MGTCF: Multi-Generator Tropical Cyclone Forecasting with Heterogeneous Meteorological Data

Cheng Huang[1], Cong Bai[1,2*], Sixian Chan[1,3], Jinglin Zhang[4], YuQuan Wu[5]

[1]College of Computer Science, Zhejiang University of Technology
[2]Key Laboratory of Visual Media Intelligent Processing Technology of Zhejiang Province
[3]KLME, CIC-FEMD, Nanjing University of Information Science & Technology
[4] School of Control Science and Engineering, Shangdong University
[5] Institute of Software Chinese Academy of Sciences
{chenghuang, congbai, sxchan}@zjut.edu.cn, jinglin.zhang@sdu.edu.cn, yuquan@iscas.ac.cn

Abstract

Accurate forecasting of tropical cyclone (TC) plays a critical role in the prevention and defense of TC disasters. We must explore a more accurate method for TC prediction. Deep learning methods are increasingly being implemented to make TC prediction more accurate. However, most existing methods lack a generic framework for adapting heterogeneous meteorological data and do not focus on the importance of the environment. Therefore, we propose a **M**ulti-**G**enerator **T**ropical **C**yclone **F**orecasting model (**MGTCF**), a generic, extensible, multi-modal TC prediction model with the key modules of Generator Chooser Network (GC-Net) and Environment Net (Env-Net). The proposed method can utilize heterogeneous meteorologic data efficiently and mine environmental factors. In addition, the Multi-generator with Generator Chooser Net is proposed to tackle the drawbacks of single-generator TC prediction methods: the prediction of undesired out-of-distribution samples and the problems stemming from insufficient learning ability. To prove the effectiveness of **MGTCF**, we conduct extensive experiments on the China Meteorological Administration Tropical Cyclone Best Track Dataset. **MGTCF** obtains better performance compared with other deep learning methods and outperforms the official prediction method of the China Central Meteorological Observatory in most indexes.

Introduction

Tropical cyclones (TCs) are powerful and complex weather systems, also known as typhoons, hurricanes, or cyclonic storms. They usually develop in tropical, subtropical, and temperate zones with substantial precipitation, which maintain a balance of global heat and momentum distribution (Emanuel 2018). However, strong TCs can prove lethal for vessels and platforms at sea. Furthermore, when a TC makes landfall, it can also bring about ensuing natural disasters, such as gales, storm surges, floods (Anthes 2016), which can cause significant economic losses and casualties. To defend against these disasters, we need to be able to predict the future tendencies of the trajectory and intensity of TC in advance. These are the most important aspects of TC prediction and are also the ones that we want to forecast in

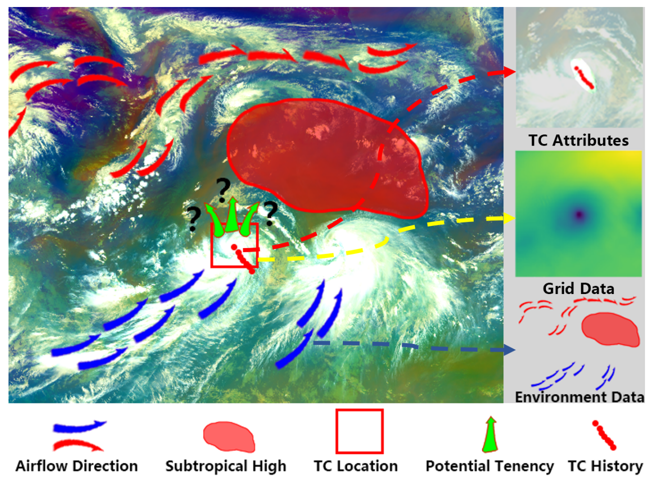


Figure 1: The development of TC with environment factors. The sub-figures in the left are the examples of heterogeneous meteorological data.

this work. However, it is very difficult to predict TCs, because there are many factors involved in the development of a TC, including the general circulation of atmosphere, the region of the subtropical high, the pressure structure and location of the TC, the sea surface temperature, and other environment-related factors, (Emanuel 2007), as shown in Figure 1. Researchers still do not understand how all these factors impact the TC. Because TC prediction is so critical, researchers began studying this topic in the previous century (Neumann and Lawrence 1975). They studied the principle of TC and tried various approaches to improve the accuracy of TC prediction, including empirical methods, statistical methods, and numerical methods. Based on these methods, other approaches have been developed in recent years (Wang et al. 2015; Chen and Zhang 2019).

Numerical Weather Prediction (NWP) systems have become the mainstream TC prediction method, used by many official meteorological forecasting agencies, including the China Central Meteorological Observatory (CMO). This method uses supercomputers to simulate various factors influencing the development of TCs. However, because the principle of TC development is complex, supercomputers

*Corresponding Author

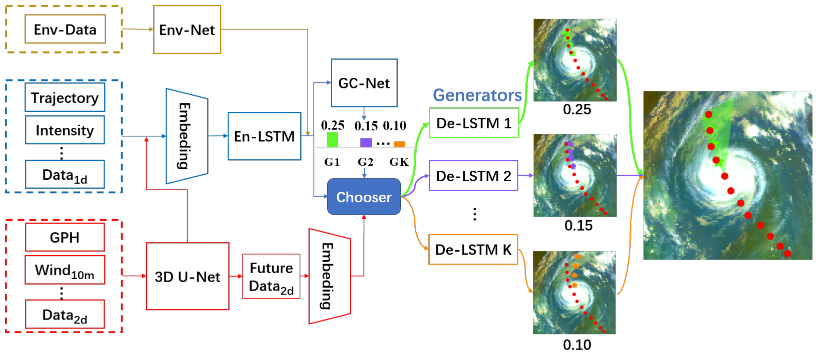


Figure 2: MGTCF framework. The golden branch is the environment data encoder. The blue branch is the inherent attributes data of TC ($Data_{1d}$) encoder. The red branch is the meteorological grid data ($Data_{2d}$) encoder. **Chooser** selects different generators by the probability array, the histogram below GC-Net, from Generator Chooser Net. All predictions from selected generators constitute the multiple potential tendencies of TC. Best viewed in color.

need to use massive amount of expensive computing resources to execute TC forecasting (Rüttgers et al. 2019). In comparison, deep learning methods use only several GPUs to execute difficult tasks. Deep learning is widely used in many domains, including computing vision (Duan et al. 2022), time series prediction (Fan et al. 2022), and other interdisciplinary disciplines (Kumar, Biswas, and Pandey 2021; Zhao et al. 2022). Due to the availability of rich heterogeneous meteorological data, deep learning technology, particularly the recurrent neural network (RNN) (Hochreiter and Schmidhuber 1997), can also be applied to the task of TC prediction. Therefore, some researchers have started to explore the application of deep learning technology to TC prediction.

Single-modal data is first used as the input of an RNN to predict TCs (Alemany et al. 2019; Pan, Xu, and Shi 2019). This is an interesting attempt to combine deep learning and TC prediction and promote the development of TC prediction methods based on deep learning. However, the features extracted from single-modal data are insufficient to fully represent the TC. Therefore, predictions using only single-modal data are not sufficiently precise. Then, heterogeneous meteorological data were used in various TC prediction methods (Giffard-Roisin et al. 2020; Liu et al. 2022). These data include the inherent attribute data of TC called $Data_{1d}$ (e.g., longitude, latitude, and wind) and meteorological grid data called $Data_{2d}$ (e.g., satellite images and meteorological fields). However, these methods used different data, and there are varying designs of the data encoder. Designing a new encoder for each set of meteorological data is inadvisable, owing to the large amounts of heterogeneous meteorological data. Therefore, it is necessary to build a generic encoder for all the different modals and dimensions of data.

Most meteorologists currently prefer methods employing a mechanism for predicting multiple potential TC tendencies, as shown in the semitransparent green region shown at the left in Figure 2. This mechanism, which was not a part of most previous methods, can provide meteorologists with more referential and reasonable forecasting. This mechanism in TC prediction is usually implemented using a single-generator generative adversarial network (GAN) with different noises (Huang et al. 2022). However, this type of method has been shown to have some drawbacks, including the prediction of undesired out-of-distribution (OOD) samples and insufficient learning ability (Dendorfer, Elflein, and Leal-Taixé 2021). These drawbacks need to be resolved in the task of TC prediction, especially when facing hard prediction examples, because these drawbacks can decrease the accuracy of long-term predictions. Fortunately, the multi-generator model has been shown to work when facing these problems in other domains (Dendorfer, Elflein, and Leal-Taixé 2021).

As previously mentioned, a TC is a complex weather system and is influenced by many factors, such as environment-related factors, including the month, the region of the subtropical high, and the location of the TC. However, these factors were ignored in previous deep learning methods (Chen, Zhang, and Wang 2020). To make the prediction results more credible and interpretable, the prediction methods should consider the influence of environment-related factors and use such data efficiently.

To tackle all these problems, we propose a **M**ulti-**G**enerator **T**ropical **C**yclone **F**orecasting model, named **MGTCF**, which is a generic, extensible, multi-modal TC prediction model with the key modules of Generator Chooser Network (GC-Net) and Environment Network (Env-Net), for predicting multiple potential tendencies of TCs. The main contributions of MGTCF are as follows: 1) MGTCF can utilize heterogeneous meteorologic data efficiently, including the inherent attribute data of TC and the meteorological grid data. 2) The Multi-generator and the GC-Net are used to tackle the prediction of undesired OOD samples and the insufficient learning ability of single-generator TC prediction methods. 3) Env-Net improves the performance of TC prediction by embedding the environment information, which has been traditionally overlooked but is very important. To our knowledge, this is the first attempt to build a module focused on environment in TC prediction. 4) Extensive experiments were conducted on the China Meteorological Administration Tropical Cyclone Best Track Dataset (CMA-BST). We obtained state-of-the-art performance in deep learning methods and obtained better results than the method of the CMO in most indexes.

Method

Definition of the Problem

We propose MGTCF to predict the trajectory and intensity of TCs in a specific environment. This is also a spatial-temporal prediction problem. We denote the inherent attribute data of a TC as $Data_{1d}$, which include the longitude, latitude, pressure, and wind. We denote the meteorological grid data, such as geopotential heigh (GPH) (ECMWF 2022), as $Data_{2d}$. Environment data, such as the month, the velocity of movement, history of moving direction (24 h), and the region of the subtropical high, are denoted as *Env-Data*. Assuming there is a TC developing in the Pacific Ocean, we receive the history data x_t and the environment data x_{env}^n of this TC as input $\mathbf{X} = \{x_1, x_2, ..., x_n; x_{env}^n\}$. The every point time data is $x_t = \{x_{1d}^t, x_{2d}^t\}$, where t $\in \{1, 2, ..., n\}$, n is the sequence length of the input data, and x_{1d}^t is the $Data_{1d}$ and x_{2d}^t is the $Data_{2d}$. When we obtain the input $\mathbf{X}$, we want our model to be able to output $\hat{\mathbf{Y}} = \{\hat{y}_{n+1}, \hat{y}_{n+2}, ..., \hat{y}_{n+m}\}$ which is as close as possible to the actual future trajectory and intensity of TC (ground-truth). The actual future tendency of TC is denoted as $\mathbf{Y} = \{y_{n+1}, y_{n+2}, ..., y_{n+m}\}$. In addition, the real and prediction data of TC on each point time are defined as $y_t = \{y_{trajectory}^t, y_{intensity}^t\}$ and $\hat{y}_t = \{\hat{y}_{trajectory}^t, \hat{y}_{intensity}^t\}$ respectively, where t $\in \{n+1, n+2, ..., n+m\}$. Here, m is the sequence length of the prediction data of TC.

MGTCF

MGTCF is a GAN-based TC prediction model with the ability to receive multi-dimension and multi-modal data. Therefore, we need to build a **2D-Data Encoder** and a **1D-Data Encoder** to extract useful the features of these sequential data. Considering the environment data, we develop the **Environment Network** to extract the environment information efficiently. When we obtain the rich features of TC history data and environmental factors, we use **Multiple Generators** to predict the multiple potential tendencies of TC. Using the information about the environment and TC, our model can choose different generators to predict tendencies more accurately by the **Generator Chooser Network**. Then, a **Discriminator** is used to judge the actual tendency and the predicted tendency. Finally, to obtain the best performance, we optimize MGTCF by multiple **Loss Functions**. The framework of MGTCF is shown in Figure 2.

2D-Data Encoder. As shown in Figure 2, the 2D-Data Encoder is the red branch. This branch has two functions: to extract the spatial-temporal features from the history TC sequence data by a component, which is based on 3D-Unet(Çiçek et al. 2016); and to predict the future $Data_{2d}$ sequence by the history $Data_{2d}$ sequence. These predicted data will be received by the Decoder-LSTM (De-LSTM) in **Multiple Generators**. The process of the 2D-Data Encoder can be expressed as follows:

$$e_{2d}^{En}, e_{2d}^{De} = Encoder_{2d}(\mathbf{X}_{2d}; W_{Encoder_{2d}}) \tag{1}$$

where $\mathbf{X}_{2d}$ denotes the history sequence $Data_{2d}$ of TC, e_{2d}^{En} denotes the $Data_{2d}$ feature input for the 1D-Data Encoder, e_{2d}^{De} denotes the $Data_{2d}$ feature input for Generators, $Encoder_{2d}(.)$ is a function combining 3D-Unet with several Multilayer Perceptrons (MLPs), and $W_{Encoder_{2d}}$ denotes the weight of 3D-Unet and MLPs. Now, we complete the feature extraction of $Data_{2d}$ and get features e_{2d}^{En} and e_{2d}^{De}.

1D-Data Encoder. In Figure 2, 1D-Data Encoder is the blue branch. This branch is a fundamental part of MGTCF. It not only encodes the $Data_{1d}$ but also fuses the features from the 2D-Data Encoder e_{2d}^{En}. First, 1D-Data Encoder encodes $Data_{1d}$ with an MLP and gets features e_{1d}^{En}. Then, we concatenate e_{2d}^{En} and e_{1d}^{En} as the input of MLP and obtain e_{fusion}^{En}, which is fused with the features from two different dimensions. Finally, En-LSTM is used to extract the temporal features from e_{fusion}^{En} . The major process of 1D-Data Encoder can be expressed as follows:

$$e_{1d}^{En} = \phi(\mathbf{X}_{1d}; W_{MLP-1d}) \tag{2}$$

$$e_{fusion}^{En} = \phi(cat(e_{1d}^{En}, e_{2d}^{En}); W_{MLP-fusion}) \tag{3}$$

$$h_n = Encoder_{LSTM}(e_{fusion}^{En}; W_{En-LSTM}) \tag{4}$$

where $\phi(.)$ is an MLP module, and $Encoder_{LSTM}()$ represents the function of En-LSTM. W_{MLP-1d}, $W_{MLP-fusion}$, and $W_{En-LSTM}$ are the corresponding weights of modules. h_n denotes the temporal-spatial features from the history $Data_{1d}$ and $Data_{2d}$ of TC, and n is the sequence length of the input data.

Environment Network (Env-Net). Unlike previous works on TC prediction (Huang et al. 2022; Alemany et al. 2019), we pay more attention to the impact of environment-related information, which plays a critical role in the development of TCs. Therefore, we build the Env-Net, depicted as a golden branch in Figure 2, to extract features of the environment. The Env-Net can receive not only the temporal sequence data but also the environment data at a specific time, because not all environment data have temporal characteristics. For example, the season for one TC, without temporal characteristics, usually does not change during the development of the TC. However, the season represents significant information for using in the TC prediction, as TCs have different tendencies in different seasons (Anthes 2016). In general, environment data are also heterogeneous meteorological data, as can be seen with $Data_{1d}$ and $Data_{2d}$. Therefore, we design the Env-Net with Convolutional Neural Networks (CNN) and MLP to adapt data with different modals and dimensions. The main process of Env-Net is:

$$e_{1d}^{Env} = \phi(\mathbf{X}_{1d}^{Env}; W_{MLP-Env}) \tag{5}$$

$$e_{2d}^{Env} = CNN(\mathbf{X}_{2d}^{Env}; W_{CNN-Env}) \tag{6}$$

$$e_{Env} = \phi(cat(e_{1d}^{Env}, e_{2d}^{Env}); W_{MLP-Envfusion}) \quad (7)$$

where $\mathbf{X}_{1d}^{Env}$ and $\mathbf{X}_{2d}^{Env}$ are the $Data_{1d}$ and $Data_{2d}$ of environment respectively, and e_{1d}^{Env} represents the features of $Data_{1d}$ and e_{2d}^{Env} represents the features of $Data_{2d}$. $W_{MLP-Env}$, $W_{MLP-Envfusion}$, and $W_{CNN-Env}$ are the corresponding weights of modules, and e_{Env} represents the final features extracted from environment data.

Generator Chooser Network (GC-Net). Due to the prediction of undesirable out-of-distribution (OOD) samples and the insufficient learning ability drawbacks of single-generator TC prediction methods, we attempted to use multiple generators. In addition, we want our model to be able to select suitable generators for prediction according to environmental and TC information. We consider probability of each generator being selected to be different during different TC predictions, that is, $P(g_i|h_n, e_{Env})$ is different, where $P(.)$ is the probability of each generator g_i, $i \in \{1, 2, ..., K\}$. Here, K is the number of generators. Each generator will have its own prediction tendency when it meets a different environment. For instance, for the trajectory prediction of TC, when one of the generators g_q is more likely to predict the northwest trajectory with a specific environment env_i and our method also considers the future direction of the TC trajectory as northwest in the environment env_i, the $P(g_q|h_n, e_{Env})$ should be larger than that of other generators. This means that the probability of g_q being selected is the highest in these generators. Only in this situation, does our prediction results become more reasonable and accurate. To obtain the probability of each generator, we design the GC-Net. The inputs of GC-Net are the multimodal features h_n and e_{Env}, and the outputs are the probabilities of each generator. The probability histogram below the GC-Net in Figure 2 represents the probabilities of each generator. The main process of GC-Net is:

$$P = \phi(cat(h_n, e_{Env}); W_{MLP-GCNet}) \quad (8)$$

where P is the probability array of each generator being chosen, $\phi(.)$ is an MLP module, and $W_{MLP-GCNet}$ is the weight of GC-Net. Then, we use the Monte Carlo method to obtain the identification (id) list gid_{list} of the selected generators by the P and the sampling number l. The length of gid_{list} is l.

Multiple Generators. As previously mentioned, we use multiple generators to solve the OOD and the insufficient learning ability drawbacks of single-generator TC prediction methods. Each generator obtains distinct features and has its own prediction tendency by the specific condition, as shown in the three sub-figures at the left of Figure 2. All generators are designed with the same architecture but they do not share weights with each other. De-LSTM is the main module of each generator. As shown in Figure 2, each generator has its own id from 1 to K. Our method selects different generators to predict by the id list gid_{list} and we obtain multiple potential tendencies of TC, as shown in Figure 2. The selected generator receives the features $v = [h_n, e_{2d}^{De}, e_{Env}]$ from the **1D-Data Encoder**, **2D-Data Encoder**, and **Env-Net** and decodes v with a random noise vector $z \sim \mathcal{N}(0, 1)$ (Gupta et al. 2018) to forecast the future trajectory and intensity of the TC. The main process of the generator g_i is:

$$\hat{\mathbf{Y}}_i = Decoder_{LSTM_i}(v, z_i; W_{De-LSTM_i}) \quad (9)$$

where $\hat{\mathbf{Y}}_i$ is the trajectory and intensity prediction of TC, $Decoder_{LSTM_i}(.)$ is a decoder function of the generator g_i, and $W_{De-LSTM_i}$ is the weight of the generator g_i, i $\in$ gid_{list}. Now, we can finish generating the TC's future tendency. This means that we have completed the generator step. Next, we will introduce the discriminator step.

Discriminator. To make the predictions of generators more like the real TC, we build the Discriminator to extract the features of its input and let it judge whether its input data belong to the real TC : $TC_{real} = [\mathbf{X}, \mathbf{Y}]$ or the prediction TC : $TC_{fake} = [\mathbf{X}, \hat{\mathbf{Y}}]$. Ideally, the Discriminator can learn the rule of TC development and regard any unreasonable tendency as "Fake" during training.

Loss Functions

After the above introduction of the key modules in our method, we know the architecture of MGTCF and why it is designed that way. To optimize our module, we use four loss functions: adversarial loss, best of many loss, 2D-Data loss, and GC-Net loss.

Adversarial Loss. Adversarial loss is a classical loss in GAN-based models. We use the original adversarial loss to optimize Generators G and Discriminator D. We want the output TC_{fake} of G to be accepted by D just as TC_{real} can be accepted by D. We also want D to be able to distinguish the unreasonable TC_{fake} in all the samples, all of which are received by D, and prompt G to generate more reasonable and accurate future tendencies of TC. The adversarial loss in our method reads as follows:

$$\mathcal{L}_{adv} = H(D([\mathbf{X}, \mathbf{Y}]), T) + H(D([\mathbf{X}, G(\mathbf{X})]), F) \quad (10)$$

where H is a cross-entropy loss, $D(.)$ outputs the probability that the input is true. $G(.)$ outputs the future TC prediction, $[\mathbf{X}, \mathbf{Y}]$ represents TC_{real}, $[\mathbf{X}, G(\mathbf{X})]$ represents TC_{fake}, T is the label of TC_{real}, and F is the label of TC_{Fake}.

Best of Many Loss. The best of many loss is a common loss in the tasks of pedestrian trajectory prediction(Gupta et al. 2018). We only choose the best prediction in the outputs of generators and calculate the loss between it and the ground truth. This loss encourages MGTCF to predict reasonable multiple potential tendencies of TC. The definition of this is as follows:

$$\mathcal{L}_{BMS} = \min_{l} \left\| Y - \hat{Y}_i \right\|_2 \quad (11)$$

where $i \in gid_{list}$, l is the number of samples, and $\|.\|_2$ is the l_2-norm function.

2D-Data Loss. In order for our method to extract better temporal-spatial features from $Data_{2d}$, we add the 2D-Data loss to further optimize 2D-Data Encoder. As described in an earlier section, we use 3D U-Net to predict the future $Data_{2d}$ sequence $\hat{\mathbf{Y}}_{2d}$. We calculate the loss between $\hat{\mathbf{Y}}_{2d}$ and the real $Data_{2d}$ sequence $\mathbf{Y}_{2d}$ in pixel wise, the main process is:

$$\mathcal{L}_{2D} = \frac{1}{hw \times c \times m} \left\| \mathbf{Y}_{2d}^{hw \times c \times m} - \hat{\mathbf{Y}}_{2d}^{hw \times c \times m} \right\|_1 \tag{12}$$

where hw is the spatial size of $Data_{2d}$, c denotes the channel number, m is the length of prediction sequence, and $\|.\|_1$ is the l_1-norm function. Overall, the training object of the GAN part of MGTCF can be expressed follows:

$$\min_G \max_D \mathcal{L}_{adv} + \lambda_{BMS}\mathcal{L}_{BMS} + \lambda_{2D}\mathcal{L}_{2D} \tag{13}$$

GC-Net Loss. It is important that GC-Net can select suitable generators conditioned on the specific environment and TC. Therefore, we use GC-Net loss to optimize the GC-Net module. We utilize the real future TC $\mathbf{Y}$ and the predicted TC $\hat{\mathbf{Y}}$ to approximate the likelihood of a particular generator distribution P_i. The main process is:

$$P(\mathbf{Y}|h_n, e_{env}, g) \propto \frac{1}{l} \sum_{i \in gid_{list}} exp\left(\frac{-\left\| \hat{\mathbf{Y}}_i - \mathbf{Y} \right\|_2^2}{2\sigma} \right) \tag{14}$$

Next, we obtain the conditional probability of each generator by Bayes' Rule:

$$P(g|h_n, e_{env}, \mathbf{Y}) = \frac{P(\mathbf{Y}|h_n, e_{env}, g)}{\sum_{i=1}^{K} P(\mathbf{Y}|h_n, e_{env}, g_i)} \tag{15}$$

In the final step, we use the cross-entropy loss between the distribution of $P(g|h_n, e_{env}, \mathbf{Y})$ and the GC-Net's output to optimize GC-Net:

$$\mathcal{L}_{GC} = H(P(g|h_n, e_{env}, \mathbf{Y}), GC(h_n, e_{env})) \tag{16}$$

where $GC(.)$ is the probability array of each generator by GC-Net.

Training Scheme. The prediction performance of generators will be greatly influenced by the output P of GC-Net. The more suitable generators our model selects, the more accurate our model's predictions. Accordingly, we first train GC-Net using Equation (16) with freezing the rest of our model's parameters in the first q epochs. After this step, GC-Net can provide a selection scheme of generators along initialized parameters. It is helpful to train the GAN part of our model. Then we train all the parameters of MGTCF and obtain the best performance model by using Equations (16) and (13).

Experiments

We conduct comparison experiments and ablation studies. We also discuss the experimental setup to help our readers to reproduce our method. We will also open our codes on Github at https://github.com/Zjut-MultimediaPlus/MGTCF.

Experimental Setup

We implemented MGTCF on the PyTorch platform and ran it on an NVIDIA RTX A6000 GPU. We used Adam (Kingma and Ba 2015) to optimize our model with an initial learning rate of 0.0001. We trained MGTCF with a batch size of 96 and 100+q epochs. The hyperparameter q was 2. The number of generators K and the sampling number l were set as 6. In all experiments, we input the past 48 h (n = 8) TC data and predicted the future 24 h (m = 4) tendencies of TC. More details of experimental setup are provided in the supplementary material.

Datasets. We collected rich meteorological data about TCs from the year 1950 to 2019. These served as the cornerstone of our method. Of the data from 1950 to 2016, 80% were used for the training set and 20% were used for the validation set. The data for the years 2017 to 2019 were regarded as the test set. The main data we used consisted of inherent attribute data of TC ($Data_{1d}$), meteorological grid data ($Data_{2d}$), and environment data.

$Data_{1d}$ we used were from the CMA-BST (Ying et al. 2014). Trajectory and intensity are the most intuitive and crucial properties of TCs. Therefore, we selected data on the longitude, latitude (trajectory), pressure, and wind (intensity) at the center of TC as the training data to describe the TC.

$Data_{2d}$ we used, geopotential height (GPH), were from the fifth-generation atmospheric reanalysis of the global climate (ERA5) (ECMWF 2022) by the European Centre for Medium-Range Weather Forecasts (ECMWF). We followed (Liu et al. 2022) and chose the 500 hPa GPH data to describe the pressure structure of TCs.

Environment data used in our method described the TC condition during its development. We extracted these data from $Data_{1d}$ and $Data_{2d}$, including the month, the velocity of movement, history of moving direction (24 h), history of intensity change (24 h), the region of the subtropical high, and the location of the TC. We also provide these data in the supplementary material.

Metrics. Absolute error was used to evaluate the performance of the different methods. For trajectory prediction, we calculated the absolute distance (km) error between the real trajectory and the prediction trajectory. In intensity prediction, including pressure (hPa) and wind (m/s) at the TC's center, we also calculated the absolute error between the real data and the prediction data.

Comparison with State-of-the-art Methods

First, we conducted experiments comparing other state-of-the-art methods with MGTCF. Then, we analyzed the experimental results from multiple aspects. The results showed that our method obtained the best prediction performance of the deep learning methods. The results also showed that MGTCF outperformed CMO in most indexes.

Quantitative Analysis. The comparison results, shown in Table 1, were evaluated by the average absolute error of TC trajectory and intensity prediction. First, we looked at four methods: LSTM (Hochreiter and Schmidhuber 1997),

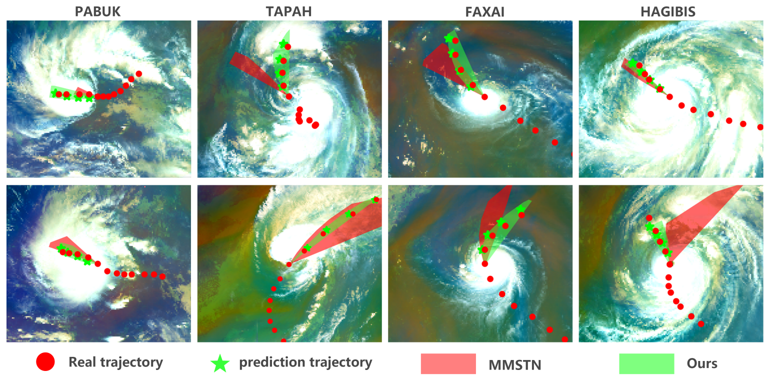


Figure 3: Examples of trajectory predictions from 6 h to 24 h and the comparison between our method and MMSTN on the potential region predictions for four TCs: PABUK (STS, Winter), TAPAH (TY, Summer), FAXAI (STY, Summer), and HAGIBIS (SuperTY, Summer). Best viewed in color.

methods	Distance (km)				Pressure (hPa)				Wind (m/s)			
	6 h	12 h	18 h	24 h	6 h	12 h	18 h	24 h	6 h	12 h	18 h	24 h
LSTM	44.23	102.24	177.60	270.84	-	-	-	-	-	-	-	-
GRU	45.85	104.07	180.29	275.77	-	-	-	-	-	-	-	-
NMPT	44.10	101.72	177.06	270.91	-	-	-	-	-	-	-	-
DLM	-	-	-	-	-	-	-	-	1.09	1.85	2.48	3.04
SGAN	28.88	61.75	98.74	140.61	1.91	3.12	4.20	5.12	1.05	1.69	2.28	2.81
GBRNN	29.93	65.06	105.74	152.06	-	-	-	-	1.16	1.89	2.52	3.10
MMSTN	27.57	59.09	96.54	139.19	1.69	2.86	3.94	4.74	0.95	1.52	2.10	2.55
CMO	37.08	52.93	**60.69**	**75.49**	2.67	4.30	5.04	6.31	2.29	3.45	2.75	5.00
MGTCF	**23.14**	**43.37**	67.09	93.08	**1.37**	**2.04**	**2.66**	**3.29**	**0.73**	**1.17**	**1.55**	**1.86**

Table 1: Comparisons of average absolute error of TC prediction of different methods.

GRU (Cho et al. 2014), NMPT (Gao et al. 2018), and DLM (Pan, Xu, and Shi 2019). In the experiment, they executed prediction tasks with trajectory or intensity modal data, and their performances were inferior to those of three methods–SGAN (Gupta et al. 2018), GBRNN (Alemany et al. 2019), and MMSTN (Huang et al. 2022)–that used trajectory and intensity information of the TC. We find that if a method can extract more features efficiently from more data, it can better forecast a TC. Our method, with the reception of heterogeneous meteorological data and effective model designs, including GC-Net, Env-Net, and multiple generators, achieved the best performance among the deep learning methods. In trajectory prediction, MGTCF surpassed the previous best method, which was MMSTN (Huang et al. 2022), by 16.08%-33.12%. In intensity prediction, our method demonstrated an 18.93%-32.49% improvement over MMSTN. Furthermore, as the prediction time became longer, MGTCF can achieved even more improvement.

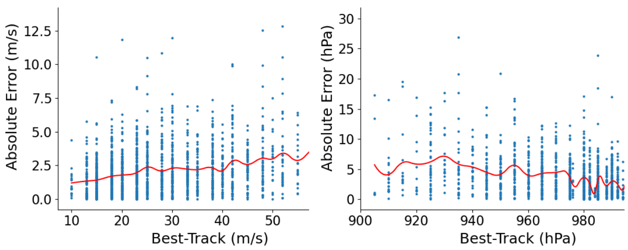
When compared with CMO (CMO 2019), MGTCF also performed better in intensity prediction and short-term trajectory prediction of a TC. Although COM outperformed MGTCF in 18 h and 24 h trajectory prediction, MGTCF only needs a GPU, not a supercomputer. We do not know which the specific method CMO used, but we know it belongs to NWP, which needs a supercomputer. In summary, MGTCF is an effective and potential method that achieves state-of-the-art performance among deep learning methods.

Qualitative Analysis of the Trajectory Prediction. To more intuitively prove the effectiveness of our method, we visualized the results of trajectory prediction and compared the prediction performances of MGTCF and the previous best deep learning method, MMSTN (Huang et al. 2022). As shown in Figure 3, the red circle points sequence represents the real trajectory, the semitransparent green region represents the potential tendencies calculated by our multiple trajectory predictions, the semitransparent red region represents the potential tendencies of MMSTN, the green star sequence represents the most accurate prediction trajectory generated by our method, and the background of prediction results represents the satellite cloud image of each TC.

To demonstrate the superior performance of MGTCF, we selected four TCs of different grades: severe tropical storm (STS) PABUK, typhoon (TY) TAPAH, severe typhoon (STY) FAXAI, and super typhoon (SuperTY) HAGIBIS. PABUK was generated in winter; the other three TCs were generated in summer. The semitransparent green regions show that MGTCF performs well when observing the TC impacted by different environmental factors (e.g., the grade and the season of TC). Expecting our model to tackle the TC in a different environment was one of the main mo-

H	M-GC	Env	Distance (km)				Pressure (hPa)				Wind (m/s)			
			6 h	12 h	18 h	24 h	6 h	12 h	18 h	24 h	6 h	12 h	18 h	24 h
			27.57	59.09	96.54	139.19	1.69	2.86	3.94	4.81	0.95	1.52	2.10	2.59
√			24.95	48.76	74.47	104.58	1.52	2.40	3.30	3.93	0.77	1.29	1.77	2.16
√		√	25.33	48.21	72.20	100.84	1.41	2.28	2.98	3.44	0.76	1.26	1.66	1.92
√	√		25.01	46.16	68.41	96.79	**1.29**	**2.02**	2.67	3.34	**0.71**	**1.17**	1.56	1.96
√	√	√	**23.14**	**43.37**	**67.09**	**93.08**	1.37	2.04	**2.66**	**3.29**	0.73	1.18	**1.55**	**1.86**

Table 2: Ablation experiments: (H) Receiving Heterogeneous Meteorological Data, (M-GC) Multi-Generators with Generator Chooser Network, and (Env) Environment Network.



(a) (b)

Figure 4: Absolute error distribution of intensity (pressure and wind) in 6 h TC prediction. Best viewed in color.

tivations for building the key module Env-Net. Otherwise, the difficulty of forecasting different TCs is different. When tackling the easy problem of linear prediction, shown in HAGIBIS's first image, MGTCF achieved results comparable with MMSTN and provided an accurate and small potential region of prediction. In the difficult example of nonlinear prediction, MGTCF provided a larger (when compared with the easy samples) but reasonable potential region of prediction. Because it is impossible for either the meteorological experts or our method to predict the future of TC flawlessly, all we need to do is to provide a reasonable and accurate prediction conditioned on limited information as much as possible. Fortunately, our multi-generators with GC-Net can perform this task. Each generator of our method can obtain the feature it wants from the history TC data and environmental factors to make a prediction. This mechanism can make predictions that each generator thinks are reasonable and prevent our method from predicting OOD samples. When facing a difficult example, each generator of MGTCF has its own prediction tendency based on known information and these prediction tendencies can constitute a more referential and more robust forecasting region for meteorologists and government officials. Therefore, the potential region of the difficult example is bigger than the region of the easy example. In addition, the performance of our method is much better than that of the MMSTN in the difficult example. All of this further proves the effectiveness of our method.

Analysis of Intensity Error Disruption. The intensity predictions were also analyzed, and the results are shown in Figure 4. Subplot (a) describes the relationship between the absolute error of predicted wind and the wind recorded in the CMA-BST. Subplot (b) demonstrates the relationship between the absolute error of predicted pressure and the pressure recorded in CMA-BST. The red curve represents a polynomial fitting curve from the relationship. We found that the number of intensity errors increased with the increase in TC intensity. This means that the mutability of the intensity of a strong TC is a big challenge in the task of TC prediction, and it needs to be focused on in the future.

Ablation Studies

We conducted ablative experiments to show the effectiveness of our method. The results are shown in Table 2, including the design for receiving heterogeneous meteorological data (H), the Multi-Generators with GC-Net (M-GC), and Env-Net (Env). First, it can be seen that the method without H was outperformed by the method with H. This means that the design for receiving heterogeneous meteorological data in MGTCF is effective, showing an average improvement of 18.68% in trajectory prediction and 15.88% in intensity prediction. Second, we prove the contribution of Env through the comparison of the method with H and the method with H and Env. The Env module provided 2.69% and 6.92% average improvement, respectively. Then, due to the stronger learning ability of M-GC, the method with H and M-GC obtained 5.17% and 13.02% average improvement in trajectory and intensity prediction, respectively. M-GC provided even more improvement in the 18 h and 24 h predictions, with 7.79% and 14.02%, respectively. This means that M-GC can prevent our model from predicting OOD tendencies in order to improve the long-term (18 h and 24 h) prediction of TC. Finally, the entire model with H, M-GC, and Env, named MGTCF, achieved a significant improvement in performance when compared with the method without any of our contributions. In summary, the ablative experiments proved that our design and the key modules are effective.

Conclusion

In this paper, we propose MGTCF for the efficient use of heterogeneous meteorological data. We also focus on the environment-related information that other deep learning methods neglect by building a module, called Env-Net, to extract rich environment features. In addition, we propose a Multi-generator with Generator Chooser Net to resolve some of the drawbacks of single-generator methods. Finally, we conducted extensive experiments and analyzed how the design of our model and its key modules works in TC prediction. The experimental results proved that MGTCF would be an effective method for predicting TCs.

Acknowledgments

This work is supported by National Key Research and Development Program of China (No. 2018YFE0126100), Zhejiang Provincial Natural Science Foundation of China under Grant No. LR21F020002, Major Basic Research Projects of Shandong Province Grant No. ZR2022ZD32, and Open Foundation of Key Laboratory of Meteorological Disaster under Grant No.KLME202108.

References

Alemany, S.; Beltran, J.; Perez, A.; and Ganzfried, S. 2019. Predicting hurricane trajectories using a recurrent neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 468–475.

Anthes, R. 2016. *Tropical cyclones: their evolution, structure and effects*, volume 19. Springer.

Chen, R.; Zhang, W.; and Wang, X. 2020. Machine learning in tropical cyclone forecast modeling: A review. *Atmosphere*, 11(7): 676.

Chen, X.; and Zhang, F. 2019. Development of a convection-permitting air-sea-coupled ensemble data assimilation system for tropical cyclone prediction. *Journal of Advances in Modeling Earth Systems*, 11(11): 3474–3496.

Cho, K.; van Merrienboer, B.; Gülçehre, Ç.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In Moschitti, A.; Pang, B.; and Daelemans, W., eds., *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 1724–1734. ACL.

Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S. S.; Brox, T.; and Ronneberger, O. 2016. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In Ourselin, S.; Joskowicz, L.; Sabuncu, M. R.; Ünal, G. B.; and III, W. M. W., eds., *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016 - 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II*, volume 9901 of *Lecture Notes in Computer Science*, 424–432.

CMO, C. M. O. 2019. Typhoon network of Central Meteorological Observatory. http://typhoon.nmc.cn/web.html. Accessed: 2003-03-10.

Dendorfer, P.; Elflein, S.; and Leal-Taixé, L. 2021. MG-GAN: A Multi-Generator Model Preventing Out-of-Distribution Samples in Pedestrian Trajectory Prediction. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, 13138–13147. IEEE.

Duan, J.; Wang, L.; Long, C.; Zhou, S.; Zheng, F.; Shi, L.; and Hua, G. 2022. Complementary Attention Gated Network for Pedestrian Trajectory Prediction. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, 542–550. AAAI Press.

ECMWF. 2022. ERA5 hourly data on pressure levels from 1959 to present. https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-pressure-levels?tab=overview. Accessed: 2016-11-04.

Emanuel, K. 2007. Environmental Factors Affecting Tropical Cyclone Power Dissipation. *Journal of Climate*, 20(22): 5497 – 5509.

Emanuel, K. 2018. 100 Years of Progress in Tropical Cyclone Research. *Meteorological Monographs*, 59: 15.1 – 15.68.

Fan, J.; Bai, J.; Li, Z.; Ortiz-Bobea, A.; and Gomes, C. P. 2022. A GNN-RNN Approach for Harnessing Geospatial and Temporal Information: Application to Crop Yield Prediction. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, 11873–11881. AAAI Press.

Gao, S.; Zhao, P.; Pan, B.; Li, Y.; Zhou, M.; Xu, J.; Zhong, S.; and Shi, Z. 2018. A nowcasting model for the prediction of typhoon tracks based on a long short term memory neural network. *Acta Oceanologica Sinica*, 37(5): 8–12.

Giffard-Roisin, S.; Yang, M.; Charpiat, G.; Kumler-Bonfanti, C.; Kégl, B.; and Monteleoni, C. 2020. Tropical Cyclone Track Forecasting Using Fused Deep Learning From Aligned Reanalysis Data. *Frontiers Big Data*, 3: 1.

Gupta, A.; Johnson, J.; Fei-Fei, L.; Savarese, S.; and Alahi, A. 2018. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2255–2264.

Hochreiter, S.; and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*, 9(8): 1735–1780.

Huang, C.; Bai, C.; Chan, S.; and Zhang, J. 2022. MMSTN: A Multi-Modal Spatial-Temporal Network for Tropical Cyclone Short-Term Prediction. *Geophysical Research Letters*, 49(4): e2021GL096898. E2021GL096898 2021GL096898.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In Bengio, Y.; and LeCun, Y., eds., *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kumar, S.; Biswas, K.; and Pandey, A. K. 2021. Prediction of Landfall Intensity, Location, and Time of a Tropical Cyclone. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, 14831–14839. AAAI Press.

Liu, Z.; Hao, K.; Geng, X.; Zou, Z.; and Shi, Z. 2022. Dual-Branched Spatio-Temporal Fusion Network for Multihorizon Tropical Cyclone Track Forecast. *IEEE Journal of Se-*

lected Topics in Applied Earth Observations and Remote Sensing, 15: 3842–3852.

Neumann, C.; and Lawrence, M. 1975. An Operational Experiment in the Statistical-Dynamical Prediction of Tropical Cyclone Motion. *Monthly Weather Review - MON WEATHER REV*, 103.

Pan, B.; Xu, X.; and Shi, Z. 2019. Tropical cyclone intensity prediction based on recurrent neural networks. *Electronics Letters*, 55(7): 413–415.

Rüttgers, M.; Lee, S.; Jeon, S.; and You, D. 2019. Prediction of a typhoon track using a generative adversarial network and satellite images. *Scientific reports*, 9(1): 1–15.

Wang, Y.; Rao, Y.; Tan, Z.-M.; and Schönemann, D. 2015. A statistical analysis of the effects of vertical wind shear on tropical cyclone intensity change over the western North Pacific. *Monthly Weather Review*, 143(9): 3434–3453.

Ying, M.; Zhang, W.; Yu, H.; Lu, X.; Feng, J.; Fan, Y.; Zhu, Y.; and Chen, D. 2014. An Overview of the China Meteorological Administration Tropical Cyclone Database. *Journal of Atmospheric and Oceanic Technology*, 31(2): 287 – 301.

Zhao, X.; Fan, W.; Liu, H.; and Tang, J. 2022. Multi-Type Urban Crime Prediction. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, 4388–4396. AAAI Press.