produced by Equation (4) leads to very high computational complexity since the adjacency matrix is too dense. On the other hand, if the matrix becomes too sparse, some useful information may be lost. To address this issue, we develop a sparse matrix normalization approach to replace the conventional normalized graph Laplacian. The proposed sparse matrix normalization approach normalizes the matrix value so that each element's value is between 0 and 1. It also enforces matrix sparsity by removing node connections that fall below the standard deviation threshold. Therefore, the normalized global correlated GCN's adjacency matrix can be expressed as:

$$\widetilde{A}^{ij}_{global} = \begin{cases} \frac{(e^{S_{ij}-\sigma}-1)^2}{\sum \sum_{ij}^{N} (e^{S_{ij}-\sigma}-1)^2+\varepsilon}, & i \neq j \ and \ S_{ij}-\sigma > 0 \\ 0, & otherwise \end{cases} \tag{5}$$

where S_{ij} is the similarity score between airport i and airport j. If we define the flattened similarity matrix as a 1D array $\hat{S} = [S_{11}, S_{12}, S_{13}, ...S_{NN}]$, $\hat{S} \in \mathbb{R}^{N^2}$, then σ is the standard deviation of $\hat{S}$ and ε is a small constant to reduce the numerical instability.

The normalized global correlated GCN adjacency matrix with self loop can be expressed as:

$$\widetilde{A}_{global} = \hat{A}_{global} + I \tag{6}$$

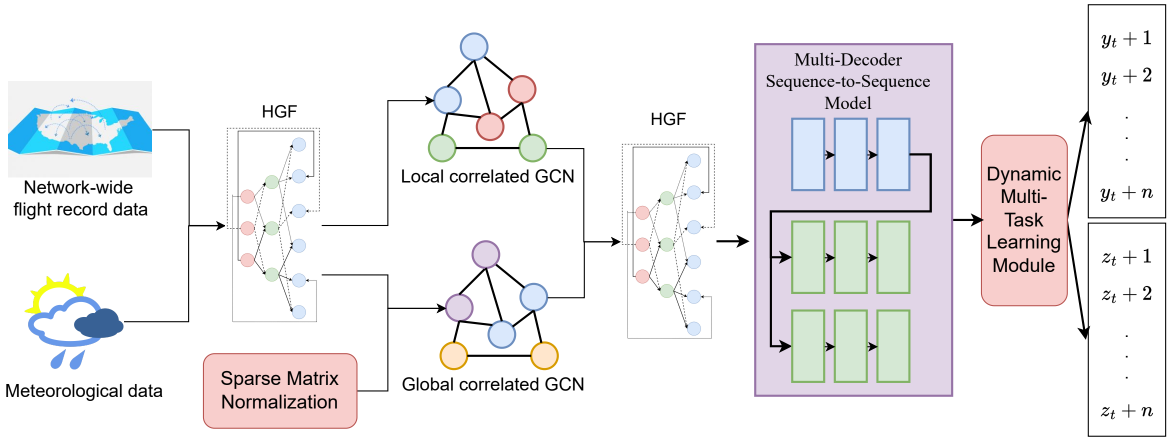

Fig. 1. An overview of the proposed MTLG-Net.

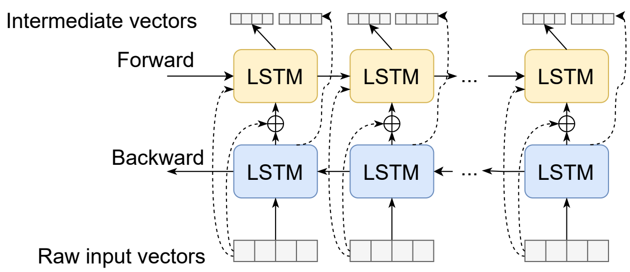

Fig. 2. Network structure of the residual Bi-LSTM layer used in our model. Each arrows indicate the direction of the data flow and $\oplus$ is the addition operation.

Finally, the final graph $G_{global} = (X, y, E, \widetilde{A}_{global})$ is obtained for the global correlated GCN. By utilizing the local and global correlated GCNs, the model could capture the local and global spatial correlations between airports without sacrificing much on the computation complexity.

B. Multi-Decoder Sequence-to-Sequence Model

A multi-decoder Seq2Seq model with the attention module is designed to capture the temporal correlations among flight delay patterns and generate the prediction results in multiple consecutive time intervals. The encoder is composed of one residual bi-directional long-short-term memory (Bi-LSTM) layer, where the sequential input signals are passed down to produce the intermediate context vectors. Figure 2 demonstrates the structure of the residual Bi-LSTM layer used in the multi-decoder Seq2Seq model. An attention module [19] is applied to generate the attention context vectors that replace the encoder's final hidden state output. It helps the model retain more information when processing long input sequences. The context vectors are fed into two separate decoders, namely the arrival delay decoder and the departure delay decoder. Each decoder is constituted of one LSTM layer that produces the prediction results for arrival and departure delays, respectively. An overview of the proposed multi-decoder Seq2Seq model is shown in Figure 3.

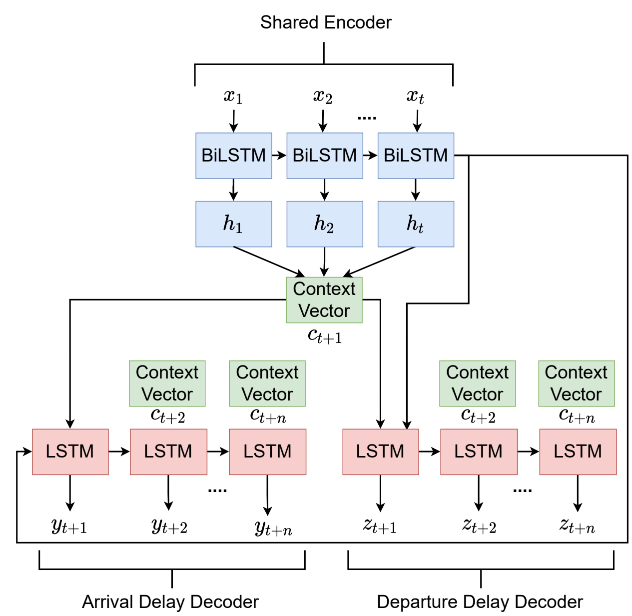

Fig. 3. Network structure of the multi-decoder Seq2Seq model used in this model. x_1, x_2 and x_t are the input sequence from different time step; y_{t+1}, y_{t+2} and y_{t+n} are the output of arrival delay task decode, z_{t+1}, z_{t+2} and z_{t+n} are the output of departure delay task decoder. The context vector c_{t+1}, c_{t+2} and c_{t+n} are generated by the attention module, which serve as the initial input for the decoder. The two tasks share the same encoder.

C. Hierarchical Graph Fusion Module

The cause of flight delays is complex and can be contributed to various factors. It is essential to leverage relevant contextual information to complement the flight record data. Variables that significantly impact air traffic may include flight date and time, flight volume and weather conditions. A common practice for multimodal fusion is to simply concatenate all feature vectors at the early stage. However, this ignores the cross-modality interactions among input sources. In order to exploit the inter-modality correlations among all features, we adopted a hierarchical graph fusion (HGF) approach from our previous work [20]. HGF utilizes a tree-based graph to combine modalities on different levels. The nodes on each level represent different modality combinations and the edge weights measure the similarity between each pair of nodes. The complexity of the combination increases as the graph level gets deeper. The level representation vectors are generated by the weighted sum of all cross-modality interactions in that level. The final output of HGF is the concatenation of all level representation vectors.

D. *Dynamic Multi-task learning Module*

Multi-task learning improves the model generality by training multiple tasks at the same time, which also reduces the potential of overfitting. In MTLG-Net, a dynamic multi-task learning (DMTL) [20] strategy is applied to help the model predict flight arrival and departure delays simultaneously. DMTL works on both task and iteration levels. The task-level component dynamically adjusts the loss weight distribution for each task during the training phase. It guides the model in prioritizing tasks with a less aggressive learning rate. The Iteration-level mechanism incorporates a task weight assignment scalar that applies to all tasks and tries to re-balance the weight assignment at each iteration based on the final training loss.

IV. Experiments and Analyses

A. *Datasets*

Flight Record Data: The Reporting Carrier On-Time Performance data published by BTS (Bureau of Transportation Statistics) is a large-scale public dataset that contains the on-time performance information of flights from all reporting carriers. Relevant attributes include flight date, origin and destination (OD), airport id, actual arrival time, actual departure time, arrival delay, departure delay, flight distance, etc. Records from January 2017 to December 2021 are collected, including 30,940,455 entries, 433 airports, and 802 OD pairs. During the data cleaning process, airports with less than 50 average daily flights, OD pairs with less than 10 average daily flights, and data entries with abnormal values (flight distance less than 100 miles and longer than 3000 miles except those OD pairs containing Hawaii) are dropped. The cleaned dataset contains 25,312,665 entries, 87 airports, and 294 OD pairs. Furthermore, we aggregated the records at the airport level, split the data into 24-hour intervals based on the actual arrival time, and obtained the average hourly arrival/departure delays for each airport.

Meteorological Data: The National Oceanic and Atmospheric Administration (NOAA) provides historical meteorological data collected at weather stations across the US. It contains the average hourly record regarding air temperature, precipitation, sky covers and clouds, sunshine, water, weather type, and wind speed. In this paper, we collected data from weather stations near the selected 87 airports and utilized air temperature, precipitation, and wind speed as part of the input variables.

B. *Experimental Setup*

The dataset is separated into training, validation, and test sets with 60%, 20%, and 20% split. All hyperparameters are tuned on the validation set. ϵ used in calculating sparse matrix normalization is set to 0.1 empirically. Adam [21] is used to optimize the training process and the initial learning rate is set to 0.01. Early drop is applied to prevent overfitting. Input vectors from different modalities are padded with zeroes to ensure a uniform dimension before being fused by HGF. The Mean Average Error (MAE) is selected as the metric to evaluate the model performance:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} \hat{y_i} - y_i \tag{7}$$

where $\hat{y_i}$ is the prediction value and y_i is the ground truth.

C. *Experimental Results*

1) Overall Comparison: We include several baseline methods to compare with the proposed MTLG-NET for flight delay and arrival prediction:

- Autoregressive Integrated Moving Average (ARIMA): a statistical analysis model that predicts future value based on the value from the previous time step. It has been used extensively to solve traffic flow related problems.
- Long Short-Term Memory (LSTM): an RNN based model utilizing a series of gate units to control the information flow.
- Sequence-to-Sequence Model (Seq2Seq): an encoder-decoder deep learning model using several LSTM layers for time series data prediction.
- AG2S-Net [17]: a deep neural network utilizing GCN and Seq2Seq model to predict flight delays in multiple hours intervals.
- DGLSTM [18]: a graph-based deep learning model that employs two adjacency matrices to represent the spherical distance and demand relationship among all airports.

Table I and Table II contain the results of the flight arrival and departure delay predictions in ten consecutive hour intervals, respectively. It can be observed that all methods produce more accurate results on short-term intervals. This can be attributed to the lack of historical patterns in long-time intervals. ARIMA performs poorly comparing to other methods. As a relatively simple model, ARIMA lacks the capacity of capturing more complex time series patterns. LSTM and Seq2Seq models yield a substantial improvement comparing to ARIMA, with Seq2Seq2 leading LSTM's results, especially in long-term predictions. Overall, time series prediction models such as ARIMA, LSTM, and Seq2Seq demonstrate inferior performance since they only capture the temporal dependency. GCN-based state-of-the-art models like

TABLE I

PERFORMANCE COMPARISON OF MTLG-NET WITH DIFFERENT BASELINES FOR AVERAGE HOURLY ARRIVAL DELAY PREDICTION

Method	MAE									
	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
ARIMA	9.331	10.021	11.263	12.834	13.097	14.765	15.932	16.283	17.727	18.316
LSTM	8.915	9.244	10.035	11.037	12.543	13.297	14.504	15.801	16.424	17.688
Seq2Seq	8.524	8.980	9.453	10.041	11.224	12.875	13.662	14.017	15.239	16.433
AG2S-Net	6.323	6.847	7.225	9.988	10.320	12.089	13.095	14.228	14.276	14.301
DGLSTM	5.013	5.877	7.044	9.635	10.098	11.767	12.237	12.768	12.980	13.176
MTLG-Net	**4.573**	**4.842**	**5.925**	**7.852**	**9.374**	**10.237**	**11.535**	**11.570**	**11.335**	**11.659**

TABLE II

PERFORMANCE COMPARISON OF MTLG-NET WITH DIFFERENT BASELINES FOR AVERAGE HOURLY DEPARTURE DELAY PREDICTION

Method	MAE									
	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
ARIMA	9.223	10.010	11.152	12.836	13.041	14.460	15.892	16.290	17.517	18.411
LSTM	8.925	9.237	10.031	11.997	12.493	13.195	14.204	15.752	16.372	17.851
Seq2Seq	8.525	8.930	9.517	10.031	11.324	12.759	13.709	14.136	15.231	16.333
AG2S-Net	6.401	6.848	7.210	9.979	10.293	12.067	13.126	14.231	14.273	14.310
DGLSTM	5.011	5.863	7.041	9.584	10.123	11.742	12.347	12.744	12.982	13.241
MTLG-Net	**4.499**	**4.831**	**5.917**	**7.858**	**9.291**	**10.239**	**11.485**	**11.573**	**11.258**	**11.536**

TABLE III

THE ABLATION STUDY OF EACH COMPONENT'S IMPACT ON FLIGHT ARRIVAL DELAY PREDICTION

Method	MAE									
	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
w/o global GCN	4.988	5.237	6.310	8.247	9.801	10.645	11.984	11.973	12.712	12.680
w/o MTL	4.601	4.875	6.011	7.903	9.408	10.288	11.594	11.583	12.425	12.394
w/o DMTL	4.725	5.023	6.176	8.062	9.603	10.471	11.754	11.772	12.548	12.609
w/o HGF	4.923	5.230	6.297	8.216	9.779	10.639	11.980	11.947	12.700	12.698
MTLG-Net	**4.573**	**4.842**	**5.925**	**7.852**	**9.374**	**10.237**	**11.535**	**11.570**	**11.335**	**11.659**

AG2S-Net and DGLSTM performed noticeably better as they consider both spatial and temporal correlations in the data.

Overall, our proposed MTLG-Net outperforms all baseline methods in the comparisons. It achieves 29.71% lower MAE for the first-hour delay prediction than AG2S-Net and 10.22% lower than DGLSTM. Even for the extreme ten-hour interval delay predictions, our model still outperforms all baseline methods.

2) Ablation Study: We also conducted ablation studies to evaluate the effectiveness of each main component in MTLG-Net.

- Effect of global correlated GCN (w/o global GCN): The global correlated GCN is designed to capture the global spatial dependency. It helps the model learn the hidden patterns that the local correlated GCN does not cover. In this test, the global correlated GCN is removed from the model.
- Effect of multi-task learning (w/o MTL): To verify the effect of multi-task learning, we only train the model to predict the arrival delay and use a single decoder in the Seq2Seq model.
- Effect of dynamic multi-task learning (w/o DMTL): To test the effect of DMTL, the w/o DMTL variant only uses a conventional multi-task learning strategy. It assigns equal loss weights to the tasks when their losses are aggregated to generate the final loss.
- Effect of hierarchical graph fusion (w/o HGF): To evaluate the effect of HGF, we replace HGF with a simple concatenation operation in the w/o HGF test.

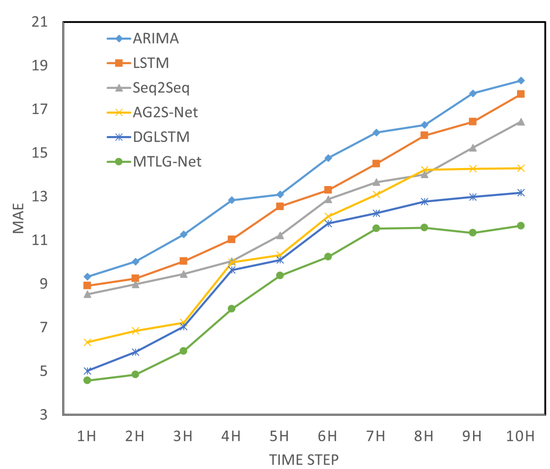

Fig. 4. Arrival delay prediction performance comparison for 10 hours intervals

As shown in Table III, the average MAE of the w/o global GCN test increases by 6.38%, demonstrating the most significant impact on the model performance among all variants. It proves that learning global spatial correlations could

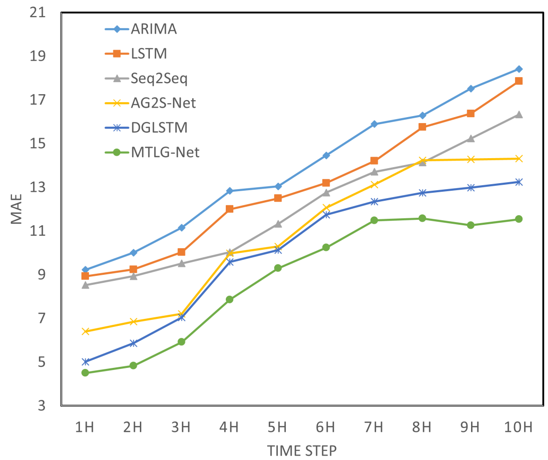

Fig. 5. Departure delay prediction performance comparison for 10 hours intervals

substantially enhance the model's capability to capture the hidden traffic patterns. Next, the w/o HGF variant produces the second-highest average MAE (6.19%). It shows that replacing HGF with simple feature concatenation limits the model's ability to capture the cross-modality interactions among input sources. Furthermore, in w/o MTL and w/o DMTL tests, the w/o MTL test achieves lesser impact than the w/o DMTL test. This shows that arbitrarily assigning equal loss weights for the tasks may not achieve the full potential of multi-task learning. The DMTL approach could efficiently optimize the model training process and achieve better results.

V. Conclusion and Future Work

In this paper, we present a novel multi-task graph-based deep learning framework for flight delay prediction. Two GCNs are introduced to capture the local and global spatial dependencies among airports. A hierarchical graph fusion approach is applied to fuse flight and meteorological data and exploit their cross-modality dependency. The multi-task learning strategy is adopted to predict the flight arrival and departure delays simultaneously to boost the model's generalization capability. A dynamic multi-task learning strategy is developed to optimize the model learning process by automatically adjusting the weight distribution between the two tasks. The proposed MTLG-Net is evaluated on a large-scale carrier on-time performance dataset against several baselines and state-of-the-art methods. Our model outperforms all baselines and demonstrates great performance in multi-time step flight delay prediction. In future studies, the proposed model could be applied to other spatio-temporal forecasting tasks.

Acknowledgment

This research is partially supported by NSF CNS-1952089 and NSF CNS-2125165.

References

[1] FAA, "Cost of delay estimates," 2020, https://www.faa.gov/data_research/aviation_data_statistics/media/cost_delay_estimates.pdf.

[2] T. J. Mofokeng and A. Marnewick, "Factors contributing to delays regarding aircraft during a-check maintenance," in *IEEE technology & engineering management conference (temscon)*, 2017, pp. 185–190.

[3] Y. Ding, "Predicting flight delay based on multiple linear regression," in *IOP conference series: Earth and environmental science*, vol. 81, no. 1. IOP Publishing, 2017, p. 012198.

[4] L. Schaefer and D. Millner, "Flight delay propagation analysis with the detailed policy assessment tool," in *IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat. No. 01CH37236)*, vol. 2, 2001, pp. 1299–1303.

[5] D. Long and S. Hasan, "Improved predictions of flight delays using lminet2 system-wide simulation model," in *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO) and Aircraft Noise and Emissions Reduction Symposium (ANERS)*, 2009, p. 6961.

[6] H. Zhang, W. Wu, S. Zhang, and F. Witlox, "Simulation analysis on flight delay propagation under different network configurations," *IEEE Access*, vol. 8, pp. 103 236–103 244, 2020.

[7] Y. J. Kim, S. Choi, S. Briceno, and D. Mavris, "A deep learning approach to flight delay prediction," in *IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, 2016, pp. 1–6.

[8] G. Gui, F. Liu, J. Sun, J. Yang, Z. Zhou, and D. Zhao, "Flight delay prediction based on aviation big data and machine learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 140–150, 2019.

[9] B. Yu, Z. Guo, S. Asian, H. Wang, and G. Chen, "Flight delay prediction for commercial air transport: A deep learning approach," *Transportation Research Part E: Logistics and Transportation Review*, vol. 125, pp. 203–221, 2019.

[10] N. Chakrabarty, "A data mining approach to flight arrival delay prediction for american airlines," in *9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)*. IEEE, 2019, pp. 102–107.

[11] M. Lambelho, M. Mitici, S. Pickup, and A. Marsden, "Assessing strategic flight schedules at an airport using machine learning-based flight delay and cancellation predictions," *Journal of Air Transport Management*, vol. 82, p. 101737, 2020.

[12] W. Renbiao, L. Jiayi, and Q. Jingyi, "Flight delay prediction model based on dual-channel convolutional neural network," *Journal of Computer Applications*, vol. 38, no. 7, p. 2100, 2018.

[13] H. Yang, Z. Jinfu, and Z. Qinyan, "Airport flight delay prediction based on svm regression," *Journal of Civil Aviation University of China*, vol. 36, no. 1, p. 30, 2018.

[14] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[15] J. Qu, T. Zhao, M. Ye, J. Li, and C. Liu, "Flight delay prediction using deep convolutional neural network based on fusion of meteorological data," *Neural Processing Letters*, vol. 52, no. 2, pp. 1461–1484, 2020.

[16] Z. Guo, G. Mei, S. Liu, L. Pan, L. Bian, H. Tang, and D. Wang, "Sgdan—a spatio-temporal graph dual-attention neural network for quantified flight delay prediction," *Sensors*, vol. 20, no. 22, p. 6433, 2020.

[17] J. Bao, Z. Yang, and W. Zeng, "Graph to sequence learning with attention mechanism for network-wide multi-step-ahead flight delay prediction," *Transportation Research Part C: Emerging Technologies*, vol. 130, p. 103323, 2021.

[18] W. Zeng, J. Li, Z. Quan, and X. Lu, "A deep graph-embedded lstm neural network approach for airport delay prediction," *Journal of Advanced Transportation*, vol. 2021, 2021.

[19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR, San Diego, CA, USA, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[20] T. Wang and S.-C. Chen, "Hierarchical multimodal fusion network with dynamic multi-task learning," in *IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)*, 2021, pp. 208–214.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations (ICLR)*, 2015.