

Research Article

Distributed Typhoon Track Prediction Based on Complex Features and Multitask Learning

Yongjiao Sun,¹ Yaning Song ,¹ Baiyou Qiao,¹ and Boyang Li²

¹School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

²School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China

Correspondence should be addressed to Yaning Song; 1901780@stu.neu.edu.cn

Received 30 April 2021; Accepted 4 July 2021; Published 13 July 2021

Academic Editor: Guanfeng Liu

Copyright © 2021 Yongjiao Sun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Typhoons are common natural phenomena that often have disastrous aftermaths, particularly in coastal areas. Consequently, typhoon track prediction has always been an important research topic. It chiefly involves predicting the movement of a typhoon according to its history. However, the formation and movement of typhoons is a complex process, which in turn makes accurate prediction more complicated; the potential location of typhoons is related to both historical and future factors. Existing works do not fully consider these factors; thus, there is significant room for improving the accuracy of predictions. To this end, we presented a novel typhoon track prediction framework comprising complex historical features—climatic, geographical, and physical features—as well as a deep-learning network based on multitask learning. We implemented the framework in a distributed system, thereby improving the training efficiency of the network. We verified the efficiency of the proposed framework on real datasets.

1. Introduction

Typhoons are tropical cyclones that occur in the Western Pacific and adjacent waters and are common climate phenomena. Given that typhoons have significant destructive power and often imperil the coastal areas where they make landfall, the nature of these typhoons has long been an important research topic [1–3].

Typhoon track prediction is a typical problem in typhoon research. Traditionally, typhoon paths are often predicted through such methods as force analysis and mathematical statistics [4–7]. In recent years, however, with the development of artificial intelligence, more researchers are using deep-learning technology to predict the movement of typhoons. For example, some studies have utilized cloud maps to locate typhoons and predict their movement via convolutional neural networks (CNNs) and generative adversarial networks (GANs) [8,9]. Given that typhoon track is a continuous process, many studies also use recurrent neural networks (RNNs) and long short-term memory (LSTMs) to process the track sequence [10]. The formation and movement of typhoons is a very complex process that is

affected by historical as well as future factors. Although this problem has been widely studied, some limitations remain and hinder the accurate prediction of the paths typhoons take.

Typhoons have complex historical features. Existing studies have evaluated the history of typhoons with respect to geopotential height, wind field, and atmospheric pressure; however, these studies did not comprehensively analyse the features of previous typhoons. Therefore, by analysing historical data, we identified additional pertinent features and categorized them into climatic, geographical, and physical features. Further, we considered some new features—such as geostrophic force—for the purposes of this study. The factors that affect typhoon movement from many aspects were categorized under multimodal features.

Although existing works apply deep learning to evaluate typhoons, most only consider the track of a typhoon as an isolated target and ignore the multiple factors that influence this track. Likewise, although a few studies have predicted typhoon tracks via a multifaceted approach, their analyses of typhoon features are too simplistic. Therefore, we combined

the complex features of typhoons, processed the features through different learning frameworks, and incorporated multitask learning to further improve the accuracy of typhoon track prediction.

However, the expansion of data and model parameters is accompanied by an increase in computational power and duration of model training. In this regard, using distributed and parallel training methods such as SparkMLlib (<http://spark.apache.org/mllib/>) can significantly improve the efficiency of model training. Therefore, to improve the training efficiency of the framework proposed in this paper, we implemented it based on Ray (<https://ray.io/>), which is an emerging distributed AI platform.

The contributions of this paper are as follows:

- (1) We propose a typhoon track prediction framework that considers both historical features and the interaction of multiple factors.
- (2) We extracted the complex features—climatic, geographical, and physical—that affect the movement of typhoons. We employed deep-learning networks and a multitask learning method to improve the accuracy of typhoon track prediction.
- (3) We utilized distributed implementation to improve the training efficiency of the network.
- (4) We used real-life datasets to conduct the experiments and verify the effectiveness of the proposed framework.

The remainder of this paper is organized as follows: Section 2 introduces related works on typhoon track prediction. Section 3 covers the problem definition and related technologies. Section 4 introduces the proposed track prediction framework, including feature selection and network structure. We then verify the efficiency of the proposed framework through experiments in Section 5 and finally summarize this paper in Section 6.

2. Related Works

2.1. Traditional Methods. Traditional methods of typhoon track prediction include numerical, statistical, regression, and integrated models. Weber [7] proposed a numerical model (STEPS) to analyse the annual performance of the numerical orbit-prediction model; the model involves a very complex atmospheric-dynamics formula and requires strong computational power to successfully predict a typhoon's path. Demaria et al. [4] proposed a statistical model (SHIPS) that modifies the predictor according to the new prediction factors of every new year to make the model more suitable for observing typhoon movement. Compared with STEPS, SHIPS has a lower computational complexity; nonetheless, its accuracy is also relatively low. Goerss and Krishnamurti et al. [5,6] demonstrated that the integrated model comprising multiple models was more accurate as opposed to individual models. Although traditional models play a crucial role in forecasting typhoon tracks, they still have many shortcomings. With the increase in meteorological detection instruments, more meteorological

spatiotemporal data (big data) will be produced. However, traditional models are inevitably becoming outmoded. It is difficult for them to capture nonlinear typhoon models from these huge datasets, which significantly reduce the accuracy of prediction.

2.2. Deep-Learning Methods. In recent years, deep learning and parameter optimization [11] have rapidly developed and provided more powerful methods for typhoon track prediction. Neural networks have the advantages of nonlinearity and nonlocality. They can utilize big data to train the network and hence determine the mapping relationships between input and output; this essentially makes the predictions more accurate.

CNN-based methods: Wang et al. [9] used 2250 infrared satellite images to train the CNN network. The average angular error of typhoon track prediction was thus reduced to 27.8 degrees, indicating the great potential of CNN in typhoon path prediction. Giffard-Roisin et al. [12] proposed a fusion neural network comprising a neural network using past trajectory data and a CNN involving the reanalysis of atmospheric wind-field images.

GAN-based methods: Rüttgers et al. [8] used GAN in conjunction with satellite images and meteorological data to forecast the central location of typhoons. It has been proven that GAN utilizes many features that otherwise cannot be used by traditional models, thus preventing the otherwise inevitable errors associated with some traditional models.

RNN- and LSTM-based methods: Moradi Kordmahalleh et al. [13] used sparse RNNs with flexible topology in which a genetic algorithm (GA) was used to optimize the weight connection. Alemany et al. [14] proposed a fully connected RNN in the grid system; the proposed approach can be used to model the complex and nonlinear temporal behavior of typhoons. Further, it can accumulate the historical information of the nonlinear dynamics of the atmospheric system by updating the weight matrix, hence improving the accuracy of typhoon track prediction. Chandra and Dayal and Chandra et al. [15,16] also proved that RNNs are suitable for typhoon track prediction. Lian et al. [17] proposed a novel data-driven deep-learning model composed of a multidimensional feature-selection layer, a convolution layer, and a gating-cycle unit layer. It uses spatial locations and a variety of meteorological features to predict typhoon trajectories. Compared with CNNs and RNNs without a feature-selection layer, the novel model has higher accuracy. Using records from 1949 to 2012 as the training data, Gao et al. [10] proposed a typhoon track prediction method based on LSTM; the research shows that the model can predict the typhoon track 6–24 hours in advance with better accuracy. Kim et al. [18] proposed a large number of temporal and spatial prediction models based on the ConvLSTM model.

Multitask learning-based methods: Chandra [19] proposed a coevolutionary multitask learning algorithm that combines the functions of modularization and multitask learning. This approach coordinates multitask learning, dynamic programming, and coevolution algorithms. Furthermore, it can train neural networks via feature sharing and modular knowledge representation. It can also be used to predict typhoon intensity, with limited input [20]. This shows that, compared with traditional models, the algorithm not only solves the problem of dynamic time series but also improves the prediction accuracy. Mukherjee and Mitra [21] proposed a joint learning model that can learn the distance and direction of typhoons simultaneously via two different structures with multiple LSTMs and multiple fully connected layers; initial layer parameters are shared according to past typhoon track data. The research results show that the model can predict direction and distance (i.e., displacement) simultaneously.

3. Preliminaries

In this section, we first introduce the relevant technologies utilized in our framework and then proceed to define our problem.

3.1. CNN and ResNet. CNN is a type of deep-learning model that has been successfully implemented in image recognition [22]. The convolution layer is one of the core structures of CNNs. The input of the convolution layer includes one or more matrices of the same size, each of which is called a channel. Each convolution layer uses common parameters known as convolution kernels. For 2D input, the function of the convolution layer is to weigh the corresponding submatrices according to the size of kernels; thus, the convolution layer output is generated.

Another important structure is the pooling layer, which aims to reduce the parameters of the model and strengthen the network while improving the computing speed. The strategies of the pooling layer include the maximum and average pooling.

ResNet is a CNN model widely used for feature extraction [23]. To solve the migration problem in deep networks, ResNet proposes residual learning. ResNet replaces the feature $H(x)$ obtained by convolution layers with the residual $H(x) - x$ of feature and input. In contrast with ordinary CNN, ResNet adds a shortcut mechanism between every two layers to realize residual learning.

3.2. LSTM. LSTM is a special type of RNN that is deliberately designed to avoid long-term dependence. It introduces a gate to solve gradient disappearance or explosion [24]. LSTM contains four important structures, namely, the forget gate, the input gate, the update stage, and the output gate. As shown in Figure 1, this framework operates as follows:

- (1) The function of the forget gate is implemented by sigmoid to determine which information needs to be forgotten according to the input x_t and the output h_{t-1} of the previous cell.
- (2) The input gate determines the information that will be stored in the current cell.
- (3) The update stage updates C_t of the current cell.
- (4) The output gate outputs the final information to the next cell.

3.3. Multitask Learning. In single-task learning (involved in the previous models), the model learns only one task at a time. For complex problems, single-task learning decomposes the problems into multiple independent subproblems for separate training and then combines them. However, in practical applications, these subproblems frequently contain correlation information that is often ignored by the single-task learning method.

In this regard, the goal of multitask learning is to integrate multiple related tasks through shared representations [25]. It entails hard and soft parameter sharing. Hard parameter sharing shares some parameters among all tasks and only uses the tasks' unique parameters at a specific layer. In soft parameter sharing, each task has unique parameters. Finally, the similarity is expressed by adding constraints to the differences between parameters of different tasks.

3.4. Problem Definition. The problem of typhoon track prediction can be expressed in terms of the features of a given typhoon at several past instances or moments; the goal is to predict the locations at certain times or instances in the future. The past-feature sequence of the typhoon is denoted as $S(s_1, s_2, \dots, s_t)$, where s_i represents the features of the typhoon at time i and t is the length of the sequence. The track of the typhoon at the future moment is $T[(x_{t+1}, y_{t+1}), (x_{t+2}, y_{t+2}), \dots, (x_{t+n}, y_{t+n})]$, where (x_j, y_j) is the geographical coordinate (latitude and longitude) at time t_j . The goal of this study is to establish the mapping model $M: T \rightarrow S$ and hence calculate the future trajectory sequence T through the historical sequence S .

4. Framework

Figure 2 illustrates the structure of our proposed framework. It entails feature selection, weighted fusion, and multitask prediction. In this section, we will introduce all the parts individually.

4.1. Features. There are three types of features in our framework, namely, climatic, geographical, and characteristic features. The climatic features include sea surface temperature, geopotential height, and specific humidity. Geographical features include geostrophic forces. The characteristic features are the speed and position of the typhoon.

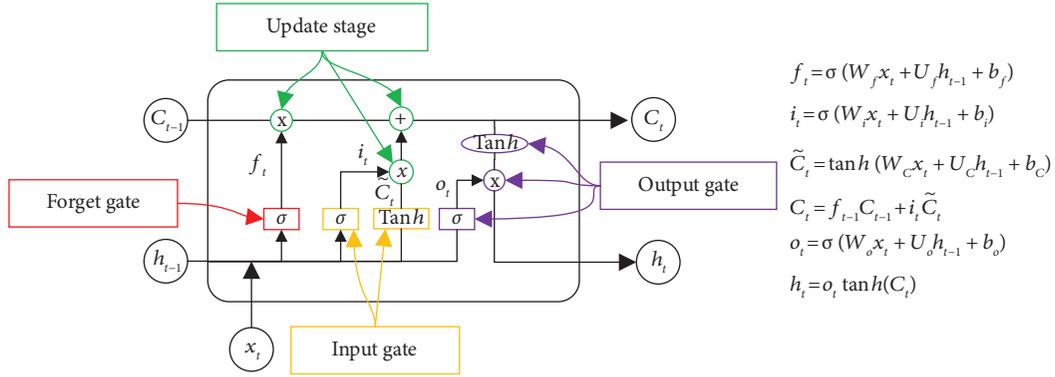


FIGURE 1: Structure of LSTM.

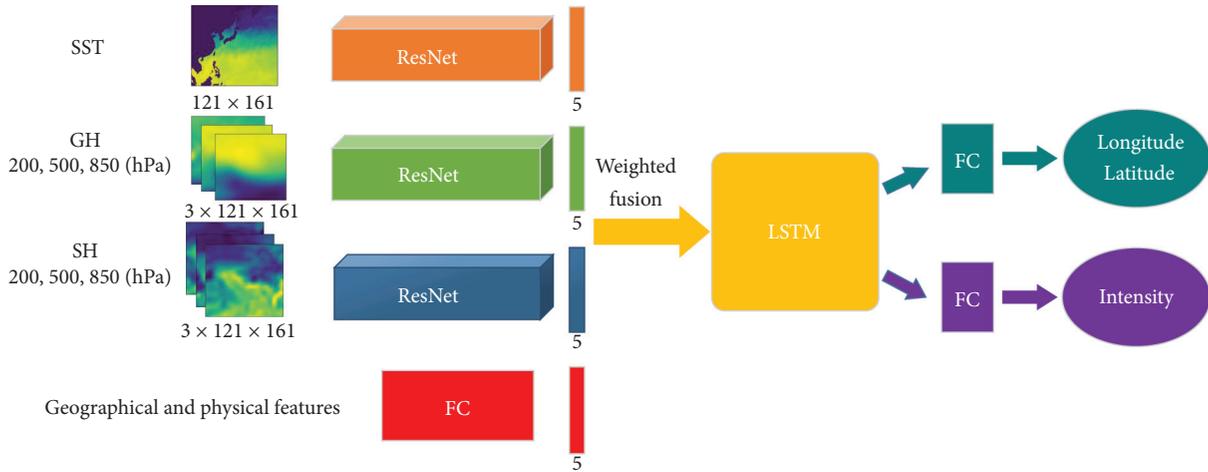


FIGURE 2: The structure of our framework.

4.1.1. Climatic Features. By studying the influence of climate on typhoons, we selected three main factors as climatic features in this study.

Sea surface temperature (SST): SST is one of the most important factors in meteorological research. In general, SST decreases when latitude increases. SST plays a pivotal role in the formation and movement of typhoons; typhoons are formed above the sea surface where SST is higher than 26.5°C and the intensity of the typhoon increases through continuous absorption of energy. SST is also one of the main factors influencing the direction of motion and landing location of typhoons. In this study, we mainly considered the region within 0°N and 60°N latitudes and 100°E and 180°E longitudes. The SST in this area was regularly collected by the sensor. As shown in Figure 3, we used a matrix of 121 rows and 161 columns to represent SST, in which the SST near the equator is above 30°C, whereas the SST at higher latitudes is approximately 0°C. We also distinguished land from sea; the darkest shades in Figure 3 are land.

Geopotential height (GH): GH is an imaginary height in meteorology, expressed in terms of the work done against gravity by an object of unit mass rising from sea

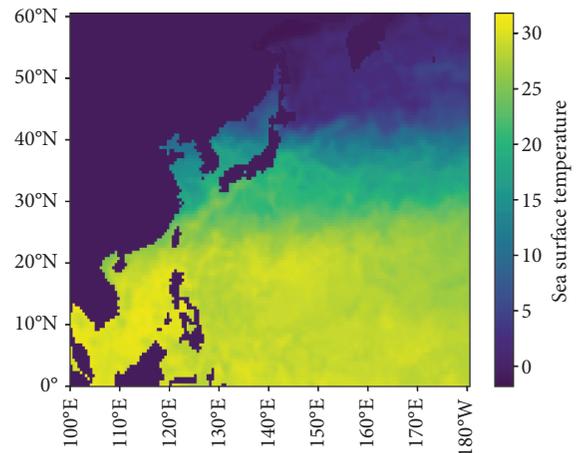


FIGURE 3: An example of SST.

level to a certain height. GH also plays an important role in maintaining the intensity and motion of typhoons. For example, the large geopotential height gradient between the Western Pacific subtropical high and typhoon determines the direction of movement of typhoon *Ambi* to a certain extent [13]. We studied GH in the same region described previously. In contrast to

SST, we choose three different GHs under different hPa. Figures 4(a)–4(c) show examples of GH, which is also represented by matrices with 121 rows and 161 columns. It is evident from these charts that GH increases with latitude.

Specific humidity (SH): SH refers to the ratio of the mass of water vapor in the atmosphere to the total mass of air. There is a strong relationship between typhoons and vertical air motion, and SH is usually used when discussing the vertical motion. Therefore, we introduced SH as a distinct climatic feature. Figures 4(d)–4(f) show 3 SH data charts under different hPa. We can observe that SH in the south is higher than that in the north.

4.1.2. Geographical Feature. (1) Geostrophic force (GF): GF, also known as the Coriolis force, was derived to describe the force exerted on moving objects on the surface of the Earth as a result of the Earth’s rotation. Owing to the existence of GF, a rotating flow of air is formed, and eventually, a typhoon is formed under the combined action of various factors. The typhoon is also affected by GF during its movement. In the northern hemisphere, the GF of the typhoon is to the right, which determines the typhoon’s direction of movement to a certain extent. GF can be expressed as

$$F = 2mv\omega \sin \theta, \quad (1)$$

where m is the mass of the object, v is the velocity of the object, ω is the angular velocity of the Earth’s rotation, and θ is the latitude of the object before it begins to move. Given that the mass of typhoons is difficult to estimate, we use the geostrophic force gradient to represent the influence of GF on typhoons, denoted as

$$\frac{\partial F}{\partial m} = 2v\omega \sin \theta. \quad (2)$$

4.1.3. Physical Features. We use physical characteristics to describe the time series and tracks of typhoons.

Location and direction: Given that the track of a typhoon is a series of coordinates, we used the latitudes and longitudes (lat, lon) or offsets (Δlat , Δlon) to describe the location and direction of motion of typhoons. Since typhoon data were collected every 6 hours, we calculated the movement and direction of the typhoon every 6 hours.

Speed: The typhoon data are coarse. Therefore, we used the average of the velocities of the typhoon at two consecutive moments to describe the moving velocity of the typhoon.

Intensity: The intensity of a typhoon is determined by its wind speed. Existing studies have validated the relationship between the central pressure of a typhoon and the maximum wind speed [26]. Therefore, we used the maximum central pressure to express the intensity characteristics of a typhoon.

4.2. Network. Owing to the different modes of features, we used different networks to process the features and then used feature fusion for learning. The entire network architecture is illustrated in Figure 2.

4.2.1. Feature Extraction. We used climatic, geographical, and physical features. Some of these were two-dimensional matrices, whereas some were one-dimensional vectors. Consequently, we used different networks for different features.

For climatic features, all inputs were two-dimensional images. We therefore used three ResNets to process the images. The ResNets employed in our framework have 18 hidden layers [23] as shown in Figure 5. GH and SH have trichannel inputs, whereas SST has single-channel input. The first layer is a convolution layer. The size of the convolution kernel is 7×7 and the stride is (2, 2). Based on the size of the input, we set padding as (3, 3). Batch normalization (BN) and rectified linear units (ReLU) were also used in the convolution layer. After the convolution operation, the network performs a maximum-pooling operation. There are four residual blocks after the first layer. Each residual block is repeated twice. To simplify the representation, the repeated parts have been replaced by ellipses. Each residual block contains two convolution layers. Each layer contains a convolution kernel, batch normalization, and ReLU. The size of the convolution kernel is 3×3 , the stride is (1, 1), and the padding is (1, 1). The output dimensions of each residual block are 64, 128, 256, and 512. After the last residual block, the network performs an average-pooling operation. The last layer of the network is a fully connected network with 5-dimensional output.

As for the geographical and physical features, we used a fully connected network and obtained a 5-dimensional vector as the output. For feature fusion, we adopted a weight module. The weight of each feature can be regarded as the correlation between the feature and track of the typhoon. Through weighted feature fusion, for each moment, we obtained a 20-dimensional feature vector, which then became the input of the predictor.

4.2.2. Multitask Prediction. Because LSTM has a considerable advantage in the processing of sequence data, we used the classic LSTM as the predictor. The dimension of the input was $t \times 20$, where t is the length of the sequence, as introduced in Section 2. The training process is shown in Figure 6. First, we used zero-state initialization to calibrate the weight, h_0 , and C_0 . For each cell of the LSTM, the input is the i -th 20-dimensional feature vector. It should be noted that all LSTM cells share these parameters.

The LSTM output is divided into two tasks. The main task involves locating the typhoon at the next moment, and the auxiliary task involves determining the central pressure of the typhoon (i.e., the intensity of the typhoon). We used the L_2 norm as the loss function of the two tasks.

For the main task, the loss is the difference in distance between the real location and the predicted location of the typhoon, as follows:

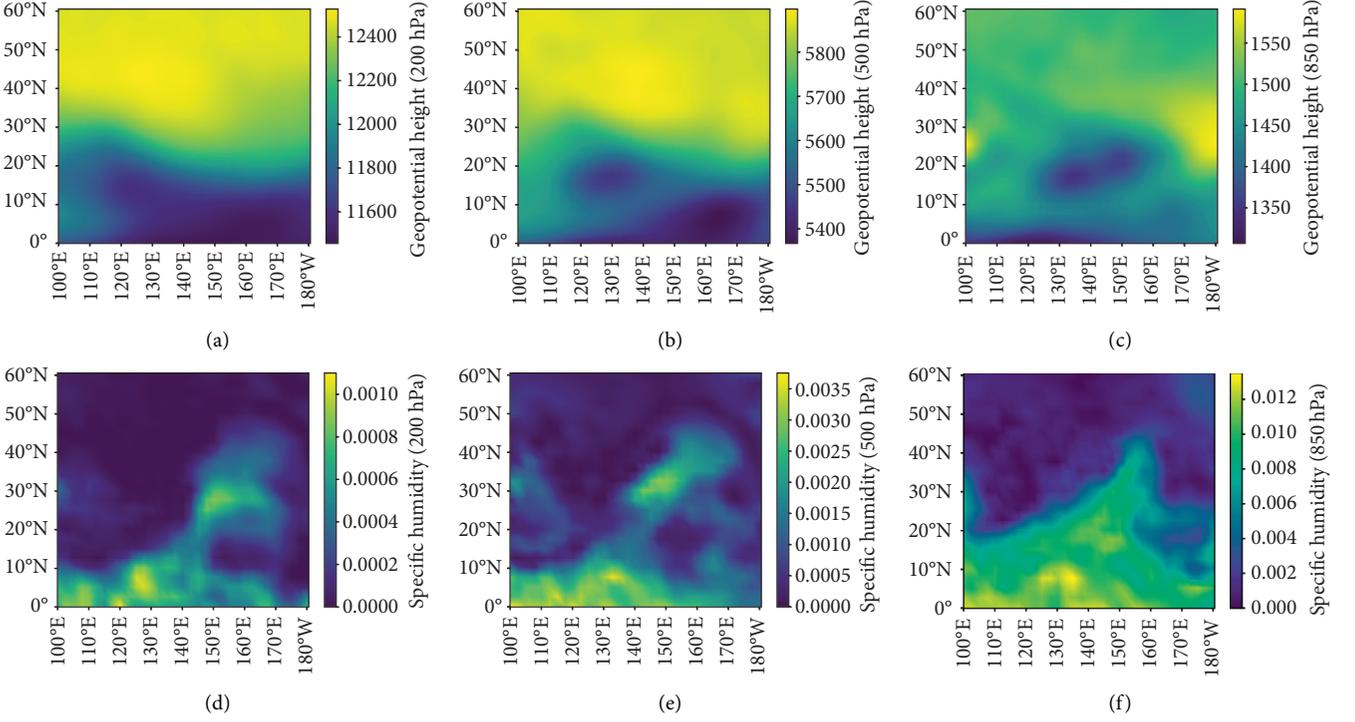


FIGURE 4: Example of GH and SH. (a) GH at 200 hPa. (b) GH at 500 hPa (c) GH at 850 hPa. (d) SH at 200 hPa. (e) SH at 500 hPa. (f) SH at 850 hPa.

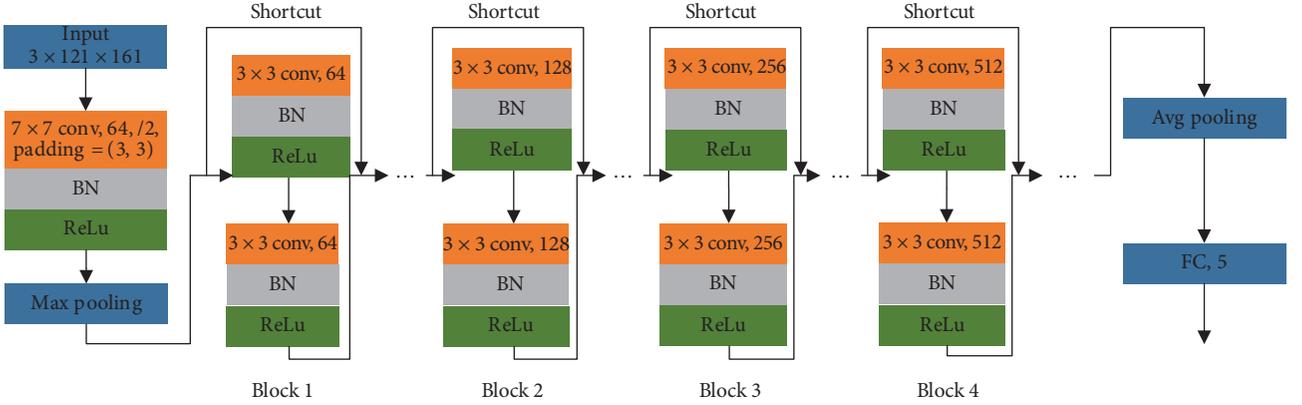


FIGURE 5: Details of ResNet in our framework.

$$L_{\text{main}} = \sqrt{(x - \hat{x})^2 + (y - \hat{y})^2}, \quad (3)$$

where (x, y) is the location of the typhoon at the next moment and (\hat{x}, \hat{y}) is the output of the predictor. The longitude and latitude offset can also be used as input, and the corresponding loss will become the difference in the offset. For the auxiliary task, the loss function is denoted as

$$L_{\text{auxiliary}} = \sqrt{(p - \hat{p})^2}, \quad (4)$$

where p is the central pressure of the typhoon and \hat{p} is the prediction result.

Therefore, the total loss of our framework is as follows:

$$L_{\text{total}} = \alpha L_{\text{main}} + (1 - \alpha) L_{\text{auxiliary}}. \quad (5)$$

In this loss function, α is a hyperparameter.

4.3. Distributed Implementation. To ensure that the proposed framework can handle big data and consequently improve the efficiency of training, we implemented a distributed framework based on Ray. Ray is a very popular distributed AI platform implemented via Python. This facilitates the rapidly distributed computing of the Python

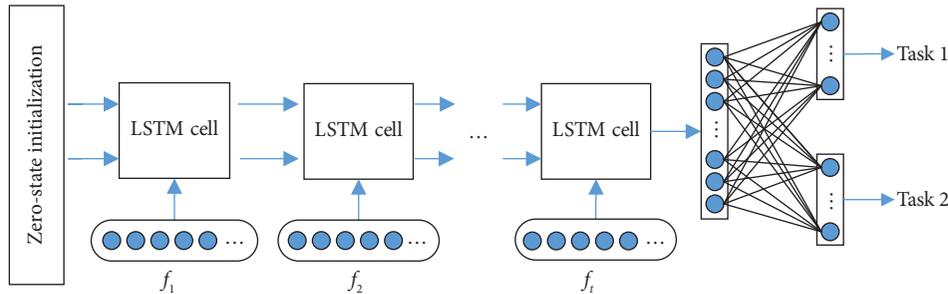


FIGURE 6: The details of multitask prediction.

code. In the implementation, each network structure (such as convolution layer, pooling layer, and FC layer) is implemented as a class, also known as an actor in Ray. Multiple actors construct the entire network through the data flow. In the calculation process, each calculation node starts multiple workers as the basis of calculation. Each actor is assigned to the corresponding worker for execution. In the training process, the data flows through gRPC and shared memory to the corresponding worker for calculation. For example, in each ResNet, after the calculation of the current layer is completed, the data will flow to the worker of the next layer. There is no data dependence between multiple ResNets; therefore, parallel training can be realized.

5. Experiments

5.1. Setup. We use a real dataset to verify the effectiveness of our framework. The dataset is the Western Pacific Typhoon track data from the JTWC (be Typhoon Warning Center, the Joint Typhoon Warning Center). The dataset contains typhoon tracks from January 1, 2001, to December 31, 2005. The attitude is from 0°N to 60°N and the longitude is from 100°E to 180°E . Statistics of the experimental setup are shown in Table 1.

We use the metric of distance error (same as L_{main}) to verify the effectiveness of our framework. We first verify the benefits of multitask learning technology to this framework. Next, we use different weights to discuss the relationship between features and results. The framework is implemented by Python 3, and the experiments are conducted on a cluster in which each node has Intel Purley 4110 CPUs and Tesla P100 GPUs.

5.2. Results. In this section, we will introduce the experimental results in the real-life dataset. We report and analyse the results by changing the parameters. Then, we choose some real typhoon tracks to show our prediction results.

Distance error with respect to multitask and single-task learning: firstly, we compare the results of multitask learning (MTL) and single-task learning (STL), as shown in Figure 7. We can observe that MTL can get better results than STL in most cases. In the 6 h prediction results, MTL is similar to STL. However, in other cases, MTL can achieve about 20% performance improvement. It proves that it is feasible to improve the

effect of track prediction by auxiliary tasks. What is more, the best results in 6 h, 24 h, 48 h, and 72 h are about 40 km, 70 km, 220 km and 380 km which are better than most existing models. It also proves the effectiveness of our framework.

Distance error with respect to $|T||T|$: secondly, we report the distance error with different size of input $|T|$. The results are also shown in Figure 7. We find that $|T|$ has a great influence on our framework in different cases. The optimal value is 3, 7, 4, and 5 in 6 h, 24 h, 48 h, and 72 h. As $|T|$ becomes larger or smaller, the distance error gradually increases. In the later experiments, we selected the best value of $|T|$ in each case to verify the effect of feature weight on the distance error.

Then, we study the relationship between features and prediction results.

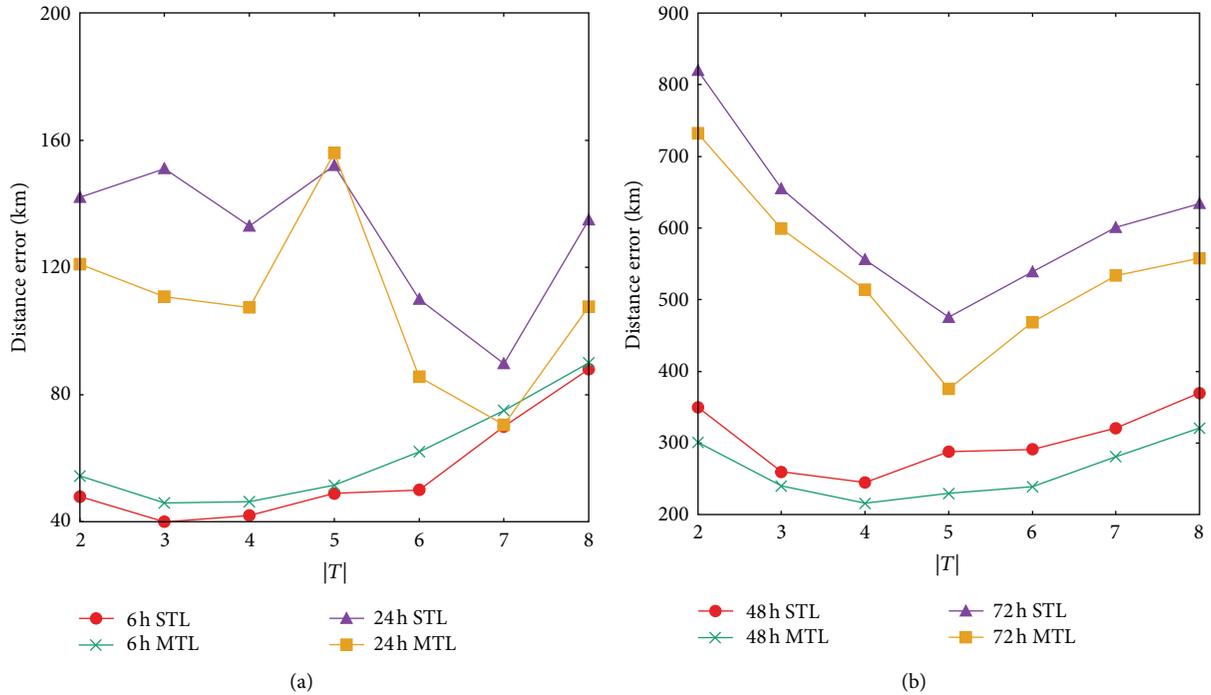
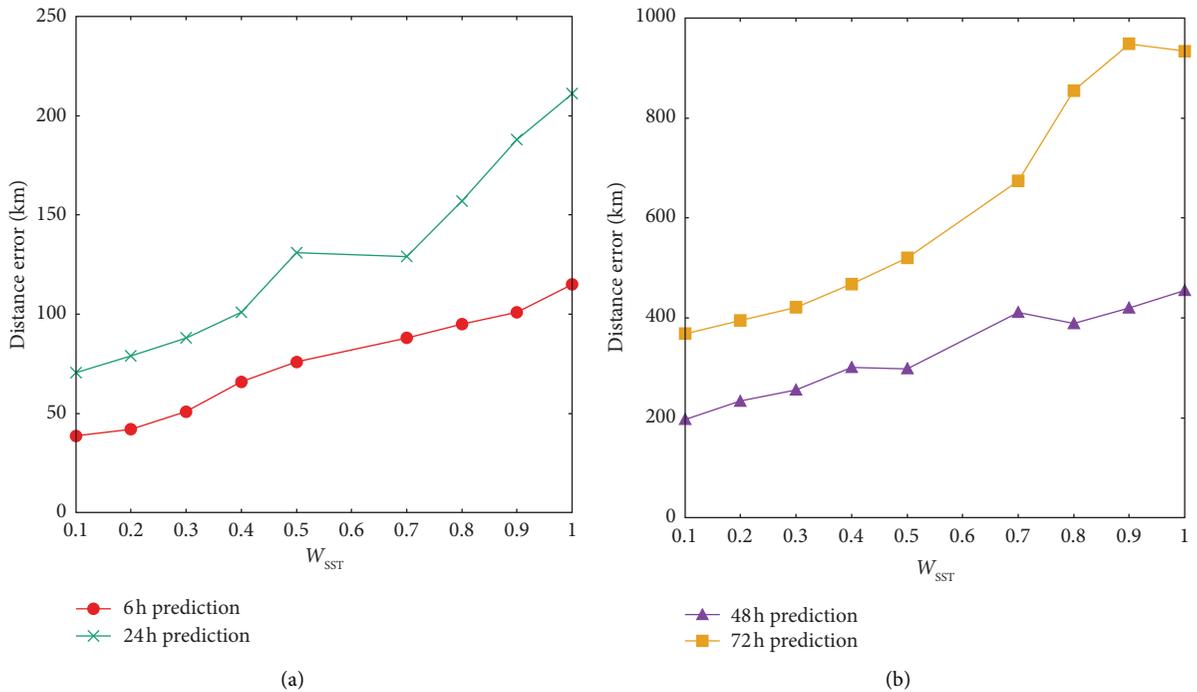
Distance error with respect to $w_{\text{SST}}w_{\text{SST}}$: to study the effect of SST, we keep w_{GH} and w_{SH} unchanged and then adjust the value of w_{SST} from 0.1 to 1.0. The results are shown in Figure 8. We can observe that SST will greatly affect the results. The best choice is to reduce w_{SST} as small as possible.

Distance error with respect to $w_{\text{GH}}w_{\text{GH}}$: to study the relationship between GH and prediction results, we keep w_{SST} and w_{SH} unchanged and then adjust the value of w_{GH} from 0.1 to 1.0. As shown in Figure 9, we can get the best results when w_{GH} is set as 0.8. The difference between the best result and the worst result in 6 h, 24 h, and 48 h is about 30 km to 100 km. In 72 h, the difference could be more than 300 km. An appropriate w_{GH} can improve the results by 30% to 40%. The experimental results show that there is a strong correlation between GH and prediction results.

Distance error with respect to $w_{\text{SH}}w_{\text{SH}}$: to study the relationship between SH and prediction results, we keep w_{SST} and w_{GH} unchanged and then adjust the value of w_{SH} from 0.1 to 1.0. The results are shown in Figure 10. To get better results, w_{SH} is smaller than w_{GH} . In 6 h and 24 h cases, we can get the best results when w_{SH} is set as 0.1. In 48 h and 72 h cases, it is better to set w_{SH} as 0.3. An appropriate w_{SH} can improve the result by 40% to 50%. The experimental results show that SH is also related to the prediction results, but the correlation is less than GH.

TABLE 1: Statistics of the experimental setup.

Region		Date range	Dimension of features		
Attitude 0°N to 60°N	Longitude 100°E to 180°E	January 1, 2001, to December 31, 2005	SST 121 × 161	GH/SH 3 × 121 × 161	Others 20 × 1

FIGURE 7: Results of varying $|T|$. (a) Results of 6h and 24h. (b) Results of 48h and 72h.FIGURE 8: Results of varying weight of w_{SST} . (a) Results of 6h and 24h. (b) Results of 48h and 72h.

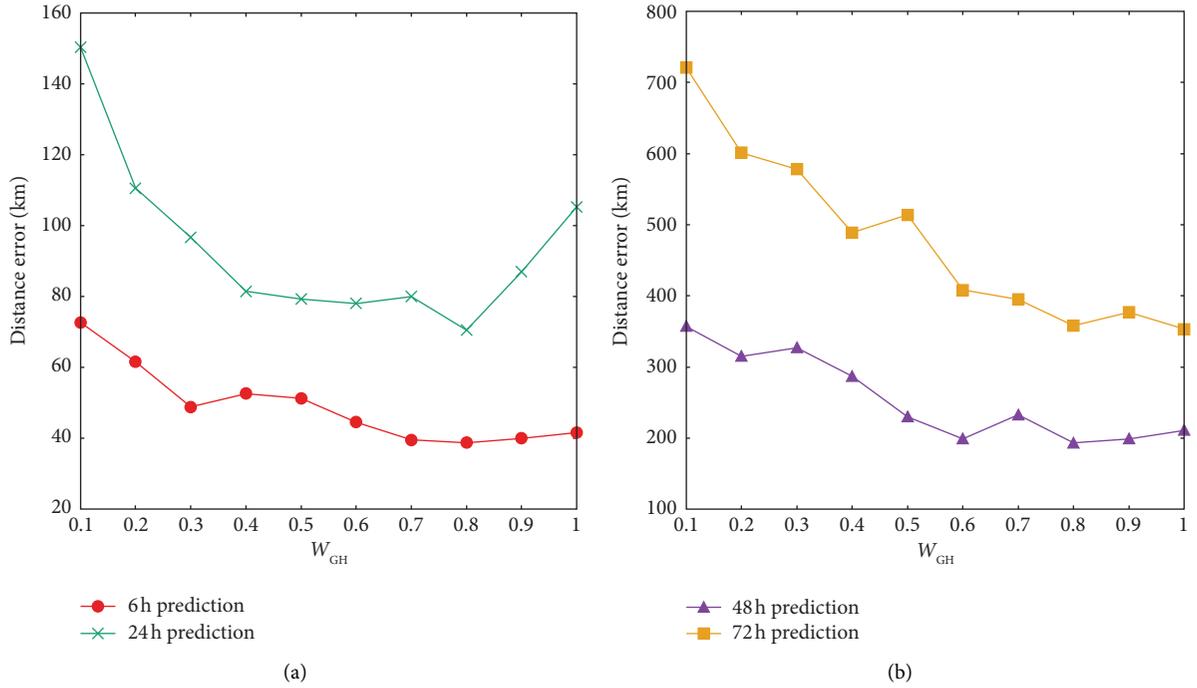


FIGURE 9: Results of varying weight of w_{GH} . (a) Results of 6 h and 24 h. (b) Results of 48 and 72 h.

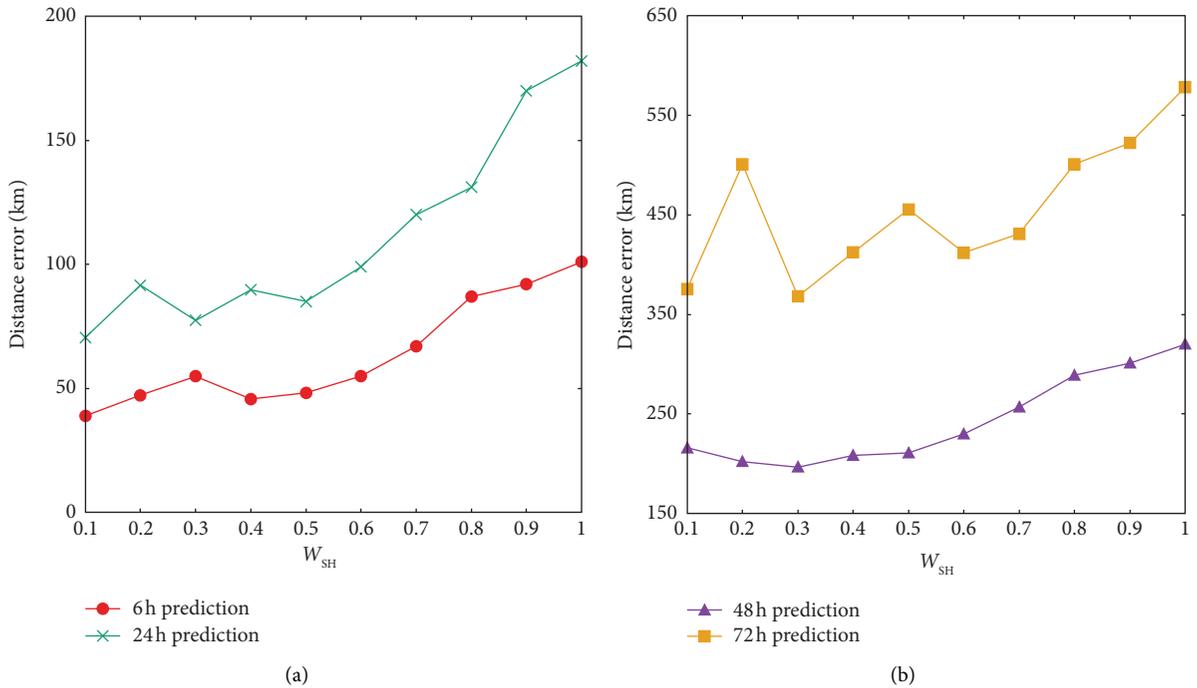


FIGURE 10: Results of varying weight of w_{SH} . (a) Results of 6 h and 24 h. (b) Results of 48 and 72 h.

Case study: We use some real typhoons to compare the real tracks and the prediction results. We select Saola, Damrey, and Longwang that are formed in 2005; the real tracks and 6 h prediction results are shown in Figures 11–13. Typhoon Saola was formed on September 20th; the average distance error of 6 h

prediction results is 40.33 km. Typhoon Damrey was formed on September 21, the average distance error is 40.59 km, the minimum error is 8.9 km, and the maximum error is 60.33 km. Typhoon Longwang was formed on September 26; the average distance error is 46.51 km.

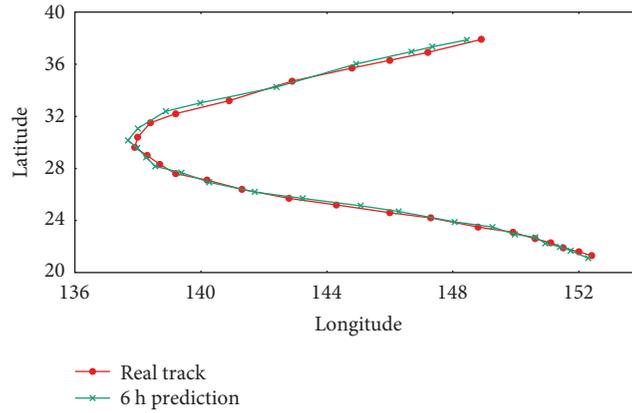


FIGURE 11: 6 h prediction results of Saola.

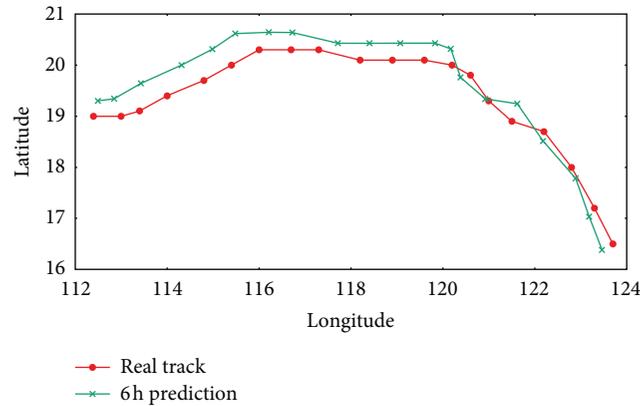


FIGURE 12: 6 h prediction results of Damrey.

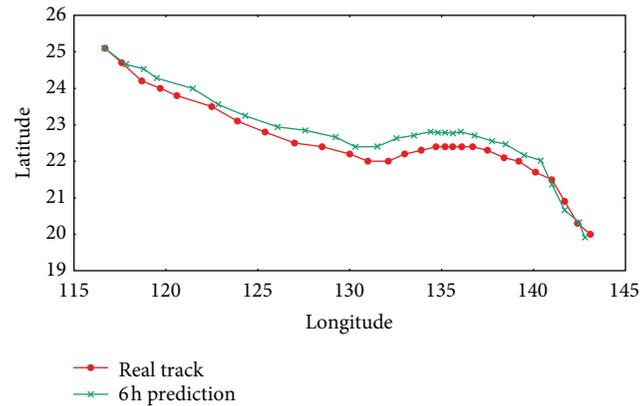


FIGURE 13: 6 h prediction results of Longwang.

Comparison with existing works: Finally, we compare our framework with several existing works [8,10,12,27]. According to the previous introduction, Rüttgers et al. [8] introduced a GAN-based model, used satellite images as the input, and predicted locations after 6 hours. Gao et al. [10] introduced an LSTM-based model. The work by Giffard-Roisin et al. [12] was based

on CNN and feature fusion. Lv et al. [27] used the least square method and FC network to predict the locations. We still use distance error to verify the effectiveness and the results are shown in Table 2. Compared with these works, our framework can achieve high prediction results, especially in 48 h and 72 h cases. In 72 h results, our framework improves the accuracy by 60%.

TABLE 2: Results compared with the existing works.

	6 h	24 h	48 h	72 h
Our framework	38.75	69.54	196.61	368.1
Gao et al. [10]	45.95	105.68	332.54	974.50
Giffard-Roisin et al. [12]	—	136.1	—	—
Rüttgers et al. [8]	95.6	—	—	—
Lv et al. [27]	—	158.34	361.76	—

5.3. *Summary.* In this section, we verify the effect of different parameters on the performance of our framework in the real dataset. In general, our framework can achieve good results based on multitask and feature weighting. We find that GH has a strong correlation with the movement of typhoons, followed by SH, and SST has the weakest correlation. Through the training results, the optimal prediction results can be obtained by selecting the appropriate parameters for different scenes.

6. Conclusion

In this paper, we proposed a typhoon track prediction framework based on multitask learning and feature weighting. We analysed the correlation between the climatic, geographical, and physical features and typhoon movement through the method of feature weighting. We designed a network based on ResNet and LSTM and used a multitask learning method to improve the prediction accuracy. We implemented the network in a distributed platform. Finally, we conducted experiments on real datasets to prove the effectiveness of the framework. In future works, we will analyse more features and use the attention mechanism to automatically process the weight of features.

Data Availability

The data are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Acknowledgments

The work was supported by the National Key R&D Program of China (Grant no. 2016YFC1401902), the National Natural Science Foundation of China (Grant no. 61972077), and the LiaoNing Revitalization Talents Program (Grant no. XLYC2007079).

References

- [1] W. Liu, K. Fujii, Y. Maruyama, and F. Yamazaki, "Inundation assessment of the 2019 typhoon hagibis in Japan using multi-temporal sentinel-1 intensity images," *Remote Sensing*, vol. 13, no. 4, p. 639, 2021.
- [2] J. Cai, Y. Zhang, R. J. Doviak, Y. Shrestha, and P. W. Chan, "Diagnosis and classification of typhoon-associated low-altitude turbulence using HKO-TDWR radar observations and machine learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 6, pp. 3633–3648, 2019.
- [3] J. Li, Q. Zheng, M. Li, Q. Li, and L. Xie, "Spatiotemporal distributions of ocean color elements in response to tropical cyclone: a case study of typhoon mangkhut (2018) past over the northern south China sea," *Remote Sensing*, vol. 13, no. 4, p. 687, 2021.
- [4] M. Demaria, M. Mainelli, L. K. Shay, J. A. Knaff, and J. Kaplan, "Further improvements to the statistical hurricane intensity prediction scheme (SHIPS)," *Weather and Forecasting*, vol. 20, no. 4, pp. 531–543, 2005.
- [5] J. S. Goerss, "Tropical cyclone track forecasts using an ensemble of dynamical models," *Monthly Weather Review*, vol. 128, no. 4, pp. 1187–1193, 2000.
- [6] T. N. Krishnamurti, C. M. Kishtawal, Z. Zhang et al., "Multimodel ensemble forecasts for weather and seasonal climate," *Journal of Climate*, vol. 13, no. 23, pp. 4196–4216, 2000.
- [7] H. C. Weber, "Hurricane track prediction using a statistical ensemble of numerical models," *Monthly Weather Review*, vol. 131, no. 5, pp. 749–770, 2003.
- [8] M. Rüttgers, S. Lee, S. Jeon, and D. You, "Prediction of a typhoon track using a generative adversarial network and satellite images," *Scientific Reports*, vol. 9, no. 1, p. 6057–6115, 2019.
- [9] C. Wang, Q. Xu, X. Li et al., "CNN-based tropical cyclone track forecasting from satellite infrared images," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, pp. 5811–5814, Waikoloa, HI, USA, September 2020.
- [10] S. Gao, P. Zhao, B. Pan et al., "A nowcasting model for the prediction of typhoon tracks based on a long short term memory neural network," *Acta Oceanologica Sinica*, vol. 37, no. 5, pp. 8–12, 2018.
- [11] J. Chen, M. Zhong, J. Li, D. Wang, T. Qian, and H. Tu, "Effective deep attributed network representation learning with topology adapted smoothing," *IEEE Transactions on Cybernetics*, 2021.
- [12] S. Giffard-Roisin, M. Yang, G. Charpiat, C. Kumler Bonfanti, B. Kégl, and C. Monteleoni, "Tropical cyclone track forecasting using fused deep learning from aligned reanalysis data," *Frontiers in Big Data*, vol. 3, p. 1, 2020.
- [13] M. Moradi Kordmahalleh, M. Gorji Sefidmazgi, and A. Homaifar, "A sparse recurrent neural network for trajectory prediction of atlantic hurricanes," in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 957–964, Lille, France, July 2016.
- [14] S. Alemay, J. Beltran, A. Perez et al., "Predicting hurricane trajectories using a recurrent neural network," in *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, pp. 468–475, Honolulu, HI, USA, January 2019.
- [15] R. Chandra and K. Dayal, "Cooperative neuro-evolution of Elman recurrent networks for tropical cyclone wind-intensity prediction in the south pacific region," in *Proceedings of the*

- IEEE Congress on Evolutionary Computation (CEC)*, pp. 1784–1791, Sendai, Japan, May 2015.
- [16] R. Chandra, K. Dayal, and N. Rollings, “Application of co-operative neuro-evolution of Elman recurrent networks for a two-dimensional cyclone track prediction for the South Pacific region,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Killarney, Ireland, July 2015.
- [17] J. Lian, P. Dong, Y. Zhang, J. Pan, and K. Liu, “A novel data-driven tropical cyclone track prediction model based on CNN and GRU with multi-dimensional feature selection,” *IEEE Access*, vol. 8, pp. 97114–97128, 2020.
- [18] S. Kim, H. Kim, J. Lee et al., “Deep-hurricane-tracker: tracking and forecasting extreme climate events,” in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 1761–1769, Waikoloa, HI, USA, January 2019.
- [19] R. Chandra, “Dynamic cyclone wind-intensity prediction using co-evolutionary multi-task learning,” in *Proceedings of the International Conference on Neural Information Processing*, pp. 618–627, Guangzhou, China, November 2017.
- [20] R. Chandra, Y.-S. Ong, and C.-K. Goh, “Co-evolutionary multi-task learning for dynamic time series prediction,” *Applied Soft Computing*, vol. 70, pp. 576–589, 2018.
- [21] A. Mukherjee and P. Mitra, “Joint learning for cyclone track nowcasting,” in *Proceedings of the ECML/PKDD, CEUR Workshop*, Ghent, Belgium, September 2020.
- [22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.
- [23] K. He, X. Zhang, S. Ren et al., “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [24] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [25] K.-H. Thung and C.-Y. Wee, “A brief review on multi-task learning,” *Multimedia Tools and Applications*, vol. 77, no. 22, pp. 29705–29725, 2018.
- [26] K. Chen, “Calculation of the maximum wind speed of typhoon in the western pacific,” *Marine Science Bulletin*, 1985.
- [27] Q. P. Lv, J. Luo, K. Zhu et al., “Experiments on predicting tracks of tropical cyclones based on artificial neural network,” *Guangdong Meteorology*, pp. 19–22, 2009.