

Article

Multi-Task Dynamic Spatio-Temporal Graph Attention Network: A Variable Taxi Time Prediction Model for Airport Surface Operation

Xiaoyi Yang ¹, Hongyu Yang ¹, Yi Mao ², Qing Wang ¹ and Suwan Yin ^{1,2,*}

¹ College of Computer Science, Sichuan University, Chengdu 610065, China; 2022223040082@stu.scu.edu.cn (X.Y.); yanghongyu@scu.edu.cn (H.Y.); 2022223045143@stu.scu.edu.cn (Q.W.)

² Key Laboratory of Maritime Intelligent Network Information Technology, Ministry of Education, Hohai University, Nanjing 210024, China; maoyi@hhu.edu.cn

* Correspondence: syin@scu.edu.cn

Abstract: Variable taxi time prediction is the core of the Airport Collaborative Decision Making (A-CDM) system. An accurate taxi time prediction contributes to enhancing airport operational efficiency, safety and predictability. The deep dynamic spatio-temporal correlation inherent in airport traffic data is critical for taxi time prediction. However, existing machine learning (deep learning) methods have been unable to thoroughly exploit these correlations. To address this issue, we propose a deep learning-based model called the multi-task dynamic spatio-temporal graph attention network (MT-DSTGAN). Our model also predicts future entire airport traffic flow and taxiing segment traffic flow as auxiliary tasks, with the goal of enhancing the accuracy of aircrafts' taxi time prediction. The proposed MT-DSTGAN model is implemented and assessed through a case study of Beijing Capital International Airport with a real-world dataset. The advantage of the proposed model, which shows better performance in various evaluation metrics, is demonstrated in a comparative study with other baseline works. In summary, the proposed MT-DSTGAN exhibits promising capabilities in perceiving the dynamic changes in the taxiing process of aircraft and demonstrates the ability to capture complex spatio-temporal correlations in airport traffic data.

Keywords: variable taxi time prediction; deep learning; dynamic spatio-temporal correlations; graph neural networks; multi-task learning



Citation: Yang, X.; Yang, H.; Mao, Y.; Wang, Q.; Yin, S. Multi-Task Dynamic Spatio-Temporal Graph Attention Network: A Variable Taxi Time Prediction Model for Airport Surface Operation. *Aerospace* **2024**, *11*, 371. <https://doi.org/10.3390/aerospace11050371>

Academic Editor: Michael Schultz

Received: 24 March 2024

Revised: 26 April 2024

Accepted: 2 May 2024

Published: 8 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

International Civil Aviation Organization (ICAO) predicts that aviation demand will continue to grow annually at a rate of 4.6% until 2032 [1]. The number of flights made globally by the airline industry has reached 39 million in 2019 [2]. Moreover, it is expected that passenger numbers could double to 8.2 billion from 2017 to 2037. This increased demand is expected to contribute to congestion at airports and their surrounding areas, resulting in flight delays, as well as safety and environmental issues. To address this issue, significant research efforts have been dedicated to introducing new operational decision support tools by Single European Sky ATM Research (SESAR) and Next-Generation Air Transportation System (NEXTGEN), for example, the Airport Collaborative Decision Making (A-CDM) system, which aims to improve the overall efficiency of airport operations by enhancing the predictability of aircraft operation processes and optimizing the utilization of existing resources. The successful implementation of A-CDM in 28 European airports highlights the crucial role of accurate taxi time estimation [3]. However, despite the successful application of A-CDM at hub airports, most airports still rely on default taxi times, either based on the experience of air traffic controllers (ATCOs) or on average taxi times derived from historical data. This approach may result in flights failing to arrive at the runway on time. For instance, arriving too early may lead to an excessively long runway queue, while arriving too late may cause flights to miss their runway slots. Therefore, accurate taxi time

prediction is crucial for the efficient operation of airport systems, and it plays an integral role in the implementation of NEXTGEN [4].

Owing to the inherent uncertainties in surface operations, achieving accurate taxi time predictions for individual flights poses a considerable challenge. Each aircraft taxiing on the ground operates under distinct conditions, which are influenced by factors such as varying levels of traffic congestion and fluctuating weather conditions. This complexity persists even when aircraft follow the same taxi route between the runway and the gate. Considering this, existing data-driven research can be categorized into three types: statistical analysis-based methods, traditional machine learning-based methods, and deep learning-based methods. The statistical analysis-based approach is a data-driven approach, which establishes the relationship between the taxi time and its potential influencing factors. This is achieved by investigating the impact of different factors, drawing conclusions on the most important factor for accurately modeling taxi time [5–10]. Another kind of data-driven approach is traditional machine learning, which is achieved by constructing feature sets that may affect aircraft taxi time and utilizing extensive historical operational data to establish taxi time prediction models [11–18]. However, these methods have some drawbacks: (1) Some features defined by experts are only available after the taxiing process [9,16]. (2) Some features may be overlooked or inadequately quantified, thereby reducing the performance of machine learning models.

On the other hand, deep learning-based approaches are adopted to predict aircraft taxi times, which primarily employ neural network models to learn and represent the complex features of data [19–26]. Due to the powerful feature representation learning capabilities of deep learning, it typically outperforms traditional machine learning algorithms. Existing research considers impact factors either at the macroscopic-level [16,20,21,25], the entire airport traffic flow (the number of departing and arriving aircraft), or at the microscopic-level [22], namely traffic conditions on taxiing segments. Few studies based on deep learning so far have considered the inherent deep spatio-temporal correlations in airport traffic data, despite the fact that this correlation is critical for large hub airports, where it is caused by the high dynamism and uncertainty of aircraft ground movements.

In light of this, this paper considers a more efficient neural network architecture that is capable of capturing the intricate spatio-temporal correlations inherent in airport traffic data, which can also be used to predict future entire airport traffic flow and taxiing segment traffic flow as auxiliary tasks. Obtaining estimates related to future traffic conditions is likely to be meaningful for the prediction of taxi time [21,26]. The main contributions of this work are summarized as follows:

- To fully exploit the complex dynamic spatio-temporal correlations among taxiing segments, we develop a novel dynamic spatio-temporal graph attention network to capture the correlations that are inherent in airport traffic data, which are crucial for more accurate taxi time prediction.
- We adopt a multi-task learning approach to predict future entire airport traffic flow and future taxiing segment traffic flow as auxiliary tasks, aiming to enhance the prediction accuracy of the taxi time of aircraft.
- Unlike previous research, this study considers both macroscopic-level and microscopic-level traffic conditions, thus maximizing the potential value derived from airport traffic data.
- We validate the effectiveness of the proposed MT-DSTGAN model using a real-world dataset from Beijing International Airport, and the experimental results demonstrate that MT-DSTGAN outperforms other baseline models in various evaluation metrics.

This paper is organized as follows: Section 2 provides a comprehensive literature review related to taxi time prediction; Section 3 introduces the features that could impact taxi time and provides a detailed methodology; the dataset descriptions, performance metrics, and experimental results are presented in Section 4; finally, conclusions are drawn in Section 5, highlighting the important contributions of our work and possible directions for future work.

2. Literature Review

Aircraft taxi time can be categorized into two components: taxi-out time for departing aircraft, which represents the duration from the gate to the runway; and taxi-in time for arriving aircraft, referring to the period from the runway to the gate. For simplicity, both components are collectively referred to as ‘taxi time’ throughout this study.

2.1. Taxi Time Prediction

Existing variable taxi time prediction research can be distinguished by different criteria or principles; in this work, we mainly focus on the research methodology, which can be categorized into three types: statistical analysis-based methods, traditional machine learning-based methods, and deep learning-based methods.

Early research mainly focused on statistical analysis by establishing the relationship between the taxi time and potential impact factors. Refs. [5,6] found that runway configuration, airline/terminal, downstream limitations, and departure queue size were key factors affecting the taxi time of aircraft. Refs. [7–10] tried to use linear regression, multiple linear regression, and log-linear regression to improve taxi time estimates. These studies examined various factors that could influence an aircraft’s taxi time, including taxi distance, the number of departing and arriving aircraft, the taxi turning angle, the runway configuration, the aircraft type, the weather conditions, etc. However, these methods still fail to adequately capture the complex nonlinear relationships present in airport traffic data.

As traditional machine learning techniques continue to advance, particularly with regard to their ability to mine nonlinear relationships in data, they are seeing increasing application in studies [11–18] focused on predicting the taxi time of aircraft. In general, expert-defined feature sets [16,18] that may affect an aircraft’s taxi time can be categorized into flight properties, airport operational information, traffic conditions, and weather conditions (the details of the influencing factors can be found in Appendix A). Based on the potential factors influencing the taxi time, Refs. [14–16,18] conducted a series of experiments to predict the taxi time of aircraft using various traditional machine learning methods. These studies provided an objective evaluation and thorough analysis of each model, highlighting the superior performance of ensemble learning techniques such as the random forest (RF) and the gradient boosting regression tree (GBRT). However, certain expert-defined features are only accessible at the end of the taxiing process [9,16]. In addition, some effective features may be neglected or inadequately quantified, limiting the performance of the traditional machine learning model.

With advances in neural network architectures and optimization methods, researchers are applying deep learning methods with increasing frequency [19–26] to predict the taxi time of aircraft. Such methods include convolutional neural networks [19,23], wide and deep models [20], graph neural networks [22], and deep metric learning [24,25]. And deep learning-based methods often lead to superior results compared to traditional machine learning algorithms due to their powerful capabilities of feature representation learning. In addition, Refs. [21,26] state that obtaining estimates about future traffic conditions is likely to be meaningful for predicting taxi time.

Although existing methods based on machine learning (deep learning) have achieved good performance in terms of taxi time prediction, these methods are still not sufficient to effectively capture the deep dynamic spatio-temporal correlations inherent in airport traffic data, which are crucial for more accurate taxi time prediction. Most of the studies [16,20,21,25] focus on macroscopic-level influencing factors, such as the number of departing and arriving aircraft, namely entire airport traffic flow, and ignore the impact of microscopic-level influencing factors, such as traffic conditions, on individual taxiing segment; such factors have been considered by only a few studies [22]. While the study [22] considers the impact of the traffic condition on individual taxiing segments, it overlooks the dynamic spatio-temporal correlations among taxiing segments. At large hub airports, the complex spatio-temporal correlations among taxiing segments become particularly significant due to the high dynamism and uncertainty of aircraft ground movements. Therefore, there is a

pressing need to create a more efficient neural network architecture capable of capturing this complexity.

2.2. Graph Neural Networks

In recent years, Graph Neural Networks (GNNs) have been widely used in several domains, including recommender systems [27], biology [28], and traffic forecasting [29], due to their ability to explicitly model correlations in unstructured data, i.e., graph-structured data.

The spatio-temporal graph neural network is a variant of the GNNs designed to capture the spatio-temporal correlations inherent in traffic data. Some works [30–36] have applied it to the travel time-estimating tasks. Ref. [31] supposed that traffic congestion on a road segment could have a high probability of causing traffic congestion on its neighboring road segments in a short time; thus, a road segment's future traffic condition is strongly correlated with the historical traffic conditions of the road segment itself, as well as its neighbor road segments. This effect is even more significant at large hub airports, where numerous aircraft are waiting to depart or arrive. In order to build a fast inference system for travel time prediction, Ref. [32] used a positional encoding technique to record the sequential information of the journey path. Ref. [35] proposed a multi-task learning framework that predicts travel time while also predicting future traffic speeds and taking into account the driver's individual preferences. Building on the work [31], Ref. [36] used a cross-graph attention network to capture the dynamic correlation between the bus route and the bus transportation network.

In light of this, we propose a novel neural network architecture called the multi-task dynamic spatio-temporal graph attention network (MT-DSTGAN) to estimate the taxi time of aircraft. Our architecture captures the complex dynamic spatio-temporal correlations inherent in airport traffic data. We adopt a multi-task learning approach to predict future airport traffic flow and future taxiing segment traffic flow as auxiliary tasks, aiming to enhance the prediction accuracy of the taxi time of aircraft. More accurate taxi time prediction can improve the predictability of aircraft operation processes and enhance airport operational efficiency, providing scientific pre-tactical and tactical support for airport operational control departments and air traffic control units.

3. Methodology

3.1. Problem Definition

We first define the related concepts of taxi time prediction as follows:

- **Airport network.** In this study, the airport network is defined as a directed graph $G = (V_l, E_l)$, where V_l is the set of all taxiing segments and E_l is the edge set. For the sake of convenience, the “taxiing segment” is also referred to as the “link” hereafter. Edge $e_{ij} \in E_l$ denotes the edge connecting link l_i and link l_j , if there is a connection from link l_i to link l_j . See Figure 1 for an overview.
- **Taxi route.** In this study, we define the taxi route as a sequence link on the intended origin–destination path, $route = \{v_1, \dots, v_{|route|}\}$, where $|route|$ is the number of links found along the taxi route.
- **Taxi route query.** A taxi route query is defined as a query tuple $req = \langle r, p, t \rangle$, where r is the taxi route, p represents the flight properties, and t is the start time.
- **Problem: Taxi time prediction.** When an aircraft sends a query request, we aim to predict the aircraft's taxi time based on the taxi route query and historical traffic conditions.

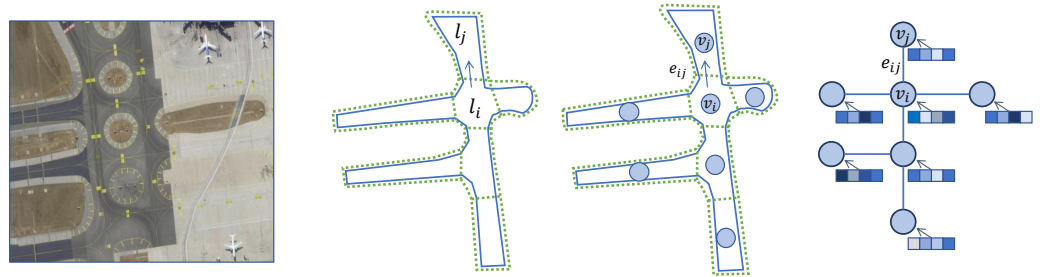


Figure 1. An example of an airport surface network.

3.2. Feature Extraction

Effective features that may affect the taxi time of an aircraft are extracted from a large amount of historical data for taxi time prediction.

- **Airport network:** The airport network describes the layout of taxiways, including the relation between the links. The following features are extracted for each link: ID, length, width, and type (i.e., whether there is a maneuvering area or not). In addition, the graph structure of the airport network is used to describe geographic relationships.
- **Flight properties:** Aircraft type, airline affiliation, arrival or departure status, and taxiing distance are considered in this work.
- **Traffic condition:** It is known that historical traffic conditions have a significant impact on traffic forecasts. At the microscopic level, the impact of features such as taxi speed and traffic flow on individual taxiing segments can be considered. At the macroscopic level, we can consider the average taxi speed and traffic flow of the entire airport. In this study, we only focus on traffic flow as the traffic condition feature. For more details, see Section 4.
- **Temporal information:** The aircraft's taxi time can be influenced by various temporal factors. We consider the hour of the day, minute of the hour, and day of the week as temporal features.

3.3. Model Framework

The Figure 2 shows the architecture of the proposed model framework, the multi-task dynamic spatio-temporal graph attention network (MT-DSTGAN), which consists of four modules: a time series block, dynamic spatio-temporal learning, dynamic airport traffic flow learning, and integration. The time series block module utilizes historical traffic conditions to generate future traffic conditions. The dynamic spatio-temporal learning module fully exploits the complex dynamic spatio-temporal correlations among taxiing segments. The dynamic airport traffic flow learning module determines the impact of the entire airport traffic flow on the taxi time of aircraft. The integration module concatenates the feature representations from all modules' outputs and then employs fully connected layers to output the prediction of the taxi time.

The algorithm aims to train a parametric model capable of learning the mapping of high-dimensional features embedding of traffic conditions, flight properties, temporal information, and airport network to the taxi time of aircraft using a large amount of historical data. (The definitions are shown in Sections 3.1 and 3.2.)

The historical taxiing segments traffic flow (*HTF*), historical entire airport traffic flow (*HAF*), flight properties (*p*), taxi route (*r*), start time (*t*), and airport network (*G*) are input into the model. The model then outputs the aircraft's taxi time ($\widehat{TAT}$).

$$\widehat{TAT} \leftarrow Model(\langle HTF, HAF, p, r, t \rangle | G) \quad (1)$$

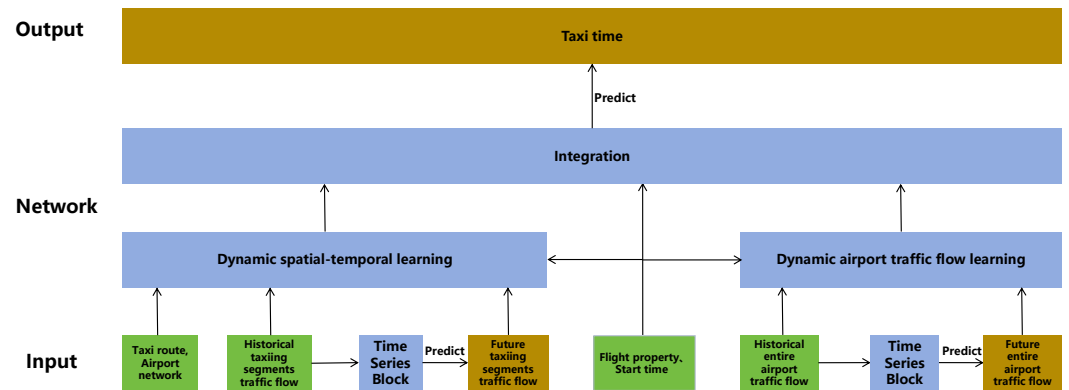


Figure 2. The framework of the proposed MT-DSTGAN model.

This study does not address the path optimization problem, specifically focusing on predicting the taxi time for a given taxi route.

3.4. Time Series Block

Intuitively, the aircraft's taxi time is not only related to historical traffic conditions but also to future traffic conditions. Incorporating future traffic conditions has been shown to facilitate a better prediction of the taxi time of aircraft [21,26]. The time series block module utilizes historical traffic conditions to generate future traffic conditions (as shown in Figure 3), aiming to enhance the accuracy of taxi time predictions. This study employs a state-of-the-art time series module, iTransformer, which is specifically designed for time series data, to predict future traffic conditions. For more details about iTransformer, see [37].

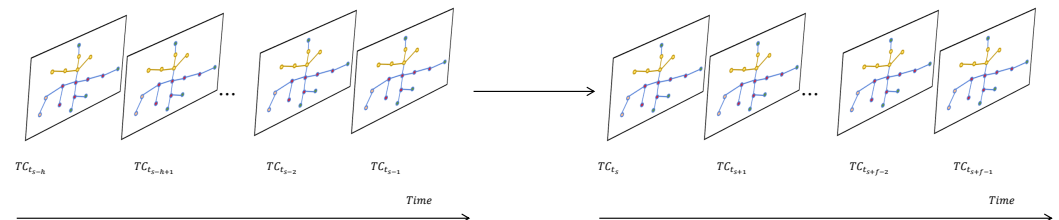


Figure 3. An example of the time series block function.

In this study, we only focus on traffic flow as the traffic condition feature; more details can be found in Section 4. At the microscopic level, the traffic flow of the taxiing segments at time step t is denoted as TF^t . At the macroscopic level, the traffic flow of the entire airport at time step t is denoted as AF^t . The inputs of the module relate to the historical taxiing segments' traffic flow, which is $HTF = [TF^{s-t_h}, \dots, TF^{s-1}] \in R^{t_h \times N_{TF}}$, and historical entire airport traffic flow, which is $HAF = [AF^{s-t_h}, \dots, AF^{s-1}] \in R^{t_h \times N_{AF}}$. The outputs of the module are future taxiing segments traffic flow, which is $\widehat{FTF} = [TF^s, \dots, TF^{s+t_f-1}] \in R^{t_f \times N_{TF}}$, and future entire airport traffic flow, which is $\widehat{FAF} = [AF^s, \dots, AF^{s+t_f-1}] \in R^{t_f \times N_{AF}}$. Here, t_h denotes the input series length, t_f denotes the predict series length, and the current time step is s . Then, the generated future taxiing segments traffic flow and future entire airport traffic flow are fed into the dynamic spatio-temporal learning module and dynamic airport traffic flow learning module, respectively, aiming to enhance the prediction accuracy of the taxi time of aircraft.

3.5. Dynamic Spatio-Temporal Learning

Inspired by related work [31,32,35,36], dynamic spatio-temporal learning is divided into three phases. In the first stage, we assume that the taxi time of the individual taxiing segment is directly influenced by the traffic conditions of the taxiing segment itself, as well as its neighbor taxiing segments [31]. Spatial and temporal relationships between and within taxiing segments are captured using spatial and temporal attention, respectively [35]. This process generates the hidden states of the dynamic spatio-temporal tensor of taxiing segments. In the second stage, a 3D cross spatio-temporal attention network [36] is utilized to adaptively capture the joint dynamic spatio-temporal correlation between the hidden states of flight properties and the hidden states of the taxiing segments through which the taxi route passes and their neighbors. This process generates a higher-level dynamic spatio-temporal representation of the taxi route. In the third stage, the final representation of the taxi route is obtained by adding sequence information to the links using a position embedding technique [32].

Attention mechanisms are widely used techniques in deep learning. By incorporating attention mechanisms, neural networks are able to automatically learn and selectively focus on important information in the input, thus enhancing the model's performance and generalization ability. The well-known self-attention mechanism is proposed in transformer [38], which consists of a QKV structure. Attention scores are computed using QK, after which important information is extracted from V according to these scores. In this module, we use different attention mechanisms to capture the complex spatio-temporal correlations in airport traffic data.

Static spatio-temporal tensors of the taxiing segments need to be constructed first. We leverage the concept of [31]. The dynamic links hidden states $X^{(D)} \in R^{T \times N_l \times d^{(D)}}$ (where $T = t_h + t_f$, N_l denotes the number of links) incorporate historical and future taxiing segments traffic conditions features (generated by the time series block) which are encoded through a fully connected layer. The static links' hidden states $X^{(S)} \in R^{N_l \times d^{(S)}}$ are encoded by discrete features such as the link ID and continuous features such as link length using a lookup table technique and a fully connected layer, respectively. The hidden states of temporal information $X^{(T)} \in R^{T \times d^{(T)}}$ are encoded through a fully connected layer. Then, matrices $X^{(D)}$, $X^{(S)}$, and $X^{(T)}$ are combined into spatio-temporal matrices $X^{(ST)} \in R^{T \times N_l \times d^{(ST)}}$, by expanding matrices $X^{(S)}$ and $X^{(T)}$,

$$X^{(ST)} = \text{Concat}(X_{i,k}^{(D)}, X_i^{(S)}, X_k^{(T)}), \quad (2)$$

$$i \in [1, N_l], k \in [1, t_h + t_f]$$

The static spatio-temporal hidden representation of a taxiing segment i can be expressed as a tensor $X_i^{(ST)} \in R^{T \times d^{(ST)}}$, where T is the number of time slots, which is the sum of historical and future time steps, and $d^{(ST)}$ is the dimension of the hidden representation. $x_{i,k} \in R^{d^{(ST)}}$ represents static spatio-temporal hidden representation of taxiing segment i at time step k .

3.5.1. Spatial Attention and Temporal Attention

(a) Spatial attention

The term "dynamic spatial correlation" describes how the influence weights of other taxiing segments in the airport network change over time, impacting the traffic state of the target taxiing segment. For example, as illustrated in Figure 4, the upstream taxiing segment in shallow blue may negatively affect the traffic state of the green taxiing segment during rush hour. This impact may diminish when the congestion eases. To capture the dynamic spatial correlations, we designed a spatial attention network, the multi-head graph attention network (MGAT) [39], to adaptively model spatial correlations between neighbor taxiing segments.

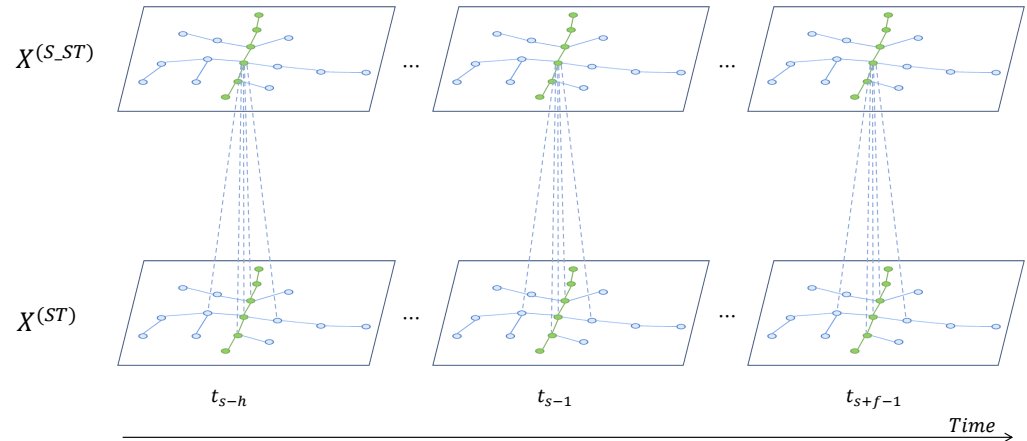


Figure 4. The spatial correlations between neighbor taxiing segments.

In an airport network $G = (V_l, E_l)$, for a taxiing segment $v_i \in V_l$, at time step k , the set of spatial-related taxiing segments is $N_{i,k}^S = \{v_{s,k} | e_{is} \in E_l\} \cup \{v_{i,k}\}$, which includes the taxiing segment at time step k itself and its neighbor taxiing segments at time step k . Then, we use the following graph attention settings to compute the dynamic spatial attention hidden representation of the taxiing segment $x_{i,k}^{(S-ST)}$.

The static spatio-temporal hidden representation of taxiing segment i at time step k , where $x_{i,k}^{(ST)}$ is taken as the *query* of the attention mechanism. The static spatio-temporal hidden representations of spatial-related taxiing segments $X_{i,k,s}^{(ST)}$ ($s \in [1, |N_{i,k}^S|]$) are taken as the *keys* and *values* of the attention mechanism. To be specific, the attention mechanism is formulated as follows:

$$Q_{i,k}^m = W_s^m x_{i,k}^{m,(ST)} \quad (3)$$

$$K_{i,k,s}^m = W_s^m x_{i,k,s}^{m,(ST)} \quad (4)$$

$$V_{i,k,s}^m = W_s^m x_{i,k,s}^{m,(ST)} \quad (5)$$

$$f(Q_{i,k}^m, K_{i,k,s}^m) = \frac{(Q_{i,k}^m)^T \cdot K_{i,k,s}^m}{\sqrt{d}} \quad (6)$$

$$\alpha(Q_{i,k}^m, K_{i,k,s}^m) = \frac{\exp(f(Q_{i,k}^m, K_{i,k,s}^m))}{\sum_s \exp(f(Q_{i,k}^m, K_{i,k,s}^m))} \quad (7)$$

$$x_{i,k}^{m,(S-ST)} = \text{Attention}(Q_{i,k}^m, K_{i,k}^m, V_{i,k}^m) = \sum_s \alpha(Q_{i,k}^m, K_{i,k,s}^m) V_{i,k,s}^m \quad (8)$$

$$x_{i,k}^{(S-ST)} = \text{BN}(\|_{m=1}^M x_{i,k}^{m,(S-ST)} W_{ds} + x_{i,k}^{(ST)}) \quad (9)$$

where M is the number of heads, $\|$ represents the concatenation, W_s^m is the learnable weight matrix of the m^{th} head shared by all taxiing segments, W_{ds} is the learnable weight matrix shared by all taxiing segments, $\exp$ refers to the exponential function, d refers to the embedding dimension, and BN is the batch normalization operation. Then, the dynamic spatial attention hidden representation of taxiing segment i , at time step k can be encoded as $x_{i,k}^{(S-ST)}$ by Equation (9) (the residual connection [40] and batch normalization (BN) [41] are added to prevent feature loss and an internal covariate shift).

The initial input of the spatial attention network is static spatio-temporal hidden representation of taxiing segments $X^{(ST)} \in R^{T \times N_l \times d^{(ST)}}$ and the output is the dynamic spatial attention hidden representation of taxiing segments $X^{(S-ST)} \in R^{T \times N_l \times d^{(ST)}}$.

(b) Temporal attention

The term “dynamic temporal correlation” refers to how the influence weights of a given taxiing segment’s historical traffic conditions on its future traffic state change over time in the airport network. For example, morning rush hour congestion might be influenced by preceding traffic conditions, and this influence may accumulate over time until it diminishes. To dynamically capture relationships across multiple time steps, we implement a temporal attention network based on the mask mechanism of the Transformer [38], as depicted in Figure 5.

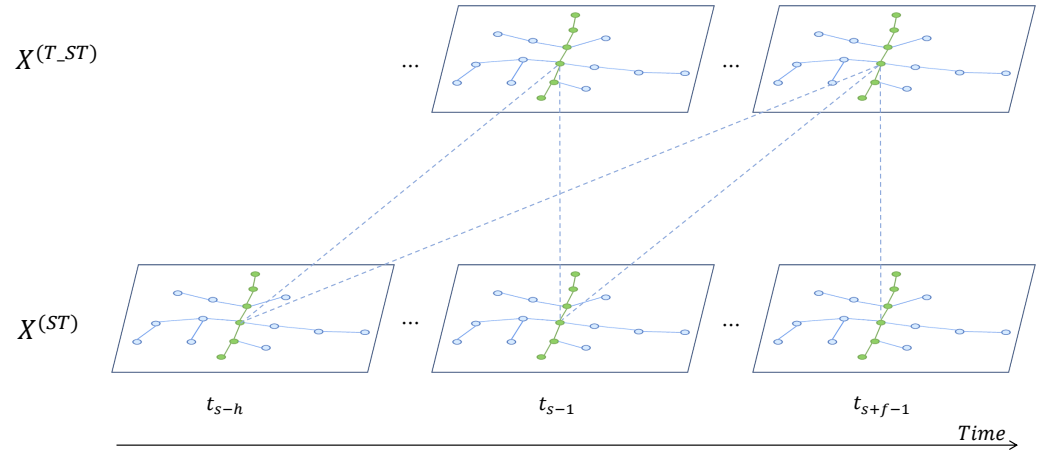


Figure 5. The temporal correlations between different time steps.

In an airport network $G = (V_l, E_l)$, for a taxiing segment $v_i \in V_l$, at time step k , the set of temporal-related taxiing segments is $N_{i,k}^t = \{v_{i,t} | t < k\} \cup \{v_{i,k}\}$, which includes the taxiing segment at step k itself and the taxiing segment before time step k . Then, we use the following temporal attention settings to compute the dynamic temporal attention hidden representation of the taxiing segment $x_{i,k}^{(T-ST)}$.

The static spatio-temporal hidden representation of taxiing segment i at time step k , $x_{i,k}^{(ST)}$ is taken as the *query* of the attention mechanism. The static spatio-temporal hidden representation of temporal-related taxiing segments $X_{i,k,t}^{(ST)}$ ($t \in [1, |N_{i,k}^t|]$) are taken as the *keys* and *values* of the attention mechanism. To be specific, the attention mechanism is formulated as follows:

$$Q_{i,k}^m = W_t^m x_{i,k}^{m,(ST)} \quad (10)$$

$$K_{i,k,t}^m = W_t^m x_{i,k,t}^{m,(ST)} \quad (11)$$

$$V_{i,k,t}^m = W_t^m x_{i,k,t}^{m,(ST)} \quad (12)$$

$$f(Q_{i,k}^m, K_{i,k,t}^m) = \frac{(Q_{i,k}^m)^T \cdot K_{i,k,t}^m}{\sqrt{d}} \quad (13)$$

$$\alpha(Q_{i,k}^m, K_{i,k,t}^m) = \frac{\exp(f(Q_{i,k}^m, K_{i,k,t}^m))}{\sum_t \exp(f(Q_{i,k}^m, K_{i,k,t}^m))} \quad (14)$$

$$x_{i,k}^{m,(T-ST)} = \text{Attention}(Q_{i,k}^m, K_{i,k,t}^m, V_{i,k,t}^m) = \sum_t \alpha(Q_{i,k}^m, K_{i,k,t}^m) V_{i,k,t}^m \quad (15)$$

$$x_{i,k}^{(T-ST)} = \text{BN}(\|_{m=1}^M x_{i,k}^{m,(T-ST)} W_{dt} + x_{i,k}^{(ST)}) \quad (16)$$

where W_t^m is the learnable weight matrix of the m^{th} head shared by all taxiing segments and W_{dt} is the learnable weight matrix shared by all taxiing segments. The meanings of the other symbols remain consistent with those previously determined. Then, the dynamic temporal attention hidden representation of taxiing segment i at time step k can be encoded as $x_{i,k}^{(T-ST)}$ by Equation (16).

The initial input of temporal attention is static spatio-temporal hidden representation of taxiing segments $X^{(ST)} \in R^{T \times N_l \times d^{(ST)}}$ and the output is the dynamic temporal attention hidden representation of taxiing segments $X^{(T-ST)} \in R^{T \times N_l \times d^{(ST)}}$.

(c) Fusion network

The traffic states on a taxiing segment at a given time step are related to the traffic states at previous time steps and the neighboring taxiing segments. To obtain a dynamic spatio-temporal hidden representation of the taxiing segment, the dynamic spatial attention hidden representation and dynamic temporal attention hidden representation of a taxiing segment are fused using a fully connected layer. To be specific, the fusion mechanism is formulated as follows:

$$X^{(DST)} = Dense(Concat(X^{(S-ST)}, X^{(T-ST)})) \quad (17)$$

where *Concat* represents the concatenation and *Dense* represents the fully connected layers with activation functions.

The initial inputs of the fusion networks are the dynamic spatial attention hidden representation $X^{(S-ST)}$ and dynamic temporal attention hidden representation $X^{(T-ST)}$, the output is the dynamic spatio-temporal hidden representation of taxiing segments $X^{(DST)} \in R^{T \times N_l \times d^{(DST)}}$.

3.5.2. Three-Dimensional Cross Spatio-Temporal Attention

Intuitively, the taxi time for an aircraft within a given taxiing segment is related to the historical and future traffic states of the taxiing segment itself and its neighbors, as well as the flight properties. For instance, aircraft of different types may require different amounts of time on the same taxiing segment, even under the same traffic conditions. Therefore, in the second stage, we design a 3D cross spatio-temporal attention network [31,36] to adaptively model the joint dynamic spatio-temporal correlations between the hidden states of flight properties and hidden states of the taxiing segments (generated in the first stage) through which the taxi route passes, as well as those of their neighbors. Then, we obtain the higher-level spatio-temporal representation of the taxi route, as illustrated in Figure 6.

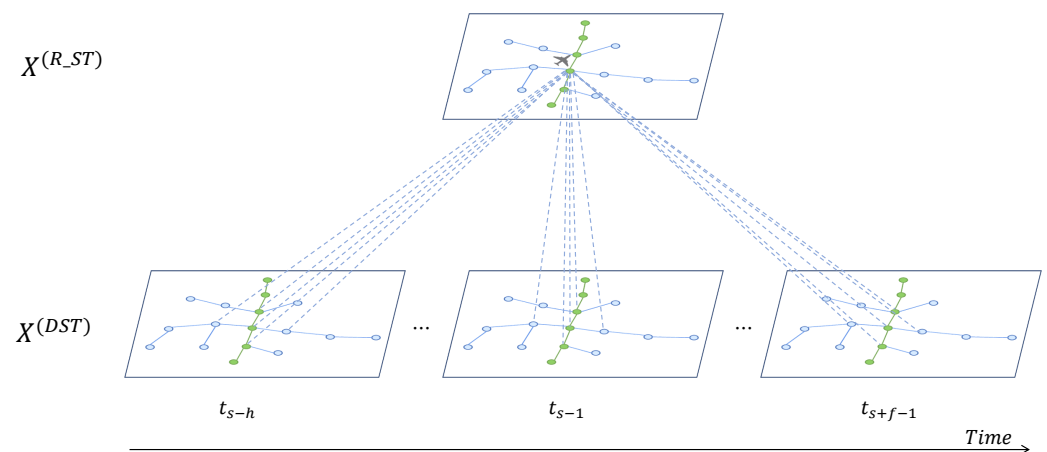


Figure 6. The joint dynamic spatio-temporal correlations between the hidden states of flight properties and the hidden states of taxiing segments (generated in the first stage) through which the taxi route passes and those of their neighbors.

The hidden states of flight properties $x^{(R-C)} \in R^{d^{(DST)}}$, which incorporate features of aircraft type, airline affiliation, arrival or departure status, and total taxi distance, are encoded using a lookup table technique and a fully connected layer. The hidden states of start-time information $x^{(R-T)} \in R^{d^{(DST)}}$ are encoded through a fully connected layer.

Then, the combination of the representation of flight properties $x^{(R-C)}$ and the representation of start time $x^{(R-T)}$ is taken as the *query* of the attention mechanism. The dy-

dynamic spatio-temporal hidden states of taxiing segments through which the taxi route passes and their neighbors in the last t_h and future t_f time slots $x_{i,j,k}^{(DST)}$ ($i \in [1, |route|], j \in [1, |N_{i,k}^S|], k \in [1, t_h + t_f]$) are taken as the *keys* and *values* of the attention mechanism. To be specific, the attention mechanism is formulated as follows:

$$x_r = \text{Concat}(x^{(R_C)}, x^{(R_T)}) \quad (18)$$

$$Q_r^m = W_{rq}^m x_r^m \quad (19)$$

$$K_{i,j,k}^m = W_{rk}^m x_{i,j,k}^{m,(DST)} \quad (20)$$

$$V_{i,j,k}^m = W_{rv}^m x_{i,j,k}^{m,(DST)} \quad (21)$$

$$f(Q_r^m, K_{i,j,k}^m) = \frac{(Q_r^m)^T \cdot K_{i,j,k}^m}{\sqrt{d}} \quad (22)$$

$$\alpha(Q_r^m, K_{i,j,k}^m) = \frac{\exp(f(Q_r^m, K_{i,j,k}^m))}{\sum_{j,k} \exp(f(Q_r^m, K_{i,j,k}^m))} \quad (23)$$

$$x_i^{m,(R_DST)} = \text{Attention}(Q_r^m, K_i^m, V_i^m) = \sum_{j,k} \alpha(Q_r^m, K_{i,j,k}^m) V_{i,j,k}^m \quad (24)$$

$$x_i^{(R_DST)} = \text{BN}(\|_{m=1}^M x_i^{m,(R_DST)} W_{dst}) \quad (25)$$

where W_{rq}^m is the learnable weight matrix of the m^{th} head shared by all aircraft and W_{rk}^m and W_{rv}^m are the m^{th} head learnable weight matrices shared by all taxiing segments. The meanings of the other symbols remain consistent with those previously determined. Then, the higher-level dynamic spatio-temporal attention hidden states of taxiing segment i can be encoded as $x_i^{(R_DST)}$ by Equation (25).

The initial inputs of a 3D cross spatio-temporal attention network are the hidden states of flight properties $x^{(R_C)} \in R^{d^{(DST)}}$, the hidden states of start time $x^{(R_T)} \in R^{d^{(DST)}}$, and the dynamic spatio-temporal hidden states of taxiing segments through which the taxi route passes and their neighbors in the last t_h and future t_f time slots $x_{i,j,k}^{(DST)}$ ($i \in [1, |route|], j \in [1, |N_{i,k}^S|], k \in [1, t_h + t_f]$). And the output is the higher-level dynamic spatio-temporal representation of taxiing segments through which the taxi route passes, $X^{(R_DST)} \in R^{|route| \times d^{(DST)}}$.

3.5.3. Positional Encoding

The taxiing segments along each taxi route exhibit a clear sequential structure, which is significant for training a highly accurate taxi time prediction model. However, recurrent neural networks are very time-consuming during inference, especially with large sequence lengths, and may not meet the real-time requirements of taxi time prediction tasks. Therefore, in this study, a positional encoding technique [32] designed to capture the sequence relationship of taxiing segments is used to incorporate sequence information into the taxiing segments. For the sequence encoding problem, we generate a series of sinusoids:

$$PE(pos, index) = \cos\left(\frac{pos}{10000^{index/dst}}\right) \quad (26)$$

where pos is the taxiing segment's position in a taxi route, $index$ is the dimension index of taxiing segment hidden representation, and dst is the size of taxiing segment hidden representation. We denote $PE_i = PE(i, :)$ as the positional encoding vector for position

i ($i \in [1, |route|]$). Then, the final hidden representation of the taxi route $x^{(R-r)}$ can be encoded as follows:

$$x^{(R-r)} = \sum_i (1 + \beta \cdot PE_i) \cdot x_i^{(R-DST)} \quad (27)$$

where β is a hyper-parameter that is used to control the effect of sequence information.

The initial input of positional encoding is higher-level dynamic spatio-temporal representation of taxiing segments through which the taxi route passes, $X^{(R-DST)} \in R^{|route| \times d^{(DST)}}$. And the output is the final hidden representation of the taxi route $x^{(R-r)} \in R^{d^{(DST)}}$.

3.6. Dynamic Airport Traffic Flow Learning

The aircraft's taxi time on a given taxi route is influenced not only by the microscopic-level traffic conditions and the traffic flow on individual taxiing segments, but also by the macroscopic-level traffic conditions and the traffic flow throughout the entire airport [16]. Hence, the dynamic relationship between the hidden states of flight properties and the hidden states of the entire airport traffic flow is adaptively modeled using the cross-attention mechanism to derive the dynamic hidden representation of the entire airport traffic flow $x^{(R-AF)}$.

The hidden states of entire airport traffic flow $X^{(AF)} \in R^{T \times d^{(af)}}$ (where $T = t_h + t_f$), which incorporate features of the historical and future entire airport traffic flow (generated by a time series block) are encoded through a fully connected layer.

Then, the combination of the representation of flight properties $x^{(R-C)}$ and the representation of start time $x^{(R-T)}$ is taken as the *query* of the attention mechanism. The hidden states of the entire airport traffic flow in the last t_h and future t_f time slots are taken as the *keys* and *values* of the attention mechanism. To be specific, the attention mechanism is formulated as follows:

$$x_r = \text{Concat}(x^{(R-C)}, x^{(R-T)}) \quad (28)$$

$$Q_r^m = W_{fq}^m x_r^m \quad (29)$$

$$K_k^m = W_{fk}^m x_k^{m,(AF)} \quad (30)$$

$$V_k^m = W_{fv}^m x_k^{m,(AF)} \quad (31)$$

$$f(Q_r^m, K_k^m) = \frac{(Q_r^m)^T \cdot K_k^m}{\sqrt{d}} \quad (32)$$

$$\alpha(Q_r^m, K_k^m) = \frac{\exp(f(Q_r^m, K_k^m))}{\sum_k \exp(f(Q_r^m, K_k^m))} \quad (33)$$

$$x^{m,(R-AF)} = \text{Attention}(Q_r^m, K^m, V^m) = \sum_k \alpha(Q_r^m, K_k^m) V_k^m \quad (34)$$

$$x^{(R-AF)} = \text{BN}(\|_{m=1}^M x^{m,(R-AF)} W_{daf}) \quad (35)$$

where W_{fq}^m is the learnable weight matrix of the m^{th} head shared by all aircraft, W_{fk}^m and W_{fv}^m are the m^{th} head learnable weight matrix shared by airport traffic flow at different time slots. The meanings of the other symbols remain consistent with those previously determined. Then the dynamic hidden representation of entire airport traffic flow can be encoded as $x^{(R-AF)}$ by Equation (35).

The initial inputs of cross-attention network are the hidden states of flight properties $x^{(R-C)} \in R^{d^{(DST)}}$, the hidden states of start time $x^{(R-T)} \in R^{d^{(DST)}}$, and the hidden states of entire airport traffic flow at multiple time steps $X^{(AF)} \in R^{T \times d^{(af)}}$. And the output is the dynamic hidden representation of entire airport traffic flow $x^{(R-AF)} \in R^{d^{(af)}}$.

3.7. Integration

Based on the above components, we concatenate all the latent representations $x^{(R_C)}$, $x^{(R_T)}$, $x^{(R_r)}$, $x^{(R_{AF})}$ as H_t , which is then fed into a fully connected layer to output the taxi time prediction $\widehat{TAT}$, defined as:

$$\widehat{TAT} = W_{ht}^T H_t + b_{ht} \quad (36)$$

where W_{ht} and b_{ht} are the parameters to be learned. Finally, we use a linear transformation to generate the final output result.

3.8. Multi-Task Learning

To enhance the accuracy of aircraft taxiing time prediction, the model predicts future entire airport traffic flow and future taxiing segments traffic flow as auxiliary tasks. This forms a multi-task learning framework, as depicted in Figure 7.

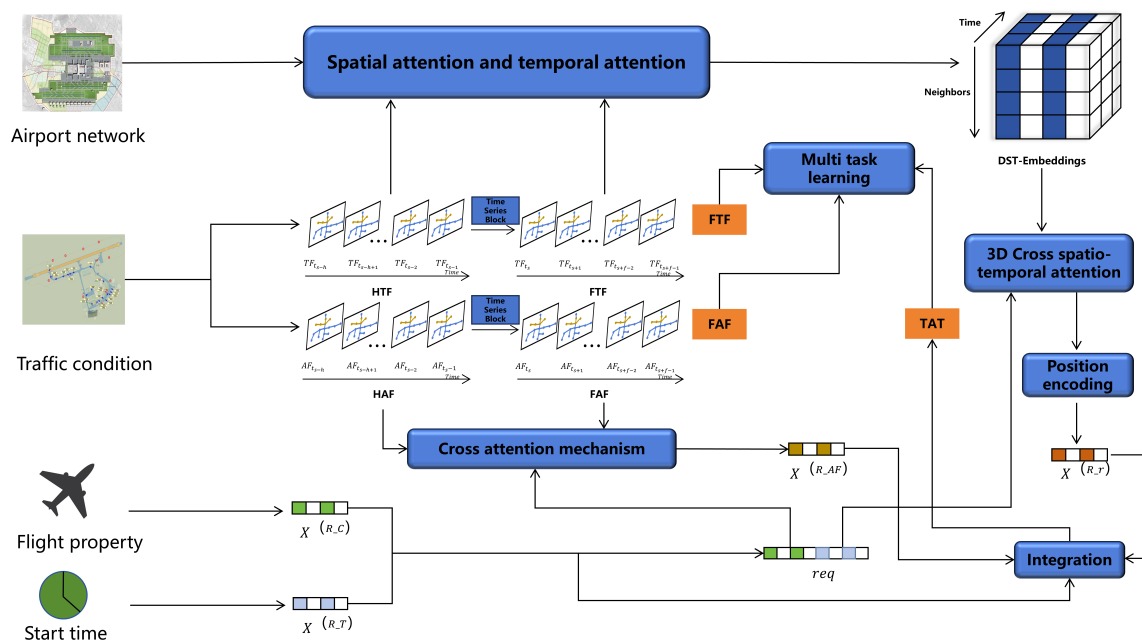


Figure 7. The model structure of the proposed method MT-DSTGAN.

By implementing the aforementioned modules, we obtain the target prediction results of the taxi time of aircraft $\widehat{TAT}$, future taxiing segments traffic flow $\widehat{FTF}$, and future entire airport traffic flow $\widehat{FAF}$.

$$\hat{Y} = \begin{cases} \hat{Y}^1 = \widehat{TAT}, \hat{Y}^1 \in R^1 \\ \hat{Y}^2 = \widehat{FTF}, \hat{Y}^2 \in R^{t_f \times N^l \times 1} \\ \hat{Y}^3 = \widehat{FAF}, \hat{Y}^3 \in R^{t_f \times 1} \end{cases} \quad (37)$$

where t_f denotes the predict series length and N^l denotes the number of taxiing segments.

The loss function of MT-DSTGAN corresponding to the multi-task layer is defined as the mean absolute percentage error (MAPE) between observed values Y and predicted values $\hat{Y}$,

$$L(\theta) = (1 - \lambda_1 - \lambda_2) \left| \frac{Y^1 - \hat{Y}^1}{Y^1} \right| + \frac{\lambda_1}{t_f \times N^l} \sum_k \sum_i \left| \frac{Y_{k,i}^2 - \hat{Y}_{k,i}^2}{Y_{k,i}^2} \right| + \frac{\lambda_2}{t_f} \sum_k \left| \frac{Y_k^3 - \hat{Y}_k^3}{Y_k^3} \right| \quad (38)$$

where θ denotes all the learnable parameters in MT-DSTGAN, λ_1 and λ_2 represent loss weights, t_f denotes the predict series length, and N_l denotes the number of links.

4. Experiments

In this section, a series of experiments is conducted to measure the performance of MT-DSTGAN. All our approaches are implemented on Python 3.10, the deep learning models are constructed by the Pytorch 2.0.1 framework, and other methods are constructed by Scikit-learn 1.2.1 python package.

4.1. Data Preparation

To validate the proposed method MT-DSTGAN, operational data from Beijing Capital International Airport in China, covering the period from 13 September 2017 to 19 September 2017, is collected for data processing and analysis. In 2019, Beijing Capital International Airport (PEK) ranked second in the world, with 100 million passengers, making it one of the busiest hub airports in the world.

This dataset involves about 11,993 pieces of information on flights, which include (1) aircraft type: a designator with a four-character alphanumeric code defined by the International Civil Aviation Organization (ICAO); (2) departure or arrival status: departing aircraft taxi from the gate to the runway, while arriving aircraft taxi from the runway to the gate; (3) gate: the gate assigned to the aircraft; (4) runway: the runway assigned to the aircraft; (5) taxi route: the assigned taxi route to the aircraft; (6) start time: off-block time for departing aircraft or landing time for arriving aircraft; (7) end time: take-off time for departing aircraft or on-block time for arriving aircraft; (8) time slot: the time slot to which the start time belongs (the time interval set to 5 min) denotes the temporal features of the start time.

This study adopts a definition of airport traffic flow to the article [16]. At the macroscopic level, for traffic flow of the entire airport AF , the number is counted if the aircraft's taxiing process occurs within this time slot as Equation (39). At the microscopic level, for traffic flow of individual taxiing segment TF , the number is counted if the aircraft's taxiing process occurs within this time slot, as shown in Equation (40).

$$AF(T_{\text{slot}}) = \sum_i \text{count}(i), \quad (39)$$

$$T_{\text{slot}} \cap [t_{\text{start}}(i), t_{\text{end}}(i)] \neq \emptyset$$

where $t_{\text{start}}(i)$ denotes the start time of i th aircraft, $t_{\text{end}}(i)$ denotes the end time of i th aircraft.

$$TF(T_{\text{slot}}) = \sum_i \sum_j \text{count}(\text{link}_j), \quad (40)$$

$$T_{\text{slot}} \cap [t_{\text{start}}(i), t_{\text{end}}(i)] \neq \emptyset, j \in \text{Route}(i)$$

where $t_{\text{start}}(i)$ denotes the start time of the i th aircraft, $t_{\text{end}}(i)$ denotes the end time of i th aircraft, $\text{Route}(i)$ denotes the links through which the i th aircraft's taxi route passes.

An example of the processed dataset is listed in Table 1, an example of traffic flow of the entire airport is listed in Table 2, and an example of the traffic flow of individual taxiing segments is shown in Table 3. In our dataset, which spans from 13 September 2017 to 19 September 2017, there are 2,016 time slices with 5-minute intervals each. Our dataset consists of a total of 484 taxiing segments. And the dataset is randomly divided into a training set and testing set in the ratio of 80% and 20%, representing 9595 and 2398 entries, respectively.

Table 1. An example of the processed dataset.

Aircraft Type	Departure/Arrival	Gate	Runway	Taxi Route	Start Time	End Time	Time Slot
A320	Departure	509	01	¹ R1	13 September 2017 09:35:05	13 September 2017 09:53:25	13 September 2017 09:40:00 ²
A319	Departure	805	36R	R2	13 September 2017 09:37:05	13 September 2017 09:58:45	13 September 2017 09:40:00
B747	Arrival	213	36L	R3	13 September 2017 09:38:05	13 September 2017 09:50:35	13 September 2017 09:40:00
...	...	...	...	...	...	...	...
B738	Arrival	213	36L	R3	18 September 2017 23:59:05	19 September 2017 00:10:35	19 September 2017 00:00:00
...	...	...	...	...	...	...	...

¹ denotes a sequence of links through which the assigned taxi route passes. ² denotes the temporal features of start time, the time interval set to 5 min. Time slot 13 September 2017 09:40:00 means that the time from 13 September 2017 09:35:00 to 13 September 2017 09:40:00.

Table 2. An example of entire airport traffic flow.

Time Slot ¹	Entire Airport Traffic Flow
13 September 2017 06:55:00	14
13 September 2017 07:00:00	15
13 September 2017 07:05:00	15
13 September 2017 07:10:00	15
13 September 2017 07:15:00	18
13 September 2017 07:20:00	18
13 September 2017 07:25:00	20
13 September 2017 07:30:00	20
13 September 2017 07:35:00	24
...	...

¹ In our dataset, which spans from 13 September 2017 to 19 September 2017, there are 2016 time slices with 5-minute intervals each.

Table 3. An example of taxiing segments traffic flow.

Time Slot\Link ID ¹	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6	...	Link 300	Link 301	...
13 September 2017 06:55:00	0	0	0	0	0	0	...	1	0	...
13 September 2017 07:00:00	0	0	0	0	0	0	...	2	1	...
13 September 2017 07:05:00	0	0	0	0	0	0	...	2	1	...
13 September 2017 07:10:00	0	0	0	0	0	0	...	1	1	...
13 September 2017 07:15:00	0	0	0	0	0	1	...	3	3	...
13 September 2017 07:20:00	0	0	0	0	0	1	...	2	2	...
13 September 2017 07:25:00	0	0	0	0	0	1	...	2	2	...
13 September 2017 07:30:00	0	0	0	0	0	0	...	2	2	...
13 September 2017 07:35:00	0	0	0	0	0	0	...	3	3	...
...	...	...	...	...	...	...	...	...	...	...

¹ our dataset consists of a total of 484 links.

4.2. Experiment Configuration

4.2.1. Evaluation Metric

In this study, the mean absolute percentage error (MAPE⁽⁴¹⁾) and the root mean square error (RMSE⁽⁴²⁾) and the $\pm 1, 3, 5$ min error window accuracy⁽⁴³⁾ are chosen as the evaluation metrics. For consistency, taxi time is measured in seconds. For the evaluation metric value of MAPE and RMSE, a smaller evaluation metric value means a better performance, while for an evaluation metric value of $\pm 1, 3, 5$ min error window accuracy, a higher evaluation metric value means a better performance.

- MAPE

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right|, \quad (41)$$

- RMSE

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \quad (42)$$

- $\pm k$ min Error Window Accuracy

$$\pm k \text{ min error window accuracy} = \frac{\sum_{i=1}^n \text{count}(if (Y_i - \hat{Y}_i) \leq k)}{n} \quad (43)$$

where n is the total number of flights, Y_i and $\hat{Y}_i$ are the observed and predicted taxi times of the aircraft, respectively.

4.2.2. Training Details

In this study, the extracted features, which include continuous and categorical variables in the forms of discrete and real values, are uniformly projected into 64-dimensional embedding vectors. The hidden state dimension is fixed at 64. We use a 5 min time interval and consider traffic conditions with historical time slots $t_h = 6$ and future time slots $t_f = 6$. The batch size is set to 64. The epoch is set to 200. There are 484 links. ReLU is selected as the activation function for all hidden layers. The hyper-parameter of positional encoding β is set to one. The loss weights λ_1, λ_2 are set to 0.1 and 0.1, respectively. During the training process, the model parameters are optimized using the Adam optimizer with the default settings, except for the learning rate, which is set to 0.001.

4.3. Comparative Baselines

In this study, the following baseline models are also designed to further confirm the superior performance of our proposed approach.

- **HA.** The average taxi time for origin–destination pairs in the historical data (only the training dataset) is taken as the prediction results;
- **DNN.** We also use a two fully-connected layers neural network with ReLU activation to predict the aircraft's taxi time.
- **GBRT.** Gradient boosted regression trees (GBRT), which is well known for its outstanding performance in previous works [15,22], are used for aircraft's taxi time prediction.
- **RF.** Random forests (RF), which have achieved a superior prediction performance in previous works [17,18], are used for aircraft's taxi time prediction.

4.4. Experimental Results

The experimental results are listed in Table 4, in which the MT-DSTGAN and other baselines for aircraft taxi time prediction are reported. A visualization of aircraft taxi time prediction is shown in Figure 8. The results show that the proposed approach MT-DSTGAN achieves the best performance for all indicators. In contrast to baseline methods, the MT-DSTGAN obtains relative percent reductions in MAPE ranging from 11.31% to 45.38%, in RMSE from 9.22% to 35.16%, and relative percent improvements in the ± 1 error window accuracy from 13.90% to 41.64%, the ± 3 error window accuracy from 3.90% to 11.23%, the ± 5 error window accuracy from 1.13% to 5.68%, respectively.

GBRT and RF outperform HA and DNN, demonstrating the superiority of the ensemble learning approach, as demonstrated by previous works [15,18,25]. RF performs well when the input features are sufficiently effective and shows strong performance on our dataset. However, these methods still not sufficient to capture the complex dynamic spatio-temporal correlations inherent in airport traffic data, and thus their performances

are weaker than that of MT-DSTGAN. Additionally, we anticipate that our method will exhibit superior performance on a larger dataset compared to other baseline methods.

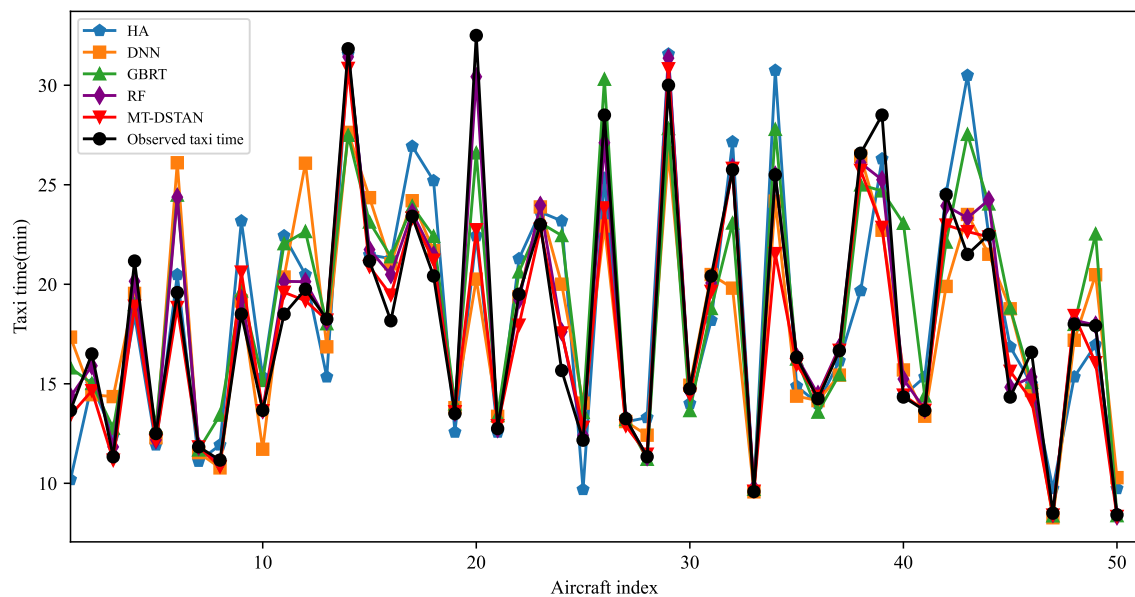


Figure 8. Comparison results of taxi time prediction based on different methods (50 randomly selected samples).

Table 4. Performance of MT-DSTGAN and baseline methods.

Method\Metrics	MAPE	RMSE (s)	± 1 min	± 3 min	± 5 min
HA	11.48%	224.78	45.87%	81.65%	90.99%
DNN	10.63%	217.94	49.54%	82.65%	91.74%
GBRT	9.81%	173.70	44.20%	81.73%	93.08%
RF	7.07%	160.54	57.04%	87.41%	95.08%
MT-DSTGAN(this work)	<u>6.27%</u>	<u>145.73</u>	<u>64.97%</u>	<u>90.82%</u>	<u>96.16%</u>

underlined text denotes best performance.

4.5. Ablation Experiment

In this subsection of the ablation experiment, the time series module for predicting future traffic conditions is removed to investigate whether the time series module is effective for accurately predicting the taxi time of aircraft. The ablation experiment can be divided into categories with no time series module (MT-DSTGAN_1), no taxiing segments traffic flow prediction (MT-DSTGAN_2), and no airport traffic flow prediction (MT-DSTGAN_3). The experimental results are listed in Table 5.

The experimental results show that incorporating future traffic conditions leads to a slight improvement in terms of prediction accuracy. It is worth noting that during the experiment, we observed that predicting future traffic conditions is indeed a challenging task. The model requires more training time to achieve improved results. Sometimes, the inclusion of additional noise can lead to worse results. A trade-off needs to be made between computing resources and model performance.

Table 5. Performance of ablation experiment.

Method \ Metrics	MAPE	RMSE (s)	± 1 min	± 3 min	± 5 min
MT-DSTGAN_1	6.46%	174.39	66.22%	89.57%	94.91%
MT-DSTGAN_2	6.38%	169.58	<u>66.81%</u>	90.07%	95.41%
MT-DSTGAN_3	6.36%	162.95	65.97%	90.49%	95.83%
MT-DSTGAN (this work)	<u>6.27%</u>	<u>145.73</u>	64.97%	<u>90.82%</u>	<u>96.16%</u>

underlined text denotes best performance.

5. Conclusions

In this study, we built a deep learning-based model, which we named the multi-task dynamic spatio-temporal graph attention network (MT-DSTGAN), to tackle the variable taxi time prediction task. MT-DSTGAN leverages a dynamic spatio-temporal graph attention network to capture complex spatio-temporal correlations inherent in airport traffic data. Additionally, it employs a multi-task learning approach to concurrently predict future entire airport traffic flow and future taxiing segments traffic flow as auxiliary tasks, with the goal of improving the accuracy of taxi time predictions and maximizing the potential value derived from airport traffic data.

The proposed approach is validated using a real-world airport traffic dataset, and the experimental results demonstrate its superior performance compared to other baseline methods, including various data-driven models. In contrast to baseline methods, the MT-DSTGAN obtains relative percent reductions in MAPE ranging from 11.31% to 45.38%, in RMSE from 9.22% to 35.16%, and relative percent improvements on the ± 1 error window accuracy from 13.90% to 41.64%, the ± 3 error window accuracy from 3.90% to 11.23%, and the ± 5 error window accuracy from 1.13% to 5.68%, respectively. In summary, the proposed MT-DSTGAN exhibits promising capabilities in perceiving the dynamic traffic states changes in the taxiing process of aircraft and demonstrates the ability to capture complex spatio-temporal correlations in airport traffic data.

In the future, we are planning to integrate explicit airport operation rules into neural networks. These rules, including safety intervals, taxiing speed limits, and runway configurations, among others, are crucial aspects of airport operations. Incorporating them into our model is expected to improve prediction accuracy. Accurately predicting future airport traffic conditions using various methods is meaningful for improving taxi time prediction. Furthermore, developing a model to assist controllers in decision-making, as discussed in previous works [17,25], is an important direction for airport operation systems. Our model can be integrated into these frameworks to achieve better performance.

Author Contributions: Conceptualization: X.Y. and S.Y.; methodology: X.Y., H.Y. and S.Y.; Data curation: X.Y. and Q.W.; formal analysis, X.Y.; funding acquisition: H.Y., Y.M. and S.Y.; writing—original draft preparation, X.Y. and S.Y.; writing—review and editing, X.Y., H.Y. and S.Y.; software: X.Y. and Q.W.; supervision: H.Y. and S.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China through grants 72201184 and U20A20161, and the Natural Science Foundation of Sichuan Province through grants 23NSFSC3451.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to confidentiality agreements.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

ICAO	International Civil Aviation Organization.
SESAR	Single European Sky ATM Research.
NEXTGEN	Next-Generation Air Transportation System.
A-CDM	Airport Collaborative Decision Making.
HTF	Historical taxiing segments traffic flow.
HAF	historical entire airport traffic flow.
FTF	Future taxiing segments traffic flow.
FAF	future entire airport traffic flow.
TAT	Taxi time.
GNNs	Graph Neural Networks.
MGAT	Multi-head Graph Attention Network.
MAPE	Mean Absolute Percentage Error.
RMSE	Root Mean Square Error.
HA	Historical Average.
DNN	Deep Neural Network.
RF	Random Forest.
GBRT	Gradient Boosted Regression Tree.
MT-DSTGAN	Multi-task dynamic spatio-temporal graph attention network.

Appendix A

Below, we have listed the potential influencing factors affecting taxi time, with some information extracted from [18,21,25]. We did not explore all of the following influencing factors and only used partial factors in Section 3.2. We plan to investigate the effects of other factors in future studies.

Table A1. Features of flight properties.

Features	Type	Description
Flight number	Categorical	Flight number information, such as CZ441 and MU735
Flight property	Binary	Passenger (1) and freight (0) transport
Aircraft type	Categorical	Type of aircraft, such as Airbus A320 and Boeing 737
Airline	Categorical	The airline to which the flight belongs, such as CZ and MU
Runway	Categorical	Runway for aircraft takeoff, such as 16R and 34L
Gate	Categorical	Gate for aircraft pushed back, such as 117L and 117R
Apron	Categorical	Apron category used by aircraft, such as T1 and T2
Boarding port	Categorical	Boarding port used by aircraft, such as C50 and D69
Boarding character	Categorical	Boarding characters used by aircraft: domestic, international, and mixed
Gate type	Binary	Corridor bridge (1) and remote (0) gate
Engine type	Binary	Two engines (1) or not (0)
Flag limited	Binary	Limited flight (1) or not (0)
Taxi distance	Numerical	Aircraft taxi distance based on runway and gate
Turning angle	Numerical	Sum of turning angle in degrees in the taxiing process
Month	Numerical	Month of the flight
Day	Numerical	Day of the flight
Weekday	Numerical	Weekday of the flight
Hour	Numerical	Hour of the flight
Minute	Numerical	Minute of the flight

Table A2. Features of airport operational information.

Features	Type	Description
Runway configuration	Categorical	Runway operating patterns based on wind direction, such as 25L 07R, 07L 25R, and 25R 25R
Airport network	Categorical and Numerical	The layout of taxiways including the connection between the taxiing segments and the ID, length, and width of taxiing segments

Table A3. Features of weather conditions.

Features	Type	Description
Wind direction	Numerical	Average wind direction, in degrees
Wind speed	Numerical	Average wind speed, in meters per second
Direction change	Binary	Wind direction has changed significantly (1) or not (0)
Visibility	Numerical	Airport visibility, in meters
Precipitation	Categorical	Type of precipitation, such as rain, snow, and hail
Vision obstruction	Categorical	Type of vision obstruction, such as fog and haze
Cloud cover	Numerical	Number of units for cloud cover
Cloud ceiling	Numerical	Cloud ceiling, in meters
Temperature	Numerical	Airport temperature, in Celsius
Dew point	Numerical	Airport dew point, in Celsius
Pressure	Numerical	Air pressure, in hectopascals
Flag precipitation	Binary	Precipitation (1) or not (0)
Flag vision	Binary	Vision obstruction (1) or not (0)
Flag thunderstorm	Binary	Thunderstorm (1) or not (0)

Table A4. Features of traffic conditions.

Macroscopic-level		
Features	Type	Description
NDepDep	Numerical	Number of other aircraft on the way to runway when current aircraft on the way to runway during a time period
NDepArr	Numerical	Number of other aircraft on the way to gate when current aircraft on the way to runway during a time period
NArrDep	Numerical	Number of other aircraft on the way to runway when current aircraft on the way to gate during a time period
NArrArr	Numerical	Number of other aircraft on the way to gate when current aircraft on the way to gate during a time period
Entire airport traffic flow	Numerical	Number of other aircraft on the way to runway or gate when currently on the way to runway or gate during a specific time period
AvgSpdLast5Dep	Numerical	Average speed of 5 latest departing aircraft
AvgSpdLast5Arr	Numerical	Average speed of 5 latest arriving aircraft
AvgSpdLast5	Numerical	Average speed of 5 latest aircraft
AvgSpdLast10Dep	Numerical	Average speed of latest departing 10 aircraft
AvgSpdLast10Arr	Numerical	Average speed of latest 10 arriving aircraft
AvgSpdLast10	Numerical	Average speed of latest 10 aircraft
Microscopic-level		
Features	Type	Description
Taxi speed	Numerical	Taxi speed of aircraft on individual taxiing segment during a time period
Taxiing segments traffic flow	Numerical	Number of aircraft passed the individual taxiing segment during a time period

References

1. ICAO. ICAO Long-Term Traffic Forecast: Passenger and Cargo. Available online: <https://www.icao.int/safety/ngap/NGAP8Presentations/ICAO-Long-Term-Traffic-Forecasts-July-2016.pdf> (accessed on 24 March 2024).
2. Association, I.A.T. International Air Transport Association Annual Review 2019. Available online: <https://www.iata.org/en/publications/annual-review/> (accessed on 24 March 2024).
3. Eurocontrol. Airport Collaborative Decision Making (A-CDM) Implementation Manual. Available online: <https://www.eurocontrol.int/publication/airport-collaborative-decision-making-cdm-implementation-manual> (accessed on 24 March 2024).
4. Yim, W.F. The Next Generation of Air Transportation System (NextGEN). In *Challenges and Advances in Sustainable Transportation Systems*; ASCE Publishing: Reston, VA, USA, 2014; pp. 25–32.
5. Idris, H.; Clarke, J.P.; Bhuva, R.; Kang, L. *Queuing Model for Taxi-Out Time Estimation*; Technical Report; MIT Library: Cambridge, MA, USA, 2001.

6. Simaiakis, I.; Pyrgiotis, N. An analytical queuing model of airport departure processes for taxi out time prediction. In Proceedings of the 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Worth, TX, USA, 13–15 September 2010; p. 9148.
7. Jordan, R.; Ishutkina, M.A.; Reynolds, T.G. A statistical learning approach to the modeling of aircraft taxi time. In Proceedings of the 29th Digital Avionics Systems Conference, Salt Lake City, UT, USA, 3–7 October 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 1.B.1-1–1.B.1-10.
8. Srivastava, A. Improving departure taxi time predictions using ASDE-X surveillance data. In Proceedings of the 2011 IEEE/AIAA 30th Digital Avionics Systems Conference, Seattle, WA, USA, 16–20 October 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 2B5-1–2B5-14.
9. Ravizza, S.; Atkin, J.A.; Maathuis, M.H.; Burke, E.K. A combined statistical approach and ground movement model for improving taxi time estimations at airports. *J. Oper. Res. Soc.* **2013**, *64*, 1347–1360. [\[CrossRef\]](#)
10. Lordan, O.; Sallan, J.M.; Valenzuela-Arroyo, M. Forecasting of taxi times: The case of Barcelona-El Prat airport. *J. Air Transp. Manag.* **2016**, *56*, 118–122. [\[CrossRef\]](#)
11. Lee, H.; Malik, W.; Zhang, B.; Nagarajan, B.; Jung, Y.C. Taxi time prediction at Charlotte Airport using fast-time simulation and machine learning techniques. In Proceedings of the 15th AIAA Aviation Technology, Integration, and Operations Conference, Dallas, TX, USA, 22–26 June 2015; p. 2272.
12. Lee, H.; Malik, W.; Jung, Y.C. Taxi-out time prediction for departures at Charlotte airport using machine learning techniques. In Proceedings of the 16th AIAA Aviation Technology, Integration, and Operations Conference, Washington, DC, USA, 13–17 June 2016; p. 3910.
13. Lee, H.; Coupe, J.; Jung, Y.C. Prediction of pushback times and ramp taxi times for departures at charlotte airport. In Proceedings of the AIAA Aviation 2019 Forum, Dallas, TX, USA, 17–21 June 2019; p. 2933.
14. Herrema, F.; Curran, R.; Visser, H.; Huet, D.; Lacote, R. Taxi-out time prediction model at Charles de Gaulle Airport. *J. Aerosp. Inf. Syst.* **2018**, *15*, 120–130. [\[CrossRef\]](#)
15. Diana, T. Can machines learn how to forecast taxi-out time? A comparison of predictive models applied to the case of Seattle/Tacoma International Airport. *Transp. Res. Part E Logist. Transp. Rev.* **2018**, *119*, 149–164. [\[CrossRef\]](#)
16. Yin, J.; Hu, Y.; Ma, Y.; Xu, Y.; Han, K.; Chen, D. Machine learning techniques for taxi-out time prediction with a macroscopic network topology. In Proceedings of the 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), London, UK, 23–27 September 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–8.
17. Pham, D.; Ngo, M.; Tran, N.; Alam, S.; Duong, V. A data-driven approach for taxi-time prediction: A case study of Singapore changi airport. In *Air Traffic Management and Systems IV: Selected Papers of the 6th ENRI International Workshop on ATM/CNS (EIWAC2019) 6, Tokyo, Japan, 29–31 October 2019*; Springer: Singapore, 2021; pp. 113–129.
18. Wang, X.; Brownlee, A.E.; Woodward, J.R.; Weiszer, M.; Mahfouf, M.; Chen, J. Aircraft taxi time prediction: Feature importance and their implications. *Transp. Res. Part C Emerg. Technol.* **2021**, *124*, 102892. [\[CrossRef\]](#)
19. Li, N.; Jiao, Q.Y.; Zhang, L.; Wang, S.C. Using deep learning method to predict taxi time of aircraft: A case of Hong Kong airport. *J. Aeronaut. Astronaut. Aviat.* **2020**, *52*, 371–385.
20. Nan, L.; Qingyu, J.; Xinhua, Z.; Shaocong, W. Prediction of Departure Aircraft Taxi Time Based on Deep Learning. *Trans. Nanjing Univ. Aeronaut. Astronaut.* **2020**, *37*, 232–241.
21. Jiao, Q.Y.; Li, N. Taxi Time Prediction by Using Data Driven Approach: A New Perspective. Available online: <https://ssrn.com/abstract=4084964> (accessed on 24 March 2024). [\[CrossRef\]](#)
22. Lim, Y.; Tan, F.; Lilit, N.; Alam, S. Variable Taxi-Out Time Prediction Using Graph Neural Networks. In Proceedings of the 11th SESAR Innovation Days, Virtual Event, 7–9 December 2021; pp. 1–8.
23. Vargo, E.; Tien, A.; Jafari, A.; Hubbard, W. Airport Taxi Time Prediction and Alerting: A Convolutional Neural Network Approach. In Proceedings of the AIAA AVIATION 2022 Forum, Chicago, IL, USA, 27 June–1 July 2022; p. 4026.
24. Du, J.; Hu, M.; Zhang, W. Decision Support for Aircraft Taxi Time based on Deep Metric Learning. In Proceedings of the 23rd IEEE International Conference on Intelligent Transportation Systems, Rhodes, Greece, 20–23 September 2020.
25. Du, J.; Hu, M.; Zhang, W.; Yin, J. Finding Similar Historical Scenarios for Better Understanding Aircraft Taxi Time: A Deep Metric Learning Approach. *IEEE Intell. Transp. Syst. Mag.* **2022**, *15*, 101–116. [\[CrossRef\]](#)
26. Wang, F.; Bi, J.; Xie, D.; Zhao, X. A data-driven prediction model for aircraft taxi time by considering time series about gate and real-time factors. *Transp. A: Transp. Sci.* **2023**, *19*, 2071353. [\[CrossRef\]](#)
27. Hao, J.; Zhao, T.; Li, J.; Dong, X.L.; Faloutsos, C.; Sun, Y.; Wang, W. P-companion: A principled framework for diversified complementary product recommendation. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual, 19–23 October 2020; pp. 2517–2524.
28. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Jiang, W.; Luo, J. Graph neural network for traffic forecasting: A survey. *Expert Syst. Appl.* **2022**, *207*, 117921. [\[CrossRef\]](#)
30. Wang, Q.; Xu, C.; Zhang, W.; Li, J. GraphTTE: Travel time estimation based on attention-spatiotemporal graphs. *IEEE Signal Process. Lett.* **2021**, *28*, 239–243. [\[CrossRef\]](#)
31. Fang, X.; Huang, J.; Wang, F.; Zeng, L.; Liang, H.; Wang, H. Constgat: Contextual spatial-temporal graph attention network for travel time estimation at baidu maps. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual, 23–27 August 2020; pp. 2697–2705.

32. Fu, K.; Meng, F.; Ye, J.; Wang, Z. CompactETA: A fast inference system for travel time prediction. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual, 23–27 August 2020; pp. 3337–3345.
33. Derrow-Pinion, A.; She, J.; Wong, D.; Lange, O.; Hester, T.; Perez, L.; Nunkesser, M.; Lee, S.; Guo, X.; Wiltshire, B.; et al. Eta prediction with graph neural networks in google maps. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Online, 1–5 November 2021; pp. 3767–3776.
34. Jin, G.; Wang, M.; Zhang, J.; Sha, H.; Huang, J. STGNN-TTE: Travel time estimation via spatial–temporal graph neural network. *Future Gener. Comput. Syst.* **2022**, *126*, 70–81. [[CrossRef](#)]
35. Zou, G.; Lai, Z.; Ma, C.; Tu, M.; Fan, J.; Li, Y. When Will We Arrive? A Novel Multi-Task Spatio-Temporal Attention Network Based on Individual Preference for Estimating Travel Time. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 11438–11452. [[CrossRef](#)]
36. Rong, Y.; Yao, J.; Liu, J.; Fang, Y.; Luo, W.; Liu, H.; Ma, J.; Dan, Z.; Lin, J.; Wu, Z.; et al. GBTTE: Graph Attention Network Based Bus Travel Time Estimation. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, Birmingham, UK, 21–25 October 2023; pp. 4794–4800.
37. Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; Long, M. itransformer: Inverted transformers are effective for time series forecasting. *arXiv* **2023**, arXiv:2310.06625.
38. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6000–6010.
39. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *Stat* **2017**, *1050*, 10–48550.
40. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2016; pp. 770–778.
41. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; PMLR: Westminster, UK, 2015; pp. 448–456.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.