Received: 25 July 2022 | Revised: 27 August 2022 | Accepted: 14 September 2022 | Published online: 19 September 2022

RESEARCH ARTICLE

Artificial Intelligence and Applications
2023, Vol. 1(2) 106–113
DOI: 10.47852/bonviewAIA2202337

BON VIEW PUBLISHING

UTrans-Net: A Model for Short-Term Precipitation Prediction

Hao Cao[1], Yirui Wu[1,*], Yansong Bao[2], Xi Feng[3], Shaohua Wan[4] and Cheng Qian[5]

[1]*College of Computer and Information, Hohai University, China*

[2]*College of Atmospheric Physics, Nanjing University of Information Science and Technology, China*

[3]*College of Harbor, Coastal and Offshore Engineering, Hohai University, China*

[4]*Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, China*

[5]*Hydraulic Research Institute of Jiangsu Province, China*

Abstract: This short-term weather forecast is particularly important for human production activities and safety. However, the existing short-term weather forecasts are often difficult to meet the demands in terms of precipitation accuracy, and traditional numerical weather forecasts have low efficiency in short-term weather forecasting. Artificial intelligence (AI) can infer uncertain information based on data with lower temporal and spatial density and can use abstract knowledge in statistical and numerical models, thereby improving the prediction accuracy of short-term weather and precipitation. However, traditional AI weather forecasting technology often cannot perform feature extraction well. Therefore, this paper proposes a new U-shaped neural network model UTrans-Net that combines transformer. This proposed model adds transformer to the U-Net model to determine the weights of different meteorological elements. Afterwards, the precipitation at a later time is predicted by using the monitoring values of the previous series of time points, and the parameters of the neural network are adjusted according to the results. By conducting experiments on the weather dataset of 2400 samples given by the China Meteorological Administration, the results of our experiment show that UTrans-Net has a more accurate prediction accuracy, and the average prediction accuracy at 0.1, 3, and 100 mm thresholds is 59%, 78%, and 82%, respectively.

Keywords: precipitation prediction, transformer, deep learning, attention module

1. Introduction

As weather forecast becomes more and more important in production and life, the demand for the accuracy of short-term weather forecast is increasing day by day. As an important part of weather forecast, precipitation prediction can also be applied in many fields.

Generally speaking, precipitation is affected by multi-scale physical factors, such as air temperature, air pressure, and water vapor content (Zhang et al., 2018). With the research on precipitation prediction for many years, there are mainly two basic methods at present, namely numerical weather prediction and artificial intelligence (AI) methods (Jing, Li, Ding et al., 2019; Zhang et al., 2021). However, since short-term rainfall is a strongly random, non-linear event and can be affected by unexpected situations (such as the sudden disappearance of the main body of precipitation), it is difficult to predict the precipitation for a period of time in the future (Nguyen et al., 2021; Shi et al., 2015). Existing researchers have tried to use the sequence autoregressive model (seq2seq) to achieve effective use of historical information. By taking historical observation information and historical forecast information as the input of the next prediction moment, this method effectively uses historical spatial and temporal information to correct the prediction error to a certain extent and makes the prediction model to have a certain risk resistance capability (Zhang et al., 2017). However, the network often has the defect of insufficient feature extraction, so how to fully and effectively extract the core meteorological information has become an important problem.

In order to improve the effectiveness of feature extraction of sequence data and solve the problem of low time efficiency and accuracy of traditional AI in precipitation prediction (Shi et al., 2018), this paper proposes a U-shaped network structure (UTrans-Net) using transformer to multi-scale meteorological elements for feature extraction (Trebing et al., 2021).

The contributions of this paper are as follows:

1. Generate multiple semantic vectors for the encoder and decoder at each moment through the attention mechanism in transformer, so as to distribute the weights of the semantic information preserving the current moment

*Corresponding author: Yirui Wu, College of Computer and Information, Hohai University, China. Email: wuyirui@hhu.edu.cn

information and obtain the current meteorological features. Based on variable-length semantic information representation, data loss caused by long sequence data can be avoided.
2. In order to reduce the problem of unstable precipitation prediction, a relatively smooth loss function Smooth-L1 is introduced in this paper to realize insensitive operation to outliers, thus improving the robustness against emergencies.
3. The experimental results show that the precipitation prediction accuracy of the proposed model is tested on the data provided by the China Meteorological Administration, and the prediction accuracy is improved compared with other methods.

2. Related Work

2.1. Numerical weather forecast

According to the actual atmospheric conditions and the monitored data conditions, numerical weather forecast is used to solve the equations of the weather evolution process by means of numerical calculation on a large computer so as to predict the atmospheric movement in a period of time in the future. Specifically, numerical weather forecast mainly collects data from the grid of the earth, and the size of the grid can occasionally affect the spatial and temporal resolution of numerical weather forecast. Then, the collected meteorological element data are substituted into the atmospheric equations describing atmospheric motion, and the atmospheric equations composed of partial differential equations can be calculated after a reasonable approximation, which includes the rainfall value of precipitation. However, the computer usually needs to consume more resources to perform the calculation of the equation when there are more types of meteorological elements. In addition, how to determine which meteorological elements have higher weights for precipitation prediction needs to be further analyzed.

Bauer et al. (2015) summarize the basic concepts of numerical weather forecast and further describe the basic principles and methods of atmospheric motion models. For the formulas of atmospheric motion, they give some of these formulas and explain how these formulas can be used to predict weather forecasts at future moments. In addition, they also explain the prospects and current bottlenecks of numerical weather forecasting, give some explanations of current classical models, and propose some evaluation indicators on how to evaluate the algorithms and models.

2.2. AI weather forecast

Weather forecast often requires a large number of monitoring data of meteorological elements and extracts effective meteorological elements from a series of abstract data. However, due to the problems of insufficient distribution of existing sites and relatively high monitoring costs, the current data have insufficient spatial and temporal density. In addition, there may be certain connections between different meteorological elements, and it is difficult for numerical forecast models to accurately describe the connections of some of these meteorological elements.

AI technology can infer information based on the monitored data and abstract data with insufficient spatial and temporal density. More importantly, AI technology can summarize expert knowledge and experience to improve the average prediction level. Due to the fit between the two, AI methods can often be a powerful complement to numerical forecasting. At present, AI technology and some other computer technologies have been widely used in weather forecast and achieve some remarkable results, and are considered to be effective methods by meteorological experts (Zhao et al., 2021). There are mainly two types of forecasting networks based on traditional AI methods and forecasting networks based on deep learning (DL).

Traditional methods mainly use the motion vector of radar echo images to forecast the weather, including optical flow method, particle filter method, cross-correlation method, and so on (Torcasio et al., 2021; De Andrade et al., 2021). Ayzel et al. (2019) track the motion of the precipitation feature by the optical flow method and replace the motion to the future predicted moment for prediction and keep the feature intensity unchanged. In addition, they also develop software based on different optical flow algorithms and test a set of benchmark models. The experimental results show that the precipitation prediction based on the optical flow method can achieve a certain improvement compared with the traditional numerical forecast. As mentioned above, AI technology can use abstract data and summarize expert knowledge and experience. Therefore, McGovern et al. (2017) extract the originally unavailable information through machine learning technology and data mining technology and make use of the characteristics of high accuracy to bridge the gap between numerical model and real-time model, thus improving the prediction accuracy of various types of high-impact weather, such as tornadoes. However, in most cases, traditional AI methods do not take into account the impact of multiple related and complex factors such as storm dynamics and thermodynamics on the echoes, and how to deal with such multi-related meteorological elements has become the improvement direction of the follow-up forecast.

The prediction network based on DL can predict the future time of echo through long short-term memory (LSTM) and other neural networks, which has certain advantages over optical flow method and has achieved some success and practical application in the near prediction, but it is not widely used in short-term weather forecasting. There have been some successful and practical applications, but the application in short-term weather forecasting is not very extensive. Chen et al. (2020) propose a prediction method Conv-LSTM based on convolutional LSTM network, which is a convolutional LSTM model with star bridge layers. The model solves the chaotic and dynamic problems in strong convective weather forecasting through an end-to-end trainable model established by radar echo data and a rain-oriented loss function. Furthermore, the use of normalization techniques in this model improves the convergence performance of deep networks, enabling predictions with effective spatiotemporal resolution. Erol et al. (2019) use generative adversarial network (GAN) to extract the information of radar echo images from a series of radar puzzle data by using the convolution method. The experiments show that GAN can be used to enhance the extrapolation effect of radar images, and it has a good effect in the prediction of medium echoes, but it still needs further optimization and improvement in the prediction of strong echoes. Therefore, in order to make full use of the advantages of these two methods, Jing, Li, and Peng (2019) propose a multi-level correlated LSTM model MLC-LSTM, which combines the GAN method with the LSTM method and introduces the idea of adversarial training to the volume LSTM network. The evolution of the model is carried out through the spatiotemporal correlation between the multi-level radar echoes, and the adversarial training is used to help the model to extrapolate the real echoes, thereby further improving the prediction accuracy. DeepMind also uses a deep generative model method similar to GAN, which proposes a conditional

generative model DGMR for immediate precipitation forecasting and makes detailed and reasonable predictions of future weather information through past radar data, and the effect is remarkable.

In order to further analyze the relationship between radar echo images, contextual information modeling methods in computer vision are used to make up for this deficiency. Although the high echo value part in the radar echo image may indicate the occurrence of weather such as heavy rain, it is often ignored in the actual analysis. Luo et al. (2021) develop an interaction framework by constructing a series of coupled convolutions of input and hidden states. The framework can make full use of short-term contextual information, and a dual attention mechanism of channel and position is established to model the forgotten information. The dual-channel attention in this method can make up for the lack of low-level details and ensure that the model makes full use of effective information.

In addition, after the introduction of DL and other methods, higher requirements are often placed on the original data. In current DL models, the input data lack necessary physical process information. Pan et al. (2021) introduce differential reflectivity ZDR and differential phase shift rate KDP to describe microphysical and dynamic structural information and propose a new DL architecture. Through a high-dimensional fusion strategy, the architecture is able to capture the evolutionary characteristics of the precipitation system while maintaining multi-scale spatial information. This method, that is, introducing more meteorological factors that meet the requirements according to the target architecture, is a suitable choice and subsequent optimization direction in precipitation prediction.

2.3. Precipitation prediction

Precipitation is an event with both spatial and temporal features, so the model needs to have the ability to extract both temporal and spatial features. Shi & Yeung (2018) make the ConvLSTM2D model able to extract spatial features by replacing the dot product operation in the LSTM gate with convolution, which make up for the difficulty of extracting spatial information in the temporal convolutional network model (Hewage et al., 2020). Through state transfer, that is, skip connection, the feature evolution of the model can be captured on the basis of maintaining multi-scale spatial information.

Although these models have achieved good results in the near forecast, they still have the same problem as the numerical weather forecast in the short-term weather forecast (24–72 hours), that is, there is still a great deal of uncertainty in the prediction of emergent conditions such as newborn rain belt. Even if it is theoretically unavoidable, Seq2seq models can minimize this effect as much as possible. By using historical observation information and historical prediction information in the encoder stage of LSTM, and taking each prediction information as input in the decoder stage, the information of each observation can be utilized as much as possible. Zhang et al. (2020) constructed Seq2seq with an attention model for each cluster to screen out the more favorable physical factors for inferring precipitation, and the results show that the Seq2seq model outperformed other existing methods.

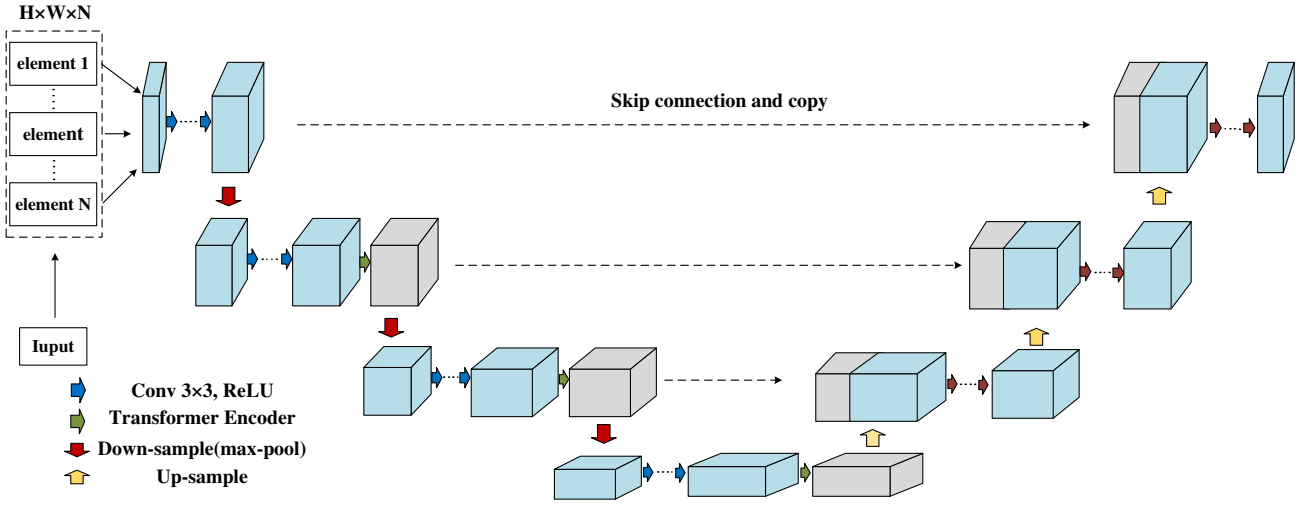
The above researches predict the extracted information through the features, and further optimizing the feature extraction can better improve the accuracy of the prediction. Diao et al. (2019) propose a short-term weather forecast model based on wavelet denoising and Catboost, which combines correlation heatmap, recursive feature elimination, and tree model and learns in advance through wavelet denoising for better feature extraction. The experiments show that this model has higher accuracy and shorter convergence time. Xie et al. (2021) update transformer structure by using layered encoder to output multi-scale features, so that features with high resolution coarse-grained and low resolution fine-grained can be captured and optimized together. In addition, the framework does not require position coding, thus avoiding the interpolation of position coding that results in performance degradation when the resolution is different between test and training. The results show that after feature extraction and optimization, the performance can be significantly improved and the amount of computation can be reduced.

3. Method

3.1. Baseline architecture

The basic framework of this paper is shown in Figure 1. The core part of the baseline framework is the U-Net structure, which is a classic fully convolutional network (Chen et al., 2021; Ronneberger et al., 2015).

Figure 1
Baseline architecture



The U-Net structure consists of convolution, max pooling, and up-sampling. In the contracting path of the left part of the network, that is, the down-sampling operation part, effective convolution is performed through blocks of different sizes, and then a down-sampling operation is completed according to a max pooling operation to obtain the feature map. In the expanded path on the right side of the network, that is, the up-sampling part, the feature map is deconvolved by using blocks of different sizes. In addition, since the size of the feature maps of the left compression path and the right expansion path are different, skip connection operations are performed in each layer structure. In this part, the corresponding feature map in the lower sampling layer is cut and spliced with the feature map obtained from the upper sampling layer to realize the normalized operation, and the process is repeated at each layer. In this way, the semantic information lost due to the reduction of feature map resolution caused by effective convolution in the up-sampling process can be recovered to a certain extent, thus ensuring a certain accuracy.

In addition, in our Baseline architecture, we also add transformer to obtain global self-attention, which compensates for U-Net's limitations in long-range dependencies.

Figure 2 gives the specific details of the transformer encoder (Vaswani et al., 2017). First, the given input information is represented by $E = [e_1, e_2, \ldots, e_N]$. After a certain linear transformation, the vectors of Q, K, and V can be obtained, as shown below:

$$Q = W_q E \tag{1}$$

$$K = W_k E \tag{2}$$

$$V = W_v E \tag{3}$$

where Q, K, and V represent the query vector, the correlation vector between the queried information and other information, and the vector of the queried information, respectively.

Then, the formula of the dot product self-attention mechanism can be given, as shown below, and W_q, W_k, W_v will be updated and changed according to the task goal, so as to ensure the effect of the self-attention mechanism:

$$Z_i = Attention(Q_i, K_i, V_i) = V_i \cdot softmax\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) \tag{4}$$

where Z represents different weight matrices, QK^T is the obtained attention matrix, $\sqrt{d_k}$ is used to turn the attention matrix into a standard normal distribution, and $softmax(\cdot)$ is used for normalization to ensure that the sum of the weights is 1, which makes the results after normalization more stable.

Figure 2
Transformer encoder

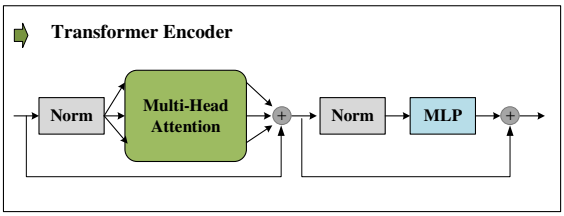
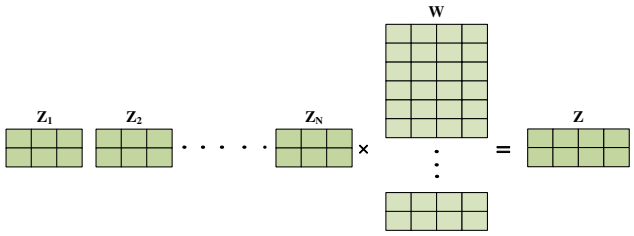


Figure 3
Input of feedforward neural network



In addition, through the multi-head attention mechanism existing in the encoder can be obtained

$$Z = Z(Q, K, V) = Z_1 \oplus Z_2 \oplus \ldots \oplus Z_N \tag{5}$$

where $\oplus$ represents the contact operation. By splicing multiple weight matrices Z_i, the output of the feedforward neural network layer (self-attention layer) can be obtained. However, since the feedforward neural network only needs one matrix, an additional weight matrix is multiplied to it. Figure 3 shows a schematic diagram of this matrix splicing. Through the multiplication operation of the matrix, the requirements of the feedforward neural network can be satisfied.

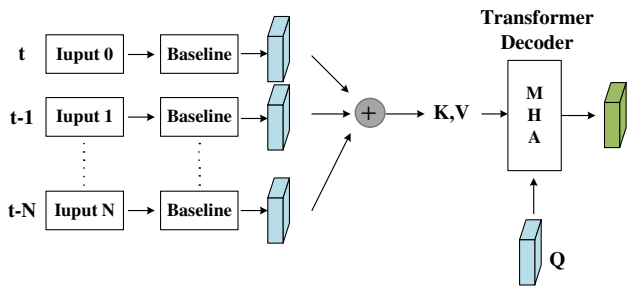
Besides, there is also a residual connection in the transformer's encoder. Since attention (Q, K, V) is consistent with the dimension of the input, we can directly add element-wise. In the later training process, the residual connection can be obtained by adding the values before and after the operation, which is convenient for the gradient to be directly transmitted to the initial layer.

Due to the adoption of multi-head attention mechanism, the model is divided into multiple heads, that is, the model can form multiple subspaces, which enables the model to pay attention to different aspects of information, so as to select meteorological elements more conducive to rainfall prediction and judgment (Diao et al., 2019; Xie et al., 2021).

3.2. Prediction model

Figure 4 shows the network structure of precipitation prediction model. The idea of the model is to use the model of the previous time period to predict the future time. For example, if there are N times of known monitoring information, we will use the information at $t-N$ to predict the information at $t-N+1$, and the corresponding K and V are calculated (Sriram et al., 2017). In the same way, we then predict the value at $t-N+2$ in terms of $(t-N,\ t-N+1)$. As shown in Figure 4, we can predict the future time according to the

Figure 4
Prediction model



monitoring data at $(t-N,\ldots,t)$, thus realizing the prediction of precipitation information.

The specific prediction process is as follows:

$$t-N \rightarrow t-N+1$$
$$(t-N, t-N+1) \rightarrow t-N+2$$
$$\cdots\cdots\cdots$$
$$(t-N, t-N+1, \ldots, t-1) \rightarrow t$$
$$(t-N, \ldots, t-1, t) \rightarrow t+1$$
$$\cdots\cdots\cdots$$

By using the historical observation information and historical numerical forecast information as the input of the model and adding the numerical forecast results in each training, the precipitation prediction error can be corrected. Moreover, by using historical observation information in the decoding stage, the model can learn the ability to be different from the numerical prediction results, so as to obtain more effective prediction results to a certain extent. The proposed sequential autoregressive model can take into account the randomness of heavy rainfall events and the regularity of precipitation forecasting at the same time by adding forecast results at each moment.

The structure of transformer decoder is shown in Figure 5. In order to solve the problem of gradient disappearance, the structure of residual network is adopted in both the encoder and decoder to include the original input and the output Z of the self-attention layer.

However, the decoder calculates the attention score from the output of the attention layer with its output and feeds the score into the feedforward neural network. Since transformer is not a sequential model, the speed of the model can be increased through parallel computation, but the location information will be lost.

In order to solve the above problems, transformer uses positional encoding. In most cases, since each time step still requires a unique positional encoding, one of

the most common approaches is to use trigonometric functions for positional encoding, which has the advantage of being able to extend to unknown sequence

lengths, and then convert the positional information into a vector. The formula for the position vector is as follows:

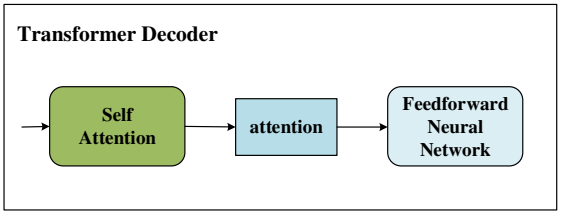
$$p_{2i} = \sin(u_i \cdot 2i) \tag{6}$$

$$p_{2i+1} = cos(u_i \cdot (2i+1)) \tag{7}$$

where p represents the position vector at the corresponding moment, and u_i represents the corresponding frequency:

$$u_i = \frac{1}{10000^{\frac{2i}{d}}} \tag{8}$$

Figure 5
Transformer decoder



3.3. Loss function

In the training process of the neural network, the error gradient represents the direction and size of the calculation and can be used to update the subsequent network weights to improve the performance of the network. However, as the number of network layers increases or cycles, the error gradient can accumulate and become large, which may cause drastic changes in the weights of the network, making the network unstable and possibly generating NaN values.

The traditional L1 loss, that is, the mean absolute error, can generate a stable gradient for any input, avoid the generation of gradient explosion, and have a stable solution. However, the function has a vertex at the center point and is difficult to derive, which makes it difficult to generate gradients and hinders the subsequent optimization of the network. Therefore, in order to make the network more robust, we choose Smooth-L1 as the loss function. The Smooth-L1 loss function smoothes the L1 function, and its specific formula is as follows:

$$smoothL1 = \begin{cases} 0.5x^2, if\ |x| < 1, \\ |x| - 0.5, otherwise. \end{cases} \tag{9}$$

It can be clearly seen from the above formula that if the predicted value is not much different from the actual value, the gradient value can be reduced by squaring, and the weight of the network is only slightly adjusted. If the predicted value is very different from the actual value, the gradient value will be directly subtracted by 0.5, which can ensure that the parameters of the network can be better adjusted. Unlike the L1 loss function that is not smooth enough, the Smooth-L1 loss function used here can avoid the generation of breakpoints. In addition, compared to the L2 function, the Smooth-L1 function is insensitive to outliers, that is, the overall performance does not change due to individual outliers.

4. Experiment

4.1. Dataset

In this paper, we conduct experiments using meteorological data released by the China Meteorological Administration to verify the effectiveness of our method Utrans-Net.

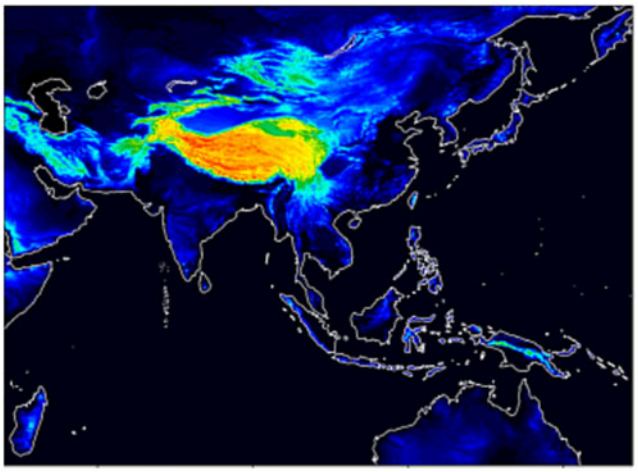
This dataset gives the numerical model (NWP) grid point data of a certain area, the size of the grid point area is 73×69, and each grid point contains the physical quantities of 23 meteorological elements, including rainfall, water vapor content, and temperature wait. The dataset contains a total of 2400 samples from different monitoring sites, each sample gives the monitoring data of 3 hours within 24 hours, and each monitoring site is processed as an NWP grid point data. The training set and test set can be divided according to certain requirements, and the target position to be predicted will be given.

Figure 6 presents a schematic representation of NWP, and the points that need to be predicted are randomly given in the figure. Based on the comparison of the predicted precipitation with the true value, the validity of the model can be verified.

4.2. Experiment setting

In our experiments, we use the threat score (TS) to evaluate the precipitation accuracy in our experiments, which is a method prescribed by the China Meteorological Administration for evaluating short-term forecast accuracy and forecasting ability (Wang, 2014). TS is calculated as follows:

Figure 6
The example of NWP



$$TS_1 = \frac{TP}{TP + TE + TN} \tag{10}$$

$$TS_2 = \frac{TP}{TP + TE + NE + NP} \tag{11}$$

where TP is the number of samples that correctly predict precipitation, TE is the number of samples that are predicted to have precipitation but actually no precipitation, NE is the number of samples that have no precipitation but actually have precipitation, and NP is the number of samples that are correctly predicted to have no precipitation. Therefore, TS_1 represents the proportion of accurately predicted precipitation, and TS_2 represents the proportion of correctly predicted precipitation. Since this paper mainly studies the accuracy of precipitation prediction, we will use the value of TS_1 by default.

In addition, we can set different thresholds for TS to evaluate the prediction accuracy of precipitation respectively, and the thresholds can be set to 0.1, 3.0, and 10.0 mm. By conducting experiments under different threshold conditions, we can verify the effect of our model at different thresholds. The settings of different thresholds can be applied to different application scenarios, such as military scenarios that are extremely sensitive to precipitation, and civilian scenarios that are not sensitive to precipitation. Our experiments are deployed on a server with an NVIDIA 1080Ti GPU.

4.3. Experimental comparison and analysis

To demonstrate the effectiveness of our method UTrans-Net, we compare it with other methods, including ECMWF (European Centre for Medium-Range Weather Forecasts, which is also an atmospheric model used by China), NCEP (weather used by the US Meteorological Center forecast model), and ARIMA (autoregressive integrated moving average model, which is a traditional time series model) (Zhang, 2003). Figure 7 shows an example heatmap of a precipitation prediction, where the dots represent a portion of the location that needs to be forecasted.

Table 1 shows the comparison of the prediction accuracy of several models in precipitation prediction under different thresholds. It can be found that ARIMA has the lowest prediction accuracy relative to other methods, while several other methods have no significant difference. Furthermore, our method presents

Figure 7
The example of precipitation prediction

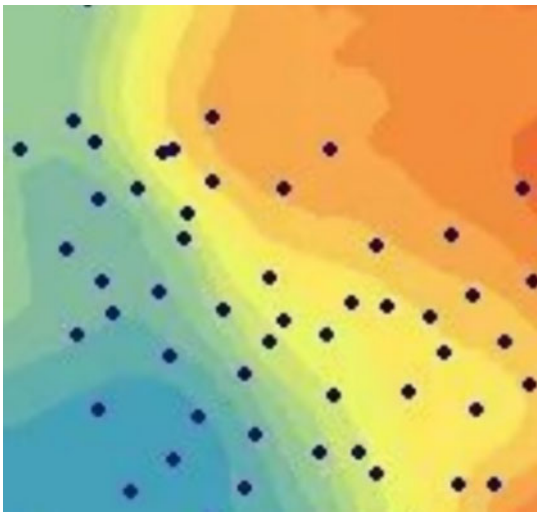


Table 1
Average prediction accuracy of precipitation prediction models

Method	0.1 mm	3 mm	10 mm
ECMWF	0.55	0.76	0.81
NCEP	0.53	0.73	0.79
3ARIMA	0.33	0.38	0.44
UTrans-Net	0.59	0.78	0.82

the best prediction results, with 7, 2, and 1% improvement under different thresholds, respectively. According to the results, under the condition of higher threshold, the improved performance becomes smaller, because as the threshold increases, the requirements for the accuracy rate of model prediction will also become lower accordingly, which also leads to the fact that most models can better meet forecast requirements. Therefore, our method is more suitable for scenarios that are more sensitive to precipitation prediction requirements.

4.4. Ablation experiment

In this paper, we design ablation experiments to verify the effectiveness of adding transformer. As can be seen from Table 2, if the transformer structure is not added, the prediction accuracy of the rainfall model will decrease.

By replacing the loss function with L1 and L2 functions, this paper also studies the role of the Smooth-L1 loss function in training the precipitation prediction model. As can be seen from Table 3, the accuracy of precipitation prediction can be slightly improved after using the Smooth-L1 loss function. Compared with the L1 loss function, the Smooth-L1 function after smoothing can

Table 2
Ablation experiment of transformer on precipitation prediction

Method	0.1 mm	3 mm	10 mm
Ours (without transformer)	0.52	0.73	0.75
Ours	0.59	0.78	0.82

Table 3
Ablation experiment of Smooth-L1 on precipitation prediction

Method	0.1 mm	3 mm	10 mm
L1	0.59	0.76	0.80
L2	0.58	0.73	0.79
Smooth-L1	0.59	0.78	0.82

handle the vertices existing in L1, which also enables Smooth-L1 to slightly improve the accuracy of precipitation prediction. In addition, the Smooth-L1 function can also accommodate some abnormal points in the data, avoiding the sudden drop of the gradient caused by the occasional abnormal data, which greatly affects the adjustment of network parameters, which can improve the fault tolerance rate of the entire model.

4.5. Discussion and analysis

Precipitation prediction has a wide range of application scenarios in real life and has a strong timeliness. Moreover, if it involves short-term precipitation prediction or long-term precipitation prediction, it is also necessary to screen the meteorological data sequence that has been monitored and to infer its evolution process as accurately as possible.

We need to evaluate our model Utrans-Net in terms of time consumption, including certain requirements in terms of training time and prediction time of the model. The model needs to train relevant parameters within a specified time and filter out more important meteorological elements. This can be met by a certain pre-trained model, so it is more necessary for the model to be able to give a prediction result through a certain monitoring data within a given time. In addition, since precipitation prediction is a time-sensitive event, it further requires that the model needs to be able to make predictions in a very short period of time after the station has monitored the required data, and our transformer structure happens to be able to perform well, increasing this demand. Moreover, our model can still make good predictions when the amount of data is insufficient or small.

5. Conclusion

Short-term weather forecasting is particularly important in daily life, and we study the precipitation events in this paper. In order to solve the problems of low accuracy and slow efficiency of existing precipitation prediction models, as well as the problem that traditional AI models cannot effectively mention the characteristics of meteorological elements, this paper proposes a U-shaped neural network model UTrans-Net using the transformer mechanism to extract features from multi-scale meteorological elements and use them for precipitation prediction. Specifically, we use the self-attention mechanism in transformer to assign weights to different meteorological elements. Then, we use the precipitation monitoring value and predicted value at the previous moment to predict the precipitation value at the next moment, so as to realize the precipitation forecast at the specified moment. Afterwards, we use the loss function Smooth-L1 to adjust the parameters of the network, which further improves the prediction accuracy of the model. The experimental results on meteorological data provided by the China Meteorological Administration prove that our method maintains higher prediction accuracy than other methods, and the average prediction accuracy at 0.1, 3, and 100 mm thresholds is 59%, 78%, and 82%, respectively. In the future, we strive to further improve our model to make it more generalized and able to predict more meteorological elements, including temperature, precipitation, and other important elements.

Acknowledgement

This work was supported in part by a grant from National Key R&D Program of China under Grant No.2021YFB3900601, the Fundamental Research Funds for the Central Universities under Grant B220202074, and Joint Fundation of the Ministry of Education under Grant No.8091B022123.

Conflicts of Interest

Yirui Wu is an associate editor for *Artificial Intelligence and Applications*, and was not involved in the editorial review or the decision to publish this article. The authors declare that they have no conflicts of interest to this work.

References

Ayzel, G., Heistermann, M., & Winterrath, T. (2019). Optical flow models as an open benchmark for radar-based precipitation nowcasting (rainymotion v0. 1). *Geoscientific Model Development*, *12*(4), 1387–1402. http://doi.org/10.5194/gmd-12-1387-2019

Bauer, P., Thorpe, A., & Brunet, G. (2015). The quiet revolution of numerical weather prediction. *Nature*, *525*(7567), 47–55. http://doi.org/10.1038/nature14956

Chen, L., Cao, Y., Ma, L., & Zhang, J. (2020). A deep learning-based methodology for precipitation nowcasting with radar. *Earth and Space Science*, *7*(2), e2019EA000812. http://doi.org/10.1029/2019EA000812

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. & Zhou, Y. (2021). Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint 2102.04306.

de Andrade, F. M., Young, M. P., MacLeod, D., Hirons, L. C., Woolnough, S. J., & Black, E. (2021). Subseasonal precipitation prediction for Africa: Forecast evaluation and sources of predictability. *Weather and Forecasting*, *36*(1), 265–284. http://doi.org/10.1175/WAF-D-20-0054.1

Diao, L., Niu, D., Zang, Z., & Chen, C. (2019). Short-term weather forecast based on wavelet denoising and catboost. In *2019 Chinese control conference*, 3760–3764.

Erol, B., Gurbuz, S. Z., & Amin, M. G. (2019). GAN-based synthetic radar micro-Doppler augmentations for improved human activity recognition. In *2019 IEEE Radar Conference*, 1–5.

Hewage, P., Behera, A., Trovati, M., Pereira, E., Ghahremani, M., Palmieri, F., & Liu, Y. (2020). Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Computing*, *24*(21), 16453–16482. http://doi.org/10.1007/s00500-020-04954-0

Jing, J. R., Li, Q., Ding, X. Y., Sun, N. L., Tang, R., & Cai, Y. L. (2019). Aenn: A generative adversarial neural network for weather radar echo extrapolation. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *42*, 89–94. http://doi.org/10.5194/isprs-archives-XLII-3-W9-89-2019

Jing, J., Li, Q., & Peng, X. (2019). MLC-LSTM: Exploiting the spatiotemporal correlation between multi-level weather radar echoes for echo sequence extrapolation. *Sensors*, *19*(18), 3988. http://doi.org/10.3390/s19183988

Luo, C., Li, X., Wen, Y., Ye, Y., & Zhang, X. (2021). A novel LSTM model with interaction dual attention for radar echo extrapolation. *Remote Sensing*, *13*(2), 164. http://doi.org/10.3390/rs13020164

McGovern, A., Elmore, K. L., Gagne II, D. J., Haupt, S. E., Karstens, C. D., Lagerquist, R., Smith, T. & Williams, J. K. (2017). Using artificial intelligence to improve real-time decision-making for high-impact weather. *Bulletin of the American Meteorological Society*, *98*(10), 2073–2090. http://doi.org/10.1175/BAMS-D-16-0123.1

Nguyen, D. H., Kim, J. B., & Bae, D. H. (2021). Improving radar-based rainfall forecasts by long short-term memory network in urban basins. *Water*, *13*(6), 776. http://doi.org/10.3390/w13060776

Pan, X., Lu, Y., Zhao, K., Huang, H., Wang, M., & Chen, H. (2021). Improving nowcasting of convective development by incorporating polarimetric radar variables into a deep-learning model. *Geophysical Research Letters*, *48*(21), e2021GL095302. http://doi.org/10.1029/2021GL095302

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 234–241.

Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., & Woo, W. C. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems*, 28.

Shi, E., Li, Q., Gu, D., & Zhao, Z. (2018). A method of weather radar echo extrapolation based on convolutional neural networks. In *International Conference on Multimedia Modeling*, 16–28.

Shi, X., & Yeung, D. Y. (2018). Machine learning for spatiotemporal sequence forecasting: A survey. *arXiv preprint: 1808.06865*.

Sriram, A., Jun, H., Satheesh, S., & Coates, A. (2017). Cold fusion: Training seq2seq models together with language models. *arXiv preprint: 1708.06426.*

Torcasio, R. C., Federico, S., Comellas Prat, A., Panegrossi, G., D'Adderio, L. P., & Dietrich, S. (2021). Impact of lightning data assimilation on the short-term precipitation forecast over the Central Mediterranean Sea. *Remote Sensing*, *13*(4), 682. http://doi.org/10.3390/rs13040682

Trebing, K., Stańczyk, T., & Mehrkanoon, S. (2021). SmaAt-UNet: Precipitation nowcasting using a small attention-unet architecture. *Pattern Recognition Letters*, *145*(2), 178–186. http://doi.org/10.1016/j.patrec.2021.01.036

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomes, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems 30*.

Wang, C. C. (2014). On the calculation and correction of equitable threat score for model quantitative precipitation forecasts for small verification areas: The example of Taiwan. *Weather and Forecasting*, *29*(4), 788–798. http://doi.org/10.1175/WAF-D-13-00087.1

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems 34*, 12077–12090.

Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, *50*, 159–175. https://doi.org/10.1016/S0925-2312(01)00702-0

Zhang, L., Huang, Z., Liu, W., Guo, Z., & Zhang, Z. (2021). Weather radar echo prediction method based on convolution neural network and long short-term memory networks for sustainable e-agriculture. *Journal of Cleaner Production*, *298*(8), 126776. http://doi.org/10.1016/j.jclepro.2021.126776

Zhang, P., Jia, Y., Gao, J., Song, W., & Leung, H. (2018). Short-term rainfall forecasting using multi-layer perceptron. *IEEE Transactions on Big Data*, *6*(1), 93–106. http://doi.org/10.1109/TBDATA.2018.2871151

Zhang, Y., Li, Y., & Zhang, G. (2020). Short-term wind power forecasting approach based on Seq2Seq model using NWP data. *Energy*, *213*(5552), 118371. http://doi.org/10.1016/j.energy.2020.118371

Zhang, P., Zhang, L., Leung, H., & Wang, J. (2017). A deep-learning based precipitation forecasting approach using multiple environmental factors. In *2017 IEEE International Congress on Big Data (BigData Congress)*, 193–200.

Zhao, Q., Liu, Y., Yao, W., & Yao, Y. (2021). Hourly rainfall forecast model using supervised learning algorithm. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–9. http://doi.org/10.1109/TGRS.2021.3054582

How to Cite: Cao, H., Wu, Y., Bao, Y., Feng, X., Wan, S., & Qian, C. (2023). UTrans-Net: A Model for Short-Term Precipitation Prediction. *Artificial Intelligence and Applications* 1(2), 106–113, https://doi.org/10.47852/bonviewAIA2202337