Synthesis-Style Pre-trained Auto-Correlation Transformer: A Zero-shot Learner on Long Ionospheric TEC Series Forecasting

Yuhuan Yuan[1], Guozhen Xia[2], Xinmiao Zhang[3], and Chen Zhou[2]

[1]Wuhan University
[2]Department of Space Physics, School of Electronic Information, Wuhan University, Wuhan, China
[3]wuhan university

March 13, 2023

Abstract

In this paper, we present a novel approach to improve the accuracy of TEC prediction through data augmentation. Prior works that adopt various deep-learning-based approaches suffer from two major problems. First, from a deep model perspective: LSTM models exhibit low performance on long-term data dependency, while self-attention-based methods ignore the temporal nature of time series, which results in an information utilization bottleneck. Second, the existing TEC actual data is limited and existing generative models fail to generate sufficient high-quality datasets. Our work leverages a two-stage deep learning framework for TEC prediction, stage 1: a time series generative model synthesis of sufficient data close to real data distribution, and stage 2: an Anto-correlation-based transformer to model temporal dependencies by presenting series-wise connections. Experiment on the 2018 TEC testing benchmark demonstrates that our method improves the accuracy by a large margin. The models trained on synthetic data had a notably lower RMSE of 1.17 TECU, while the RMSE for the IRI2016 model was 2.88 TECU. Our results show that the model significantly reduces monthly RMSE, displaying higher reliability in mid, high, low latitudes. Our model shows higher reliability and significantly reduces monthly RMSE and latitude RMSE. However, although our model performs better than IRI2016, low latitudes RMSE needs improvement, as values are generally above 2.5 TECU. This finding has important implications for the development of advanced TEC prediction models and highlights the potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

Synthesis-Style Pre-trained Auto-Correlation Transformer: A Zero-shot Learner on Long Ionospheric TEC Series Forecasting

Yuhuan Yuan[1],Guozhen Xia[1],Xinmiao Zhang[1], Chen Zhou[1]

[1]Department of Space Physics, Wuhan University, Wuhan, China

Key Points:

- TEC Data augmentation: synthesizing TEC samples by feeding selected original TEC map datasets into a variational auto-encoder model.
- Pre-train auto-correlation-based transformer and Transformer models using the imitation samples without any further action on fine-tuning.
- Improved the accuracy of the predictive auto-correlation-based transformer models through data augmentation.

Corresponding author: Chen Zhou, `chenzhou@whu.edu.cn`

Abstract

In this paper, we present a novel approach to improve the accuracy of TEC prediction through data augmentation. Prior works that adopt various deep-learning-based approaches suffer from two major problems. First, from a deep model perspective: LSTM models exhibit low performance on long-term data dependency, while self-attention-based methods ignore the temporal nature of time series, which results in an information utilization bottleneck. Second, the existing TEC actual data is limited and existing generative models fail to generate sufficient high-quality datasets. Our work leverages a two-stage deep learning framework for TEC prediction, stage 1: a time series generative model synthesis of sufficient data close to real data distribution, and stage 2: an Anto-correlation-based transformer to model temporal dependencies by presenting series-wise connections. Experiment on the 2018 TEC testing benchmark demonstrates that our method improves the accuracy by a large margin. The models trained on synthetic data had a notably lower RMSE of 1.17 TECU, while the RMSE for the IRI2016 model was 2.88 TECU. Our results show that the model significantly reduces monthly RMSE, displaying higher reliability in mid, high, low latitudes. Our model shows higher reliability and significantly reduces monthly RMSE and latitude RMSE. However, although our model performs better than IRI2016, low latitudes RMSE needs improvement, as values are generally above 2.5 TECU. This finding has important implications for the development of advanced TEC prediction models and highlights the potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

Plain Language Summary

In this paper, we tackle the challenge of accurately predicting the changes in the Ionospheric total electron content, which is a critical aspect of the Earth's space environment affecting communication and satellite positioning. To achieve this, we generate additional TEC datasets that allow the model to better capture the underlying patterns in the TEC data, and build an Anto-correlation-based transformer to model the temporal dependencies by presenting series-wise connections. The results demonstrate that our proposed model is highly effective in predicting TEC on a global scale compared with the Transformer model and IRI2016 model.

1 Introduction

Ionospheric total electron content (TEC) is one of the significant elements among STEC (The slant total electron content which refers to the total number of electrons along a path between the radio transmitter to the receiver) for Global Navigation Satellite Service (GNSS), GPS signal propagation and applications, and their applications. Additionally, L1 frequency acts as marginal sensitivity for 1 TECU causing a 0.163 range delay (Lastovicka et al., 2017). Industrial applications rely on good modeling and prediction of TEC including satellite navigation (Ratnam et al., 2018), precise point positioning (Prol et al., 2018; Z. Li et al., 2019), and time-frequency transmission (Béniguel & Hamel, 2011). For the above, despite modeling long-term dependency for TEC is hard, researchers in different societies i.e. space physics and remote sensing proposed various works of literature for TEC forecasting(Feng et al., 2019).

Recently there are mainly two directions of work for forecasting global TEC maps by the learning-based method. One direction works by following the pipeline that first predicts the spherical harmonic (SH) coefficients and then expands them to complete TEC maps. For example, (C. Wang et al., 2018) proposed an adaptive autoregressive model to predict the SH coefficients used in TEC map fitting, while (Iyer & Mahajan, 2023) uses both linear and polynomial autoregression coefficients of recent past data to forecast TEC over equatorial regions. (Liu et al., 2022) adopt a long short-term memory (LSTM) network to forecast the SH coefficient to further predict the TEC maps. In (C. Wang et al., 2018) (SH) coefficients are predicted based on the autoregressive model, and the order of the autoregressive model is determined adaptively using the F-test method.

Another stream of work lies in forecasting a sequence of global TEC maps following past given TEC maps without introducing any prior information. (Monte-Moreno et al., 2022) uses a nearest-neighbor algorithm to search the historical database for the dates of the maps closest to the current map and uses a prediction of the maps in the database. (Liu et al., 2020) adopt a convolutional neural network to extract features from past TEC maps, then predict the future TEC maps based on the extracted features. (Q. Li et al., 2022; Chen et al., 2019; Yang & Liu, 2022) proposes a generative adversarial network for TEC forecasting, which compose a generator to generate maps that are indistinguishable from real TEC maps and a discriminator trying to distinguish between the generated maps and real maps. This deep learning method can generate satisfactory ionospheric peak structures at different times and geomagnetic conditions and can be used to predict the regional TEC over China two hours in advance(Q. Li et al., 2022). (H. Wang et al., 2022; X. Lin et al., 2022) adopt the spatiotemporal network model as a source for forecasting Total Electron Content (TEC) maps, this model is used to correct ionospheric delay and improve the accuracy of satellite navigation positioning, and forecast TEC at a global scale 24 hours in advance(Cesaroni et al., 2020). LSTM can also as an end-to-end TEC forecasting model, (Xia, Zhang, et al., 2022; Cherrier et al., 2017), near real-time TEC maps can be provided no more than 5 minutes after the observation time(Mendoza et al., 2019), and these maps can be used to estimate the GPS signal delay due to the ionospheric electron content between a receiver and a GPS satellite. The recent transformer-based method(M. Lin et al., 2022) uses the self-attention mechanism of the transformer structure is utilized to capture the long-term characteristics of the TEC in China.

However, despite flourishing progress in the deep model for TEC forecasting, there are still challenges remaining. From the data perspective: First, to train a very deep model, for example, (Vaswani et al., 2017) needs a large-scale training dataset, and insufficient training data always causes over-fitting and further leads to lower performance on out-of-distribution testing samples. Second, VAE as a usual backbone for anomaly detection(Ha & Schmidhuber, 2018; Desai et al., 2021) scenarios has better abilities at synthesizing exceptional cases or creating datasets for cases such as the presence of outliers of change-points are necessary. From the backbone prediction model perspective: 1. recent RNN and LSTM-based model(Ruwali et al., 2020; Liu et al., 2022) exhibit unsatisfactory performance on modeling TEC maps' long-term dependency, gradients of RNN models propagated over many stages tend to either vanish or explode so that the distance between relevant information and the point where it is needed becomes very large, and the capacity of LSTM is limited that each unit of memory can affect every other unit in the memory with a learnable weight, this results in a number of learnable parameters in the model grow quadratically with the memory size, e.g. an LSTM with a memory of size 64KB results in parameters of size 8GB. 2. Although transformer-based method(Xia, Liu, et al., 2022) adopting point-wise self-attention module can model long-term dependency without regard for the distance in either input or output sequences, point-wise self-attention only calculating the relation between scattered points lead to ignorance of the temporal series dependencies, further causes information utilization bottleneck. We therefore ask, can we design a generative module such that we can synthesize inexhaustible samples that are high-quality enough to be regarded as "equal" as possible to a real distribution dataset? And can we design a prediction model which is expert in modeling both long-term dependencies and temporal series dependencies for long-time TEC series forecasting? And ultimately, is pretrianed-model strong enough to outperform the over-fitting deep model on the TEC training set even when zero-shot?

In our work, we proposed a novel two-stage approach for the TEC maps forecasting method by leveraging a generative model(Desai et al., 2021) with auto-correlation transformer network(Vaswani et al., 2017) as the prediction model. In the first stage, The VAE model captures both the distribution of the features and the temporal relationships in the data to generate the imitation samples. In the second stage, we use the auto-correlation transformer network as the prediction model to forecast the TEC maps. The auto-correlation transformer decomposes the time series into its trend-cyclical part and seasonal part to capture complex temporal patterns in long-context forecasting. The pre-trained auto-correlation transformer shows its robustness by outperforming other deep-learning models that suffer from overfitting. We summarize our contributions as follows:

1. Firstly, by using the VAE model to synthesize imitation samples, we solve the dilemma of the insufficient high-quality training dataset for TEC forecasting.

2. By using the auto-correlation transformer, our approach captures the complex temporal patterns in the TEC maps data, leading to more accurate forecasting results.

3. By pre-training the auto-correlation transformer on the imitation samples, our approach improves the robustness and reduces overfitting, leading to better performance in the zero-shot testing scenario.

The paper is organized as follows. The data source and preprocessing method are described in 3.1. The concrete method description is located at 2. The numerical experiment details, results, and analysis are demonstrated in 4. Finally, 5 exhibits the conclusions, discussion, and future directions.

2 Methods

Our two-stage deep learning method mainly includes two steps. First, we synthesize the sample efficiency by feeding the selected original TEC map dataset into a variational auto-encoder(VAE) model(Desai et al., 2021). Second, we pre-train the auto-correlation-based transformer using the imitation samples without any further action on finetuning, and the empirical reference International Reference Ionosphere 2016 model (IRI2016) and 1-day BUAA model developed by (C. Wang et al., 2018) are chosen as the comparison model. In this section, we demonstrate the architecture of our generation model and prediction model, as well as their training processes. The pipeline of our method is shown in Figure 1.

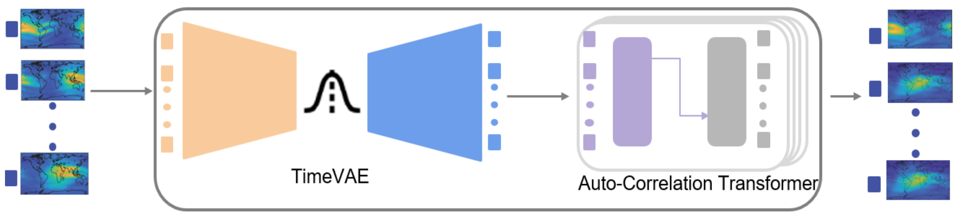

Figure 1. Overview of our pipeline. We introduce a two-stage synthesis and auto-correlation method for TEC maps forecasting. The generation model TimeVAE takes the selected original real dataset as input and captures both the distributions in features as well as the temporal relationships to synthesize generated dataset. The prediction model is an auto-correlation-based transformer that decomposes the series to learn complex temporal patterns in long-context forecasting. The pre-trained model shows its robustness by outperforming overfitting deep models in a zero-shot testing manner.

Compared to RNN and LSTM, the Transformer and Auto-correlation-based Transformer models have a lower computational complexity $O(n^2d)$, where n is the smaller sequence length and d is the dimensionality. Thus, we chose to utilize these models instead of RNN and LSTM models to achieve lower computational complexity. Considering sample efficiency, we generated the same amount of data as the original data, as we considered it to be an important factor. Additionally, we have also implemented data augmentation in our study, by generating original data twice. Instead of attempting to solve the model accuracy problem by generating an infinite amount of data, generating twice as much data gives us an attempt to improve the accuracy of the model on the data, and it turns out that this works actually. Therefore, an infinite multifold generation of data is not necessary and twice is enough for us.

2.1 Generative model: TimeVAE

TimeVAE Training Dataset. We consider each hourly TEC dataset to be an independent and identically distributed set of samples. The inputs consist of N i.i.d. samples, where N represents the total number of hours in the TEC dataset. The spatial longitude ranges from 180° west to 180° east with a resolution of 5° and the latitude ranges from 87.5° north to 87.5° south with a resolution of 2.5°. As a result, the global TEC map grid consists of 71 x 73 points, with 71 and 73 representing the latitude and longitude information of the TEC map at each hour, respectively, corresponding to different geographical locations. The structure of the generation model is shown in Figure 2, where the input dataset array, represented as (N, 71, 73), is a 3-dimensional array. The latitude and longitude information of the TEC map at each hour, represented by 71 and 73, respectively, correspond to different geographic locations, while N represents the total number of samples.

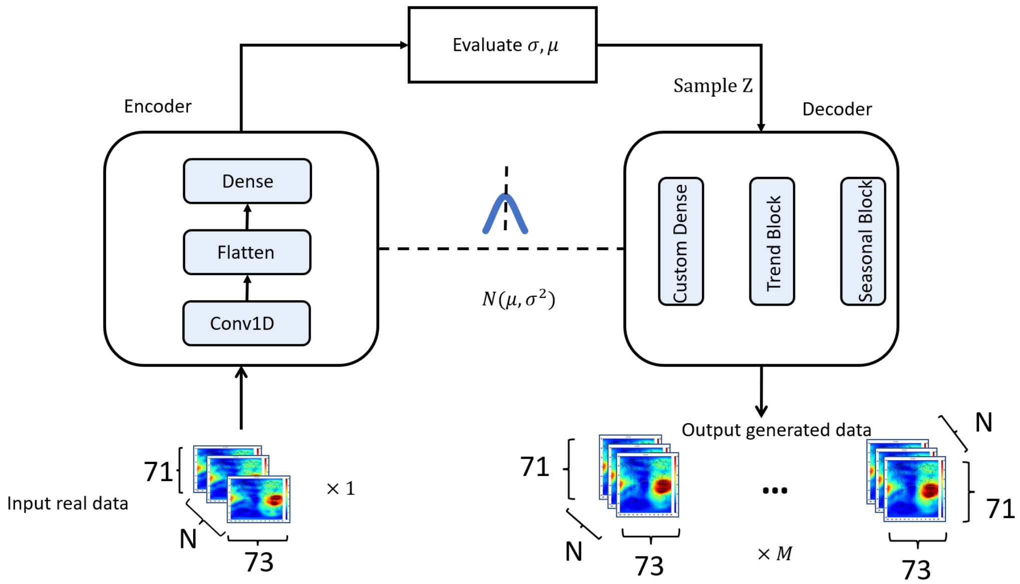

Figure 2. The architecture of TimeVAE

TimeVAE Architecture. To adapt the generation model to the synthesis of ionospheric TEC maps, we adopted an encoder-decoder VAE model. The encoder is to extract the feature of the input i.e. a 3-dimensional array of size $N \times t \times D$, N for batch size, T for the number of time steps, and D for the number of feature dimensions, into a multivariate Gaussian distribution by passing the inputs through a series of convolutional layers with ReLU activation and a fully-connected linear layer. The encoder outputs the parameters of the multivariate Gaussian which can be used to sample the latent vector z using the reparameterization trick. by taking the latent state vector z from the multivariate Gaussian, The decoder first passes the latent vector through a fully-connected linear layer, then reshapes the data into a 3-dimensional array, and passes it through a series of transposed convolutional layers with ReLU activation. Finally, the data is passed through a time-distributed fully-connected layer to produce the final output, which should have the same shape as the original TEC map signal. The goal of the decoder is to generate TEC maps that are as similar as possible to the original TEC maps, based on the information encoded in the latent vector "z".

TimeVAE Loss Function. We train TimeVAE using the Evidence Lower Bound loss(ELBO) function, which is written following:

$$ELBO = E_{q(\boldsymbol{z}|\boldsymbol{x};\phi)}\left[\log(\boldsymbol{x} \mid \boldsymbol{z};\theta)\right] - D_{KL}\left(q(\boldsymbol{z} \mid \boldsymbol{x};\phi) \| p(\boldsymbol{z};\theta)\right) \tag{1}$$

The process of BLBO loss actually is to reconstruct $\boldsymbol{x}$ given $\boldsymbol{z}$ sampled from $q(\boldsymbol{z} \mid \boldsymbol{x};\phi)$. Specifically, the Right Hand Side is composed of two parts, and the first term is the log-likelihood of our data given $\boldsymbol{z}$ sampled from $q(\boldsymbol{z} \mid \boldsymbol{x};\phi)$. The second term is the KL-Divergence loss between the encoded latent space distribution $q(\boldsymbol{z} \mid \boldsymbol{x};\phi)$ and the prior distribution $p(\boldsymbol{z};\theta)$.

2.2 Auto-correlation Based Transformer Architecture

Modeling long-term time series forecasting for TEC maps is not easy: we have to handle the intricate temporal patterns. The original transformer architecture(Vaswani et al., 2017) adopts self-attention modules to calculate the correlation between scattered points but ignores the dependencies among sub-series. In contrast, our approach leverages an auto-correlation-based transformer(Wu et al., 2021) as a prediction model which enables series-wise connections to model dependencies between each sub-series and raises the information utilization. The architecture of the auto-correlation-based transformer is shown in Figure 3.

Series Decomposition Block inherit the ideas from (Cleveland et al., 1990) separate the long time series into two parts: trend-cyclical part and seasonal part, for the former reflects overall trend and fluctuations and the latter reflects the repeating patterns(seasonality) of the series respectively. The series decomposition block is deployed along the model as it goes deeper to capture complex patterns. For the input, TEC maps series $X \in R^{I \times d}$, where I is the length of the input series, and d is the number of features of the TEC map series. The concrete inner operation(Cleveland et al., 1990) for gathering two composed series is:

$$X_t = AvgPool(Padding(X)) \; and \; X_s = X - X_t \tag{2}$$

where X_t is the trend-clynical part, and X_s is the seasonal part. Within the inner operation, AvgPool($\cdot$) is a moving average pooling with the padding operation to keep the series length unchanged.

Auto-correlation mechanism The auto-correlation mechanism is designed based on the periodicity of the time series and aims to conduct the discovery and representation aggregation of dependencies at the sub-series level. The calculation of auto-correlation involves shifting the time series by a certain tag and computing the correlation between the original time series and the shifted time series. From stochastic process theory autocorrelation $R_{xx}(\tau)$ is a time-delay similarity between original series $\{X_t\}$ and its τ lagged series $\{X_{t-\tau}\}$ which can be calculated by:

$$R_{xx}(\tau) = \lim_{L\to\infty} \sum_{t=1}^{L} X_t X_{t-\tau} \tag{3}$$

In the real-world application for TEC map predictions, we first project the embedding of TEC maps to get query Q, key K. A time delay aggregation block is then applied to

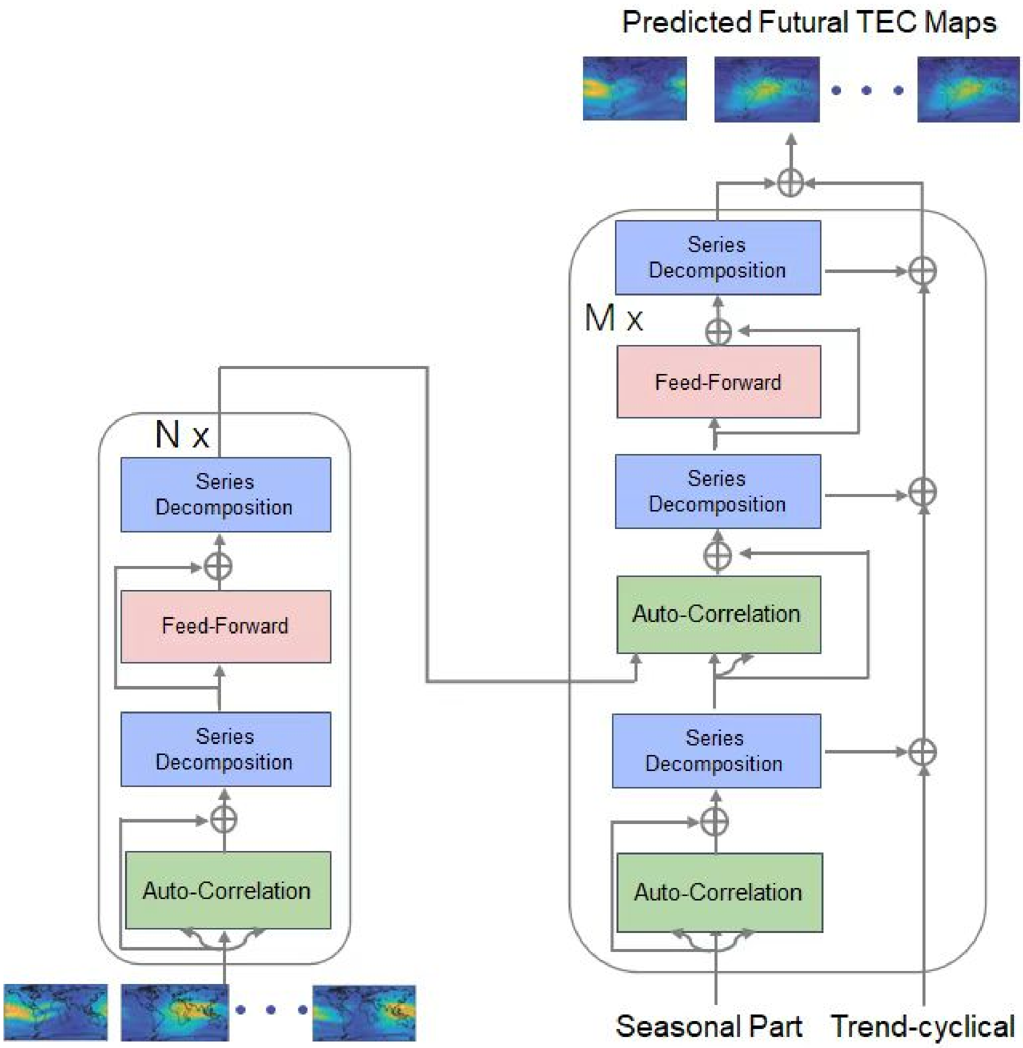

Figure 3. The architecture of auto-correlation-based transformer.

roll the series based on the selected time delay, then we are able to aggregate the sub-series by softmax. Concretely we first get the arguments of TopK autocorrelations:

$$\tau_1...\tau_k = argTopk_{\tau \in \{1,...,L\}}(R_{Q,K}(\tau)) \tag{4}$$

$R_{Q,K}$ is the autocorrelation between Q and K. After that, the series are fed into a soft-max layer:

$$R_{Q,\hat{K}}(\tau_1), ..., R_{Q,\hat{K}}(\tau_k) = SoftMax(R_{Q,K}(\tau_1)...R_{Q,K}(\tau_k)) \tag{5}$$

Then the Auto-correlation can be obtained by the relationship between the series and its time-delay shifted version. $Roll(X, \tau)$ the operation to shift the series $\{X_t\}$ with time delay τ. The expression of auto-correlation is:

$$Auto\_correlation(Q, K, V) = \sum_{i=1}^{k} Roll(V, \tau_i)(\hat{R}_{Q,K}(\tau_i)) \tag{6}$$

2.3 Transformer Architecture

The Transformer model consists of an encoder and a decoder. The encoder takes an input sequence and generates a sequence of hidden states, while the decoder takes the encoder output and generates a sequence of output tokens. Both the encoder and decoder consist of multiple layers of self-attention and feedforward neural networks. The self-attention mechanism allows the model to attend to different parts of the input sequence, while the feedforward neural networks enable the model to capture complex patterns in the data. We directly quote the model-building method mentioned in the article (Vaswani et al., 2017) to build the Transformer model.

Encoder: The encoder is composed of two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. The residual connection is around the two sub-layers, followed by layer normalization.

Decoder: The decoder is also composed of a stack of $N = 6$ identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Similar to the encoder, we employ residual connections around each of the sub-layers, followed by layer normalization. We also modify the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions. This masking, combined with the fact that the output embeddings are offset by one position, ensures that the predictions for position i can depend only on the known outputs at positions less than i.

Multi-headed Self-Attention (MSA): Multi-head self-attention allows the model to jointly attend to information from different representation subspaces at different positions. With a single attention head, averaging inhibits this. The basic structure is the same as self-attention, but when operating, it is divided into multiple heads, and then the self-attention calculation is carried out in parallel, and then the output vectors are finally spliced together, where different heads will learn different levels of knowledge.

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^O \tag{7}$$

Where $head = Attention(QW_i^Q, KW_i^K, VW_i^V)$, and the projections are parameter matrices $W_i^Q \in R^{d_{\text{model}} \times d_k}, W_i^K \in R^{d_{\text{model}} \times d_k}, W_i^V \in R^{d_{\text{model}} \times d_v}$ and $W^O \in R^{hd_v \times d_{\text{model}}} \cdot$.

Positional Encoding This part is used to inject some information about the relative or absolute position tokens from the TEC sequence. tokens in the sequence. To this end, we add "positional encodings" at the bottoms of the encoder and decoder stacks.

The positional encodings have the same dimension d_{model} as the embeddings so that the two can be summed. There are many choices of positional encodings, learned and fixed [9]. In this work, we use sine and cosine functions of different frequencies:

$$\begin{aligned} PE_{(pos,2i)} &= \sin\left(pos/10000^{2i/d_{\text{model}}}\right) \\ PE_{(pos,2i+1)} &= \cos\left(pos/10000^{2i/d_{\text{model}}}\right) \end{aligned} \tag{8}$$

where pos is the position and i is the dimension. That is, each dimension of the positional encoding corresponds to a sinusoid.

3 EXPERIMENTS

Our aim is to achieve two primary objectives. Firstly, we employ TimeVAE to enhance our capability of producing credible TEC time-series dataset samples when we have a shortage of actual TEC datasets for model training. Secondly, we seek to train advanced time forecasting models, specifically the Transformer and Auto-Correlation Transformer, using these generated samples. These two models, Transformer and Auto-Correlation Transformer are the most advanced models in long-series time forecasting. Finally, we will evaluate the accuracy of both prediction models and present our findings. To achieve this, we have prepared three different training datasets names ori_data, gen_once, and gen_twice. The advantage of TimeVAE is that it learns the distribution of TEC datasets, allowing us to generate multiple TEC datasets. This enables us to produce theoretically unlimited datasets even in the absence of sufficient real data.

3.1 Data Source and Processing

In this paper, we use the dataset called global ionospheric TEC data, which is generated by the standard Ionosphere map exchange format (IONEX) file format which is provided by International GNSS Service (IGS). TECU stands for "Total Electron Content Unit" and is a unit of measurement for the ionosphere TEC. TECU is defined as the number of free electrons that would be present in a one square meter column of unit cross section extending from the Earth's surface to the top of the ionosphere if all the free electrons were concentrated in a single point. We conduct experiments with three distinct training datasets, namely ori_data, gen_once, and gen_twice. Here is the description of the experiment datasets:

1) **ori_data:** The global ionospheric TEC data is generated by the standard Ionosphere map exchange format file format which is provided by IGS. We processed datasets into a 4-dimensional time series sequence $(number, 24, 71, 73)$. The longitude dimension consists of 73 points, spanning from 180° west longitude to 180° east longitude with a resolution of 5°. The latitude dimension consists of 71 points, ranging from 87.5° north latitude to 87.5° south latitude with a resolution of 2.5°. The scale of the global TEC map grid points is 71 × 73. 24 means the total amount hourly for the time resolution of 1 hr each day, and *number* represents the total number of days in the time range from January 1, 1998 to December 31, 2017.
2) **gen_once:** We generated synthetic data as the second dataset, which has been confirmed to have a high similarity to the original dataset ori_data with an identical size.
3) **gen_twice:** We generated ori_data twice repeatedly, each time we obtained an identical dataset with ori_data. Finally, the gen_data dataset consists of two synthetic data, which are twice the size of the original data ori_data.

During the real training model process, 90% of each of the above mentioned datasets are utilized for training while the remaining 10% is used for validation purposes.

3.2 Generated Samples

T-SNE (t-Distributed Stochastic Neighbor Embedding) is a machine learning algorithm for non-linear dimensional reduction developed by Laurens van der Maaten and Geoffrey Hinton(Van der Maaten & Hinton, 2008). It is used to reduce high-dimensional data into a lower-dimensional space for visualization or machine learning. t-SNE works on the principle of minimizing the divergence between a distribution of actual data points in high dimensional space and a distribution of corresponding points in a lower dimensional space. This is done by mapping the data points to a probability distribution in the lower dimensional space.

We use these comparison metrics proposed in (Yoon et al., 2019) to assess the quality of the synthetic data. The lower the discriminative score, the better performance. The abscissa (x-axis) and ordinate (y-axis) of t-SNE represent the positions of data points in a low-dimensional space. The t-SNE algorithm is designed to map high-dimensional data to a low-dimensional space, typically two or three dimensions, for visualization purposes. In t-SNE, each data point in the input dataset is represented as a point in the low-dimensional space, and the similarity between data points in the high-dimensional space is preserved as the distance or closeness between points in the low-dimensional space. The discriminator score of generating the TEC dataset is 0.0035±0.007, which demonstrates the high correlation between synthetic data and original data and TimeVAE performs well in generating synthetic data. Figure 4 displays the t-SNE charts of data generated from the generator TimeVAE. The TimeVAE generated data consistently shows heavy overlap with original data.

Figure 5 and figure 6 present the comparison of the TEC provided by IGS and generated by the model TimeVAE at six randomly selected time points. The top row displays TEC maps provided by IGS, while the bottom row shows TEC maps generated by the TimeVAE model. Six randomly selected time points are labeled at the top of each map, highlighting the comparison between the two sources. The comparison showcases the accuracy of the TimeVAE model in TEC mapping, as well as its potential for use in ionosphere research and satellite-based communication systems.

3.3 Forecasting deep learning models.

Evaluation metric. Root-mean-square error (RMSE) and percentage deviation (PD) following are used to estimate the forecasting performance of the model. The lower the RMSE value, the better the model's accuracy in prediction. In essence, RMSE represents the average magnitude of the errors in the predictions made by a model. The percentage deviation score is calculated by taking the absolute difference between the predicted value and the actual value, divided by the actual value, and multiplied by 100.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(TEC_{\text{ori}} - TEC_{\text{pred}}\right)^2} \tag{9}$$

$$PD = \frac{1}{N}\sum_{i=1}^{N}\frac{|TEC_{\text{ori}} - TEC_{\text{pred}}|}{TEC_{\text{ori}}} \tag{10}$$

where N is the total number of data samples, TEC_{ori} and TEC_{pred} are the observed value and forecasting value, respectively.

4 Forecasting Models Performance and Results Analysis

Our study trains two models, Auto-correlation-based Transformer and Transformer models, using three different datasets: original data, once-generated synthetic data, and

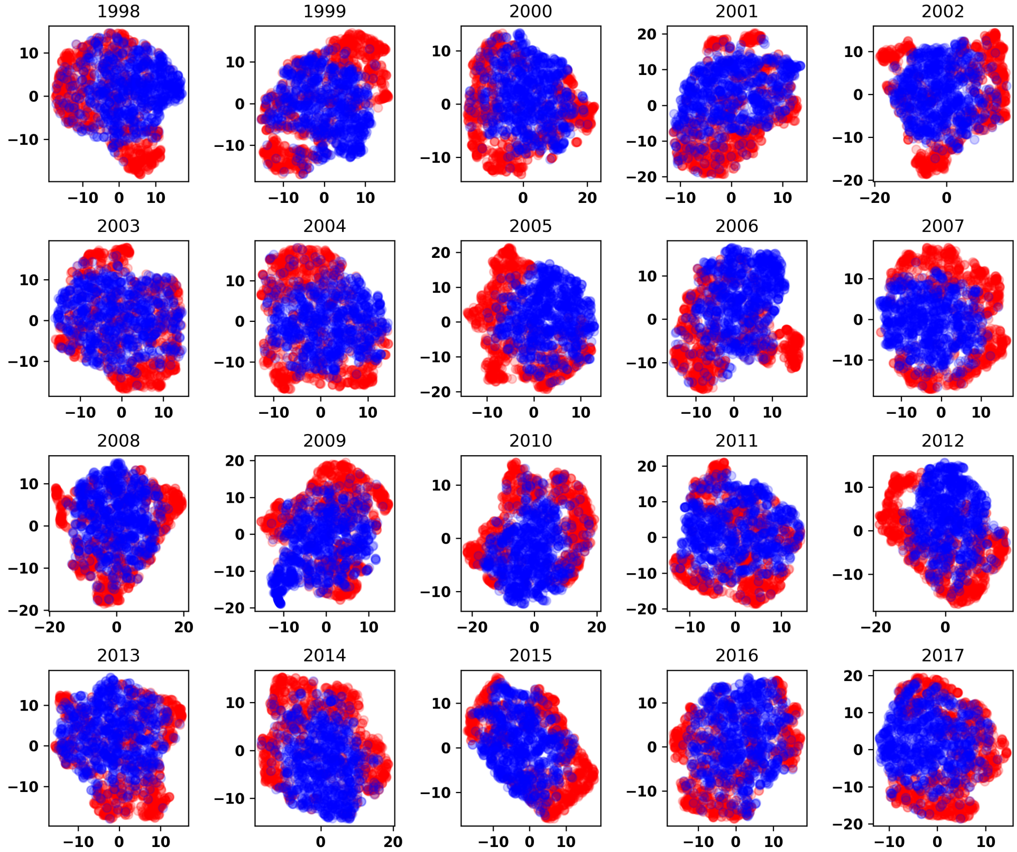

Figure 4. The visual t-SNE plots of generated and original data from 1998 to 2017. Red is for original data and blue is for synthetic data. Higher overlap rates represent higher similarity. The x-axis and y-axis of t-SNE represent the positions of data points($number, 24, 73, 71$) in a two-dimensional space.

twice-generated synthetic data (ori_data, gen_once, and gen_twice). We evaluate the Root Mean Squared Error (RMSE) of the total of six trained models in the years 2018, 2019, and 2020, as well as the RMSE of the IRI2016 model in 2018. Furthermore, we assess the RMSE and percentage deviation (PD) scores of the models in high, low, and middle latitudes to compare their predictive capabilities.

4.1 Traning Set and Testing set Results Analysis

In table 1, we compared the RMSE of IRI2016, 1-day BUAA, Transformer, and Auto-correlation-based transformer models during 2018. We find that our Transformer based models to be superior to the IRI2016 model and 1-day BUAA model. for example, the RMSE of the Auto-correlation-based transformer model is 1.55 TECU, which is 0.52 TECU less than 1-day BUAA and 1.33 TECU less than IRI2016 model. As presented in Table 2, the RMSE scores of all models (Auto-correlation-based Transformer, Transformer) trained on different datasets demonstrate that our models trained using synthetic data outperform the IRI2016 model and 1-day BUAA model. Additionally, it is observed that the more synthetic data used for model training, the better the model performance. For

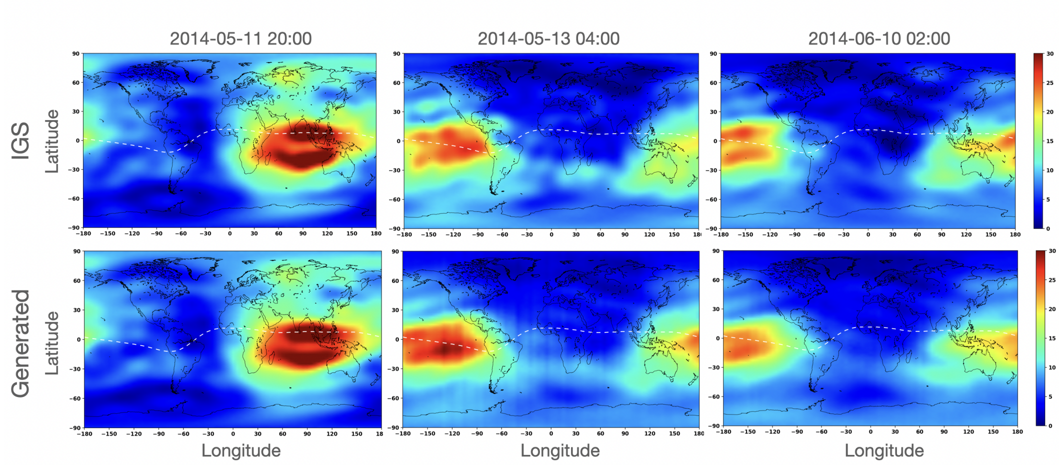

Figure 5. Comparison between the global total electron content map provided by International GNSS Service and generated by TimeVAE model for three stochastic times.

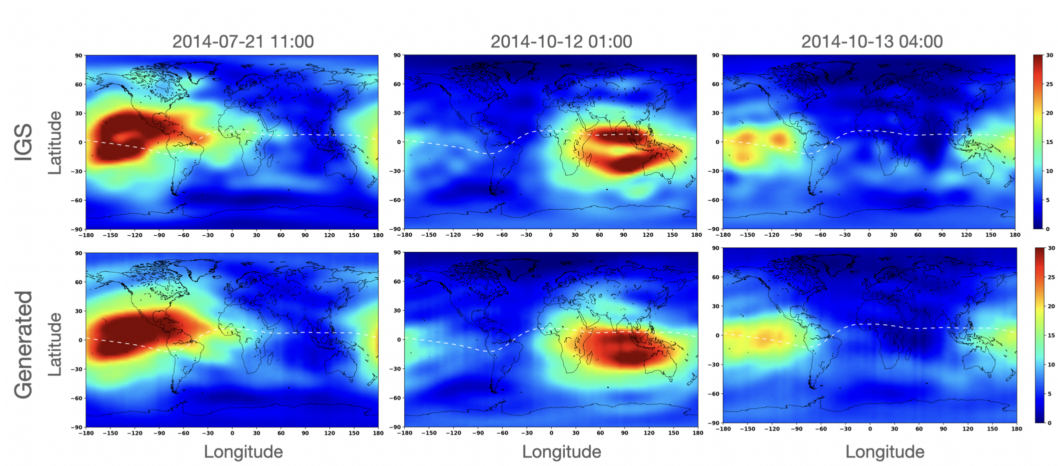

Figure 6. Comparison between the global total electron content map provided by International GNSS Service and generated by TimeVAE model for three stochastic time.

instance, the Auto-correlation-based Transformer trained using twice-generated synthetic data increases the RMSE of the IRI2016 model from 2.88 to 1.17 in 2018. In comparison to the Auto-correlation-based Transformer trained with once-generated synthetic data, the model trained with twice-generated synthetic data has a better performance with RMSE scores of 0.86 and 1.21 in 2019 and 2020 respectively. The Auto-correlation-based Transformer model trained with gen_twice data outperforms the other six models in forecasting TEC from 2018 to 2020. Compared to the model trained with original data, the Auto-correlation-based Transformer model trained with generated data significantly improves its RMSE from 1.55 to 1.17 in 2018, 1.48 to 0.86 in 2019, and 1.58 to 1.21 in 2020. Additionally, the Auto-correlation-based Transformer model demonstrates better performance than the Transformer model in certain years from 2018 to 2020.

Table 1. The accuracy RMSE scores on four different training models (IRI2016 model, 1-day BUAA, Auto-correlation-based transformer, Transformer) during 2018.

Models	IRI2016 model	1-day BUAA	Auto-correlation-based transformer	Transformer
RMSE	2.88	2.07	1.55	1.37

Table 2. Multivariate accuracy RMSE scores on three different training datasets with the pre-trained models. Training datasets include ori_data, gen_once, and gen_twice, and ori_data means IGS TEC datasets from 1998 to 2017, gen_once means the generated datasets from 1998 to 2017 and gen_twice means generated dataset twice from 1998 to 2017. The best performances are in bold, a lower RMSE score indicates a better prediction.

Models	Training datasets	RMSE (TECU)		
		2018	2019	2020
Anto-correlation-based transformerr	ori_data	1.55	1.48	1.58
	gen_once	1.22	0.91	1.25
	gen_twice	**1.17**	**0.86**	**1.21**
Transformer	ori_data	1.37	1.08	1.96
	gen_once	1.31	1.00	1.75
	gen_twice	1.29	1.00	1.73

4.2 Comparison of Predictional TEC Maps

We train the Auto-correlation-based transformer model and the Transformer model on multivariate training datasets and also evaluated their monthly root mean squared error (RMSE) during 2018 and 2020. As shown in Fig 8, we plotted the predicted TEC map and the original TEC map for January 3, 2020, with the top half coming from the original IGS data and the bottom half coming from our best-performing model Autoformer-correlation-based transformer. Fig 7 shows the TEC map on Five points in time on January 3, 2019, which also compared the original real TEC map with the predicted TEC map. We assessed the monthly RMSE of the IRI2016 model in 2018 and compared it with the original data trained Transformer model and the original data trained Auto-correlation-based Transformer model in figure 9. The results indicate that both the Auto-correlation-based Transformer model and the Transformer model offer improved predictive capabilities when compared to the IRI2016 model.

Figure 10 displays the monthly root mean squared error (RMSE) of the Auto-correlation-based Transformer model and the Transformer model trained on three different datasets in 2018. A_ori_data, A_gen_once and A_gen_twice represent Auto-correlation-based Transformer model trained on ori_data, gen_once and gen_twice respectively. And T_ori_data, T_gen_once, and T_gen_twice represent the Transformer model trained on ori_data, gen_once, and gen_twice respectively. The results presented in Figure 10 demonstrate that the model trained with twice the amount of synthetic data performed better than the model trained with only the original data in 2018. Specifically, the Auto-correlation-based transformer model trained with gen_twice performed more accurately than other models trained with only original data or gen_once for all 12 months in 2018, with the best predictions occurring in June and October, and the biggest RMSE value being 1.175 occurring in month 1. The worst model is the Transformer model trained by original data with an RMSE over 1.5 TECU but less than 2.0 TECU in every month of 2018.

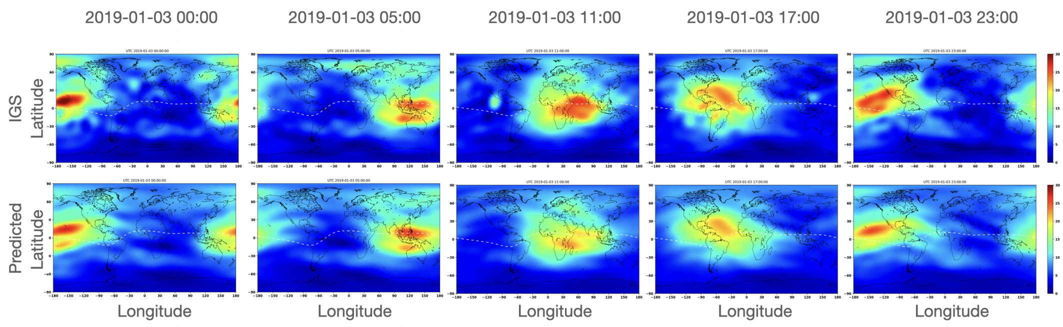

Figure 7. The comparison between the TEC map of Auto-correlation-based transformer trained by gen_twice dataset and the TEC map of the real IGS on January 3, 2019.

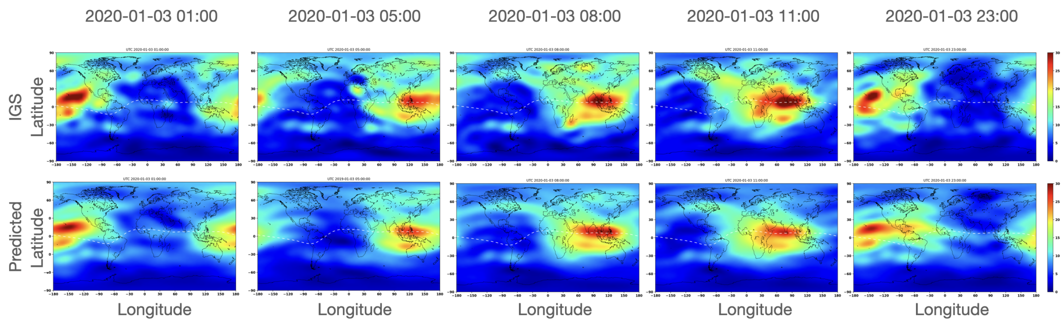

Figure 8. The comparison between the TEC map of Auto-correlation-based transformer trained by gen_twice dataset and the TEC map of the real IGS on January 3, 2020.

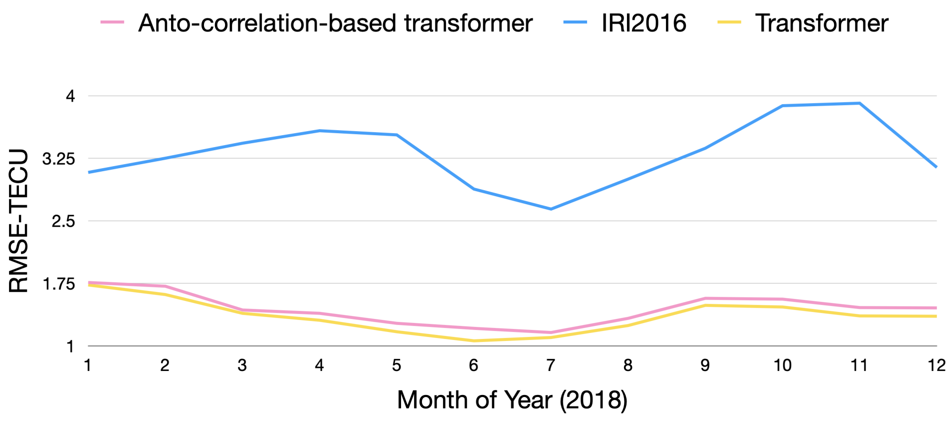

Figure 9. The monthly averaged RMSE scores of two different models(Auto-Correlation Transformer, Transformer) compared with the IRI2016 model with the same training datasets during 2018.

The results from the comparison of prediction accuracy in 2020, as shown in Figure 11, indicates that the Transformer model trained on the gen_twice dataset achieved the highest accuracy in predicting monthly TEC values. The RMSE values for all three models varied between approximately 1.4 TECU and 2.5 TECU throughout the year, with the best predictions occurring in October and November and the worst predictions occurring in July and August. In contrast, the Transformer model trained on the ori_data is expected to perform poorly in every month of 2020 when compared to the other two dataset training models, having a high level of RMSE error. Additionally, in each month of 2020, the performance of the Transformer model trained on the gen_twice dataset was slightly better than the model trained on the gen_once dataset. Overall, the Transformer models performed best in October and November, with the lowest RMSE values among the 12 months in 2020.

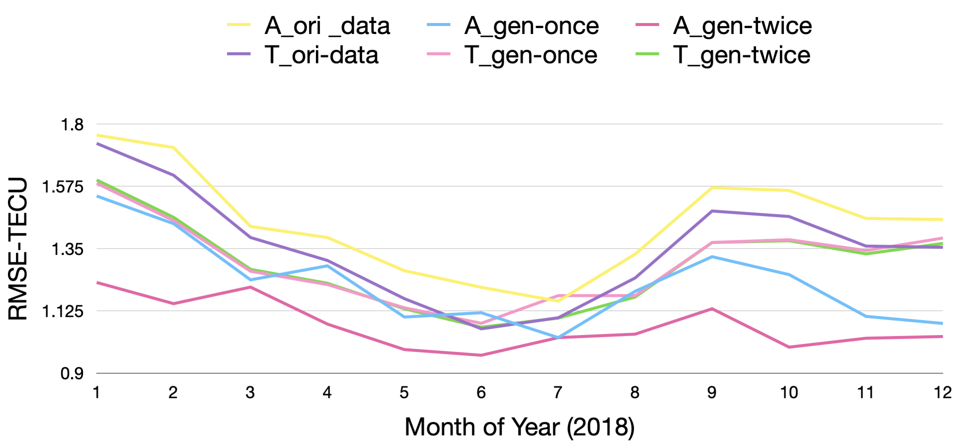

Figure 10. The monthly average RMSE scores for the Auto-Correlation Transformer models and Transformer models trained on three different datasets (ori_data, gen_once, gen_twice) during 2018. The labels for the figures begin with the letter 'A' to represent the Auto-correlation-based transformer model, while the letter 'T' is used to represent the Transformer model.

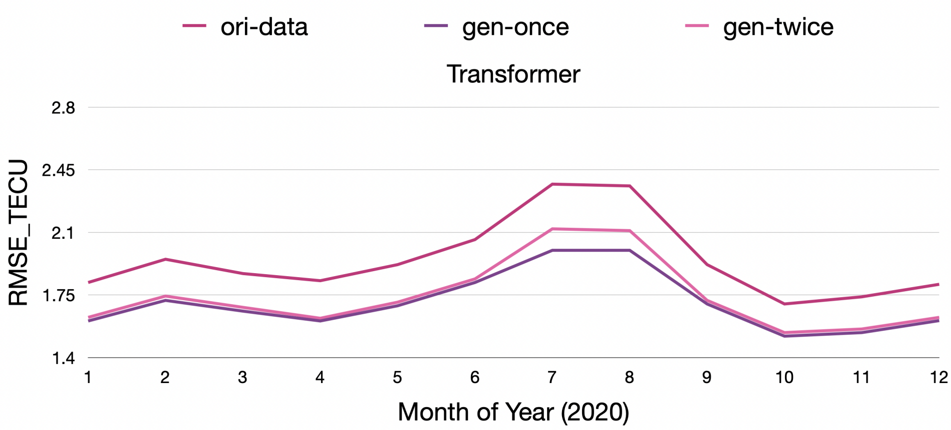

Figure 11. The monthly RMSE values of the Tramsformer model trained on 3 dataset (ori_data, gen_once, and gen_twice) in 2020.

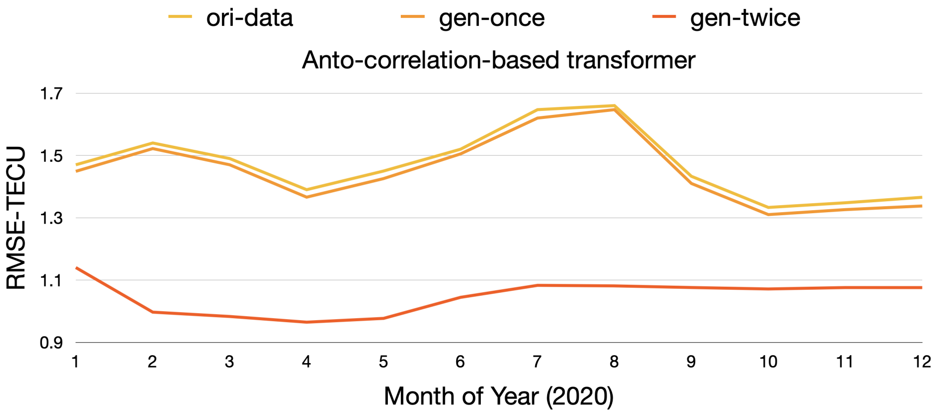

Figure 12. The comparison of monthly RMSE values of the Auto-correlation-based transformer models trained on 3 datasets (ori_data, gen_once, and gen_twice) in 2020.

The comparison of the prediction accuracy of the Auto-correlation-based transformer model in 2020 on three different training datasets in figure 12 shows that, similar to that of the Transformer model, the Auto-correlation-based transformer model trained on the gen_twice dataset has the highest accuracy in predicting monthly TEC values. The RMSE values of the three models fluctuate between approximately 0.9 and 1.3 TECU throughout the year, with the best predictions in April and March. On the other hand, the Auto-correlation-based transformer model trained on the ori_data performed the worst in every month of 2020, with slightly lower performance than gen_once every month, with the RMSE values fluctuating between approximately 1.3 and 1.7 TECU throughout the year.

4.3 Latitude Results Analysis

Figure 13 illustrates the comparison of RMSE scores in different latitudes of two different models trained on three datasets (ori_data, gen_once, gen_twice) during 2018, The first three models are Auto-correlation-based transformer models trained on ori_data, gen_once, and gen_twice, while the next three are Transformer models trained on the same three datasets. The last model represented is the IRI2016 model. Our analysis reveals that all six training models outperform the IRI2016 model, as they exhibit lower RMSE scores in high, low, and middle latitudes. In low latitudes, we have achieved a significant improvement in RMSE from around 4 TECU to approximately 2.5 TECU. Out of the six training models, five of them achieved a lower RMSE score of about 2.5 TECU in low latitude, with the remaining Auto-correlation-based transformer model trained on twice synthetic data presenting the lowest RMSE score of around 0.8 TECU. The RMSE scores of the six training models in high and middle latitudes are consistently around 1 TECU. It is noteworthy that the accuracy in high latitudes tends to be lower compared to that in middle latitudes for each of the models. The percentage deviation score is a measure of the deviation of a model's predictions from the actual values in percentage. A lower percentage deviation score indicates more accurate predictions. Figure 14 displays these results. As can be seen, all of the models perform best in the middle latitude, with a percentage deviation score of around 8% to 10%. This suggests that the models are generally more accurate in this latitude range Furthermore, Figure 14 indicates that the Auto-correlation-based Transformer model performed best in the middle latitude, with a percentage deviation score of 8.4%. In contrast, the Transformer model did not perform as well, with percentage deviation scores in the high latitude exceeding 17%. These

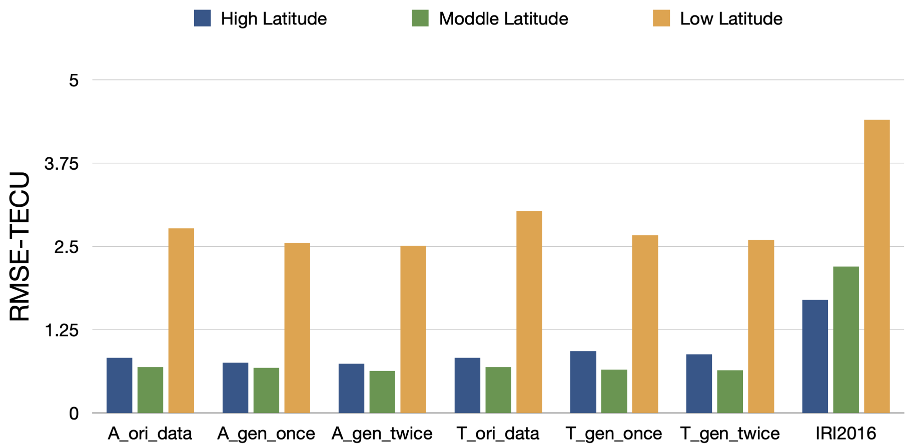

Figure 13. The averaged RMSE scores of all models with three different latitudes (high latitude, middle latitude, low latitude) during 2018.

results demonstrate that the Auto-correlation-based Transformer model may be a better choice for predicting TEC values in certain latitude ranges.

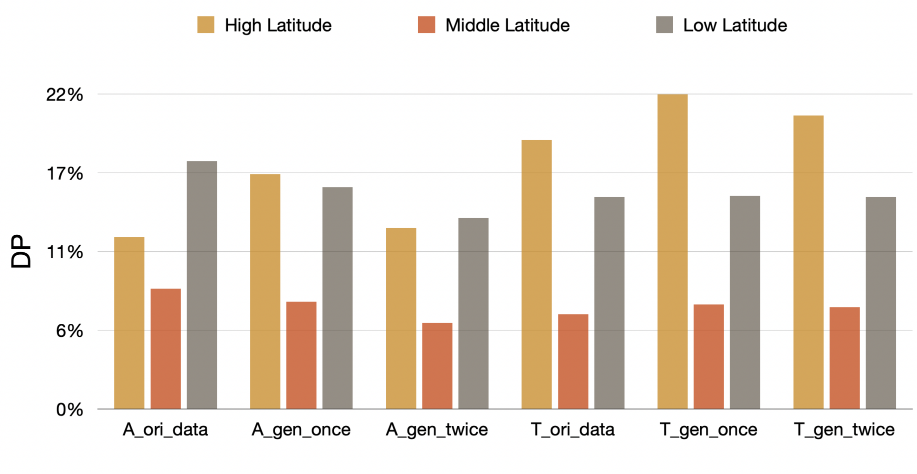

Figure 14. The DP scores of all models with three different latitudes (high latitude, middle latitude, low latitude) during 2018.

5 Conclusion

The Total Electron Content of the Earth's ionosphere plays a critical role in the satellite-based communications system. In this study, we compare the performance of two transformer models trained on synthetic data with the widely used IRI2016 model and BUAA model in TEC prediction. We applied data enhancement to TEC forecasting and found that it was effective in improving the quality of predictions.

Our model has shown significant improvement in predicting TEC at mid and high latitudes. However, at low latitudes, the RMSE accuracy of our model is still above the desired value of 2.5 TECU, indicating the need for further improvement in this area. On the other hand, both the monthly RMSE and Latitude RMSE of our model outperforms the IRI2016 model. Our results show that the transformer model trained on synthetic data achieves higher accuracy and reliability in TEC prediction, with a significant RMSE reduction of 1.17 compared to 2.88 for the IRI2016 model. This finding has important implications for the development of advanced TEC prediction models and highlights the potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

Furthermore, when tested on 2018 prediction data, the Auto-correlation-based transformer model using synthetic data exhibits an RMSE reduction of 0.38 TECU compared to the model trained on original data. We demonstrate that utilizing synthetic data can effectively enhance the prediction efficiency of the model, providing another avenue for enhancing the model's accuracy. Moreover, with the same amount of data, models trained on synthetic data offer more accurate predictions than those solely using actual data. The use of synthetic data has several contributions to the study. First, it can effectively enhance the prediction efficiency of the model, improving the accuracy of TEC prediction, but would gradually saturate as the amount of data increases. Second, it offers a way to generate more data without the need for additional data collection, which is particularly useful in cases where obtaining real data is difficult or expensive. Third, the study shows that models trained on synthetic data can outperform those trained on real data in terms of accuracy, indicating that synthetic data can provide a valuable alternative for training machine learning models. The results demonstrate the great potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

6 Open Research

The TEC GIMs provided by IGS is available from the website at `"http://pub.ionosphere.cn/products/daily/"` and the BUAA 1-day prediction data can be obtained from the website (`"http://pub.ionosphere.cn/prediction/daily/"`).

References

Béniguel, Y., & Hamel, P. (2011). A global ionosphere scintillation propagation model for equatorial regions. *Journal of Space Weather and Space Climate*, *1*(1), A04.

Cesaroni, C., Spogli, L., Aragon-Angel, A., Fiocca, M., Dear, V., De Franceschi, G., & Romano, V. (2020). Neural network based model for global total electron content forecasting. *Journal of space weather and space climate*, *10*, 11.

Chen, Z., Jin, M., Deng, Y., Wang, J.-S., Huang, H., Deng, X., & Huang, C.-M. (2019). Improvement of a deep learning algorithm for total electron content maps: Image completion. *Journal of Geophysical Research: Space Physics*, *124*(1), 790–800.

Cherrier, N., Castaings, T., & Boulch, A. (2017). Forecasting ionospheric total electron content maps with deep neural networks. In *Proc. conf. big data space (bids), esa workshop.*

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). Stl: A seasonal-trend decomposition. *J. Off. Stat*, *6*(1), 3–73.

Desai, A., Freeman, C., Wang, Z., & Beaver, I. (2021). Timevae: A variational auto-encoder for multivariate time series generation. *arXiv preprint arXiv:2111.08095*.

Feng, J., Han, B., Zhao, Z., & Wang, Z. (2019). A new global total electron content

empirical model. *Remote Sensing*, *11*(6), 706.
Ha, D., & Schmidhuber, J. (2018). World models. *arXiv preprint arXiv:1803.10122*.
Iyer, S., & Mahajan, A. (2023). Short-term adaptive forecast model for tec over equatorial low latitude region. *Dynamics of Atmospheres and Oceans*, *101*, 101347.
Lastovicka, J., Urbar, J., & Kozubek, M. (2017). Long-term trends in the total electron content. *Geophysical Research Letters*, *44*(16), 8168–8172.
Li, Q., Yang, D., & Fang, H. (2022). Two hours ahead prediction of the tec over china using a deep learning method. *Universe*, *8*(8), 405.
Li, Z., Wang, N., Wang, L., Liu, A., Yuan, H., & Zhang, K. (2019). Regional ionospheric tec modeling based on a two-layer spherical harmonic approximation for real-time single-frequency ppp. *Journal of Geodesy*, *93*, 1659–1671.
Lin, M., Zhu, X., Tu, G., & Chen, X. (2022). Optimal transformer modeling by space embedding for ionospheric total electron content prediction. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1–14.
Lin, X., Wang, H., Zhang, Q., Yao, C., Chen, C., Cheng, L., & Li, Z. (2022). A spatiotemporal network model for global ionospheric tec forecasting. *Remote Sensing*, *14*(7), 1717.
Liu, L., Morton, Y. J., & Liu, Y. (2022). Ml prediction of global ionospheric tec maps. *Space Weather*, *20*(9), e2022SW003135.
Liu, L., Zou, S., Yao, Y., & Wang, Z. (2020). Forecasting global ionospheric total electron content (tec) using deep learning. In *Agu fall meeting abstracts* (Vol. 2020, pp. NG004–0017).
Mendoza, L. P. O., Meza, A. M., & Aragón Paz, J. M. (2019). A multi-gnss, multifrequency, and near-real-time ionospheric tec monitoring system for south america. *Space Weather*, *17*(5), 654–661.
Monte-Moreno, E., Yang, H., & Hernández-Pajares, M. (2022). Forecast of the global tec by nearest neighbour technique. *Remote Sensing*, *14*(6), 1361.
Prol, F. d. S., Camargo, P. d. O., Monico, J. F. G., & Muella, M. T. d. A. H. (2018). Assessment of a tec calibration procedure by single-frequency ppp. *GPS Solutions*, *22*, 1–11.
Ratnam, D. V., Vishnu, T. R., & Harsha, P. B. S. (2018). Ionospheric gradients estimation and analysis of s-band navigation signals for navic system. *IEEE Access*, *6*, 66954–66962.
Ruwali, A., Kumar, A. S., Prakash, K. B., Sivavaraprasad, G., & Ratnam, D. V. (2020). Implementation of hybrid deep learning model (lstm-cnn) for ionospheric tec forecasting using gps data. *IEEE Geoscience and Remote Sensing Letters*, *18*(6), 1004–1008.
Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, *9*(11).
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.
Wang, C., Xin, S., Liu, X., Shi, C., & Fan, L. (2018). Prediction of global ionospheric vtec maps using an adaptive autoregressive model. *Earth, Planets and Space*, *70*(1), 1–14.
Wang, H., Lin, X., Zhang, Q., Chen, C., Cheng, L., & Wang, Z. (2022). Global ionospheric total electron content prediction based on spatiotemporal network model. In *China satellite navigation conference (csnc 2022) proceedings: Volume ii* (pp. 153–162).
Wu, H., Xu, J., Wang, J., & Long, M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, *34*, 22419–22430.
Xia, G., Liu, M., Zhang, F., & Zhou, C. (2022). Caitst: Conv-attentional image time sequence transformer for ionospheric tec maps forecast. *Remote Sensing*,

14 (17), 4223.
Xia, G., Zhang, F., Wang, C., & Zhou, C. (2022). Ed-convlstm: A novel global ionospheric total electron content medium-term forecast model. *Space Weather*, *20* (8), e2021SW002959.
Yang, K., & Liu, Y. (2022). Global ionospheric total electron content completion with a gan-based deep learning framework. *Remote Sensing*, *14* (23), 6059.
Yoon, J., Jarrett, D., & Van der Schaar, M. (2019). Time-series generative adversarial networks. *Advances in neural information processing systems*, *32*.

Synthesis-Style Pre-trained Auto-Correlation Transformer: A Zero-shot Learner on Long Ionospheric TEC Series Forecasting

Yuhuan Yuan[1],Guozhen Xia[1],Xinmiao Zhang[1], Chen Zhou[1]

[1]Department of Space Physics, Wuhan University, Wuhan, China

Key Points:

- TEC Data augmentation: synthesizing TEC samples by feeding selected original TEC map datasets into a variational auto-encoder model.
- Pre-train auto-correlation-based transformer and Transformer models using the imitation samples without any further action on fine-tuning.
- Improved the accuracy of the predictive auto-correlation-based transformer models through data augmentation.

Corresponding author: Chen Zhou, chenzhou@whu.edu.cn

Abstract

In this paper, we present a novel approach to improve the accuracy of TEC prediction through data augmentation. Prior works that adopt various deep-learning-based approaches suffer from two major problems. First, from a deep model perspective: LSTM models exhibit low performance on long-term data dependency, while self-attention-based methods ignore the temporal nature of time series, which results in an information utilization bottleneck. Second, the existing TEC actual data is limited and existing generative models fail to generate sufficient high-quality datasets. Our work leverages a two-stage deep learning framework for TEC prediction, stage 1: a time series generative model synthesis of sufficient data close to real data distribution, and stage 2: an Anto-correlation-based transformer to model temporal dependencies by presenting series-wise connections. Experiment on the 2018 TEC testing benchmark demonstrates that our method improves the accuracy by a large margin. The models trained on synthetic data had a notably lower RMSE of 1.17 TECU, while the RMSE for the IRI2016 model was 2.88 TECU. Our results show that the model significantly reduces monthly RMSE, displaying higher reliability in mid, high, low latitudes. Our model shows higher reliability and significantly reduces monthly RMSE and latitude RMSE. However, although our model performs better than IRI2016, low latitudes RMSE needs improvement, as values are generally above 2.5 TECU. This finding has important implications for the development of advanced TEC prediction models and highlights the potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

Plain Language Summary

In this paper, we tackle the challenge of accurately predicting the changes in the Ionospheric total electron content, which is a critical aspect of the Earth's space environment affecting communication and satellite positioning. To achieve this, we generate additional TEC datasets that allow the model to better capture the underlying patterns in the TEC data, and build an Anto-correlation-based transformer to model the temporal dependencies by presenting series-wise connections. The results demonstrate that our proposed model is highly effective in predicting TEC on a global scale compared with the Transformer model and IRI2016 model.

1 Introduction

Ionospheric total electron content (TEC) is one of the significant elements among STEC (The slant total electron content which refers to the total number of electrons along a path between the radio transmitter to the receiver) for Global Navigation Satellite Service (GNSS), GPS signal propagation and applications, and their applications. Additionally, L1 frequency acts as marginal sensitivity for 1 TECU causing a 0.163 range delay (Lastovicka et al., 2017). Industrial applications rely on good modeling and prediction of TEC including satellite navigation (Ratnam et al., 2018), precise point positioning (Prol et al., 2018; Z. Li et al., 2019), and time-frequency transmission (Béniguel & Hamel, 2011). For the above, despite modeling long-term dependency for TEC is hard, researchers in different societies i.e. space physics and remote sensing proposed various works of literature for TEC forecasting(Feng et al., 2019).

Recently there are mainly two directions of work for forecasting global TEC maps by the learning-based method. One direction works by following the pipeline that first predicts the spherical harmonic (SH) coefficients and then expands them to complete TEC maps. For example, (C. Wang et al., 2018) proposed an adaptive autoregressive model to predict the SH coefficients used in TEC map fitting, while (Iyer & Mahajan, 2023) uses both linear and polynomial autoregression coefficients of recent past data to forecast TEC over equatorial regions. (Liu et al., 2022) adopt a long short-term memory (LSTM) network to forecast the SH coefficient to further predict the TEC maps. In (C. Wang et al., 2018) (SH) coefficients are predicted based on the autoregressive model, and the order of the autoregressive model is determined adaptively using the F-test method.

Another stream of work lies in forecasting a sequence of global TEC maps following past given TEC maps without introducing any prior information. (Monte-Moreno et al., 2022) uses a nearest-neighbor algorithm to search the historical database for the dates of the maps closest to the current map and uses a prediction of the maps in the database. (Liu et al., 2020) adopt a convolutional neural network to extract features from past TEC maps, then predict the future TEC maps based on the extracted features. (Q. Li et al., 2022; Chen et al., 2019; Yang & Liu, 2022) proposes a generative adversarial network for TEC forecasting, which compose a generator to generate maps that are indistinguishable from real TEC maps and a discriminator trying to distinguish between the generated maps and real maps. This deep learning method can generate satisfactory ionospheric peak structures at different times and geomagnetic conditions and can be used to predict the regional TEC over China two hours in advance(Q. Li et al., 2022). (H. Wang et al., 2022; X. Lin et al., 2022) adopt the spatiotemporal network model as a source for forecasting Total Electron Content (TEC) maps, this model is used to correct ionospheric delay and improve the accuracy of satellite navigation positioning, and forecast TEC at a global scale 24 hours in advance(Cesaroni et al., 2020). LSTM can also as an end-to-end TEC forecasting model, (Xia, Zhang, et al., 2022; Cherrier et al., 2017), near real-time TEC maps can be provided no more than 5 minutes after the observation time(Mendoza et al., 2019), and these maps can be used to estimate the GPS signal delay due to the ionospheric electron content between a receiver and a GPS satellite. The recent transformer-based method(M. Lin et al., 2022) uses the self-attention mechanism of the transformer structure is utilized to capture the long-term characteristics of the TEC in China.

However, despite flourishing progress in the deep model for TEC forecasting, there are still challenges remaining. From the data perspective: First, to train a very deep model, for example, (Vaswani et al., 2017) needs a large-scale training dataset, and insufficient training data always causes over-fitting and further leads to lower performance on out-of-distribution testing samples. Second, VAE as a usual backbone for anomaly detection(Ha & Schmidhuber, 2018; Desai et al., 2021) scenarios has better abilities at synthesizing exceptional cases or creating datasets for cases such as the presence of outliers of change-points are necessary. From the backbone prediction model perspective: 1. recent RNN and LSTM-based model(Ruwali et al., 2020; Liu et al., 2022) exhibit unsatisfactory performance on modeling TEC maps' long-term dependency, gradients of RNN models propagated over many stages tend to either vanish or explode so that the distance between relevant information and the point where it is needed becomes very large, and the capacity of LSTM is limited that each unit of memory can affect every other unit in the memory with a learnable weight, this results in a number of learnable parameters in the model grow quadratically with the memory size, e.g. an LSTM with a memory of size 64KB results in parameters of size 8GB. 2. Although transformer-based method(Xia, Liu, et al., 2022) adopting point-wise self-attention module can model long-term dependency without regard for the distance in either input or output sequences, point-wise self-attention only calculating the relation between scattered points lead to ignorance of the temporal series dependencies, further causes information utilization bottleneck. We therefore ask, can we design a generative module such that we can synthesize inexhaustible samples that are high-quality enough to be regarded as "equal" as possible to a real distribution dataset? And can we design a prediction model which is expert in modeling both long-term dependencies and temporal series dependencies for long-time TEC series forecasting? And ultimately, is pretrianed-model strong enough to outperform the over-fitting deep model on the TEC training set even when zero-shot?

In our work, we proposed a novel two-stage approach for the TEC maps forecasting method by leveraging a generative model(Desai et al., 2021) with auto-correlation transformer network(Vaswani et al., 2017) as the prediction model. In the first stage, The VAE model captures both the distribution of the features and the temporal relationships in the data to generate the imitation samples. In the second stage, we use the auto-correlation transformer network as the prediction model to forecast the TEC maps. The auto-correlation transformer decomposes the time series into its trend-cyclical part and seasonal part to capture complex temporal patterns in long-context forecasting. The pre-trained auto-correlation transformer shows its robustness by outperforming other deep-learning models that suffer from overfitting. We summarize our contributions as follows:

1. Firstly, by using the VAE model to synthesize imitation samples, we solve the dilemma of the insufficient high-quality training dataset for TEC forecasting.

2. By using the auto-correlation transformer, our approach captures the complex temporal patterns in the TEC maps data, leading to more accurate forecasting results.

3. By pre-training the auto-correlation transformer on the imitation samples, our approach improves the robustness and reduces overfitting, leading to better performance in the zero-shot testing scenario.

The paper is organized as follows. The data source and preprocessing method are described in 3.1. The concrete method description is located at 2. The numerical experiment details, results, and analysis are demonstrated in 4. Finally, 5 exhibits the conclusions, discussion, and future directions.

2 Methods

Our two-stage deep learning method mainly includes two steps. First, we synthesize the sample efficiency by feeding the selected original TEC map dataset into a variational auto-encoder(VAE) model(Desai et al., 2021). Second, we pre-train the auto-correlation-based transformer using the imitation samples without any further action on finetuning, and the empirical reference International Reference Ionosphere 2016 model (IRI2016) and 1-day BUAA model developed by (C. Wang et al., 2018) are chosen as the comparison model. In this section, we demonstrate the architecture of our generation model and prediction model, as well as their training processes. The pipeline of our method is shown in Figure 1.

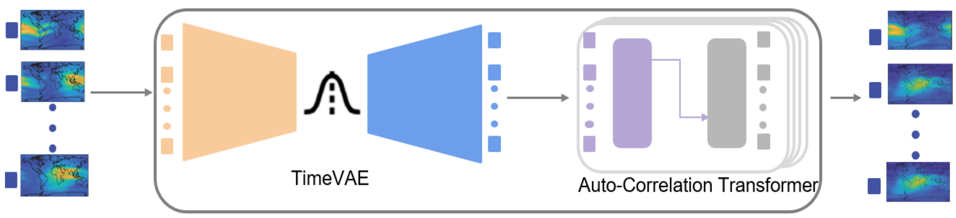

Figure 1. Overview of our pipeline. We introduce a two-stage synthesis and auto-correlation method for TEC maps forecasting. The generation model TimeVAE takes the selected original real dataset as input and captures both the distributions in features as well as the temporal relationships to synthesize generated dataset. The prediction model is an auto-correlation-based transformer that decomposes the series to learn complex temporal patterns in long-context forecasting. The pre-trained model shows its robustness by outperforming overfitting deep models in a zero-shot testing manner.

Compared to RNN and LSTM, the Transformer and Auto-correlation-based Transformer models have a lower computational complexity $O(n^2d)$, where n is the smaller sequence length and d is the dimensionality. Thus, we chose to utilize these models instead of RNN and LSTM models to achieve lower computational complexity. Considering sample efficiency, we generated the same amount of data as the original data, as we considered it to be an important factor. Additionally, we have also implemented data augmentation in our study, by generating original data twice. Instead of attempting to solve the model accuracy problem by generating an infinite amount of data, generating twice as much data gives us an attempt to improve the accuracy of the model on the data, and it turns out that this works actually. Therefore, an infinite multifold generation of data is not necessary and twice is enough for us.

2.1 Generative model: TimeVAE

TimeVAE Training Dataset. We consider each hourly TEC dataset to be an independent and identically distributed set of samples. The inputs consist of N i.i.d. samples, where N represents the total number of hours in the TEC dataset. The spatial longitude ranges from 180° west to 180° east with a resolution of 5° and the latitude ranges from 87.5° north to 87.5° south with a resolution of 2.5°. As a result, the global TEC map grid consists of 71 x 73 points, with 71 and 73 representing the latitude and longitude information of the TEC map at each hour, respectively, corresponding to different geographical locations. The structure of the generation model is shown in Figure 2, where the input dataset array, represented as (N, 71, 73), is a 3-dimensional array. The latitude and longitude information of the TEC map at each hour, represented by 71 and 73, respectively, correspond to different geographic locations, while N represents the total number of samples.

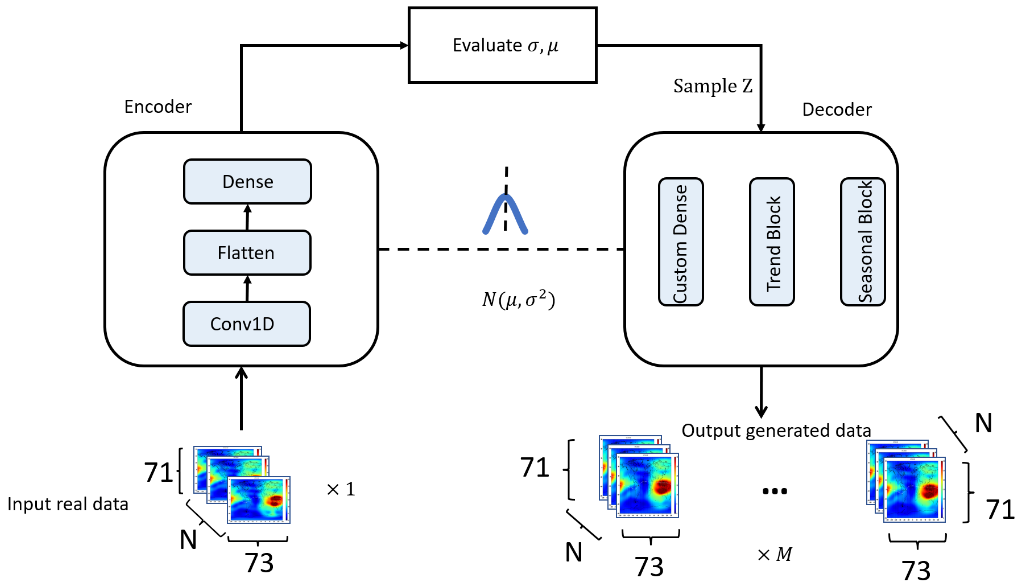

Figure 2. The architecture of TimeVAE

TimeVAE Architecture. To adapt the generation model to the synthesis of ionospheric TEC maps, we adopted an encoder-decoder VAE model. The encoder is to extract the feature of the input i.e. a 3-dimensional array of size $N \times t \times D$, N for batch size, T for the number of time steps, and D for the number of feature dimensions, into a multivariate Gaussian distribution by passing the inputs through a series of convolutional layers with ReLU activation and a fully-connected linear layer. The encoder outputs the parameters of the multivariate Gaussian which can be used to sample the latent vector z using the reparameterization trick. by taking the latent state vector z from the multivariate Gaussian, The decoder first passes the latent vector through a fully-connected linear layer, then reshapes the data into a 3-dimensional array, and passes it through a series of transposed convolutional layers with ReLU activation. Finally, the data is passed through a time-distributed fully-connected layer to produce the final output, which should have the same shape as the original TEC map signal. The goal of the decoder is to generate TEC maps that are as similar as possible to the original TEC maps, based on the information encoded in the latent vector "z".

TimeVAE Loss Function. We train TimeVAE using the Evidence Lower Bound loss(ELBO) function, which is written following:

$$ELBO = E_{q(\boldsymbol{z}|\boldsymbol{x};\phi)}\left[\log(\boldsymbol{x} \mid \boldsymbol{z};\theta)\right] - D_{KL}\left(q(\boldsymbol{z} \mid \boldsymbol{x};\phi)\|p(\boldsymbol{z};\theta)\right) \tag{1}$$

The process of BLBO loss actually is to reconstruct $\boldsymbol{x}$ given $\boldsymbol{z}$ sampled from $q(\boldsymbol{z} \mid \boldsymbol{x};\phi)$. Specifically, the Right Hand Side is composed of two parts, and the first term is the log-likelihood of our data given $\boldsymbol{z}$ sampled from $q(\boldsymbol{z} \mid \boldsymbol{x};\phi)$. The second term is the KL-Divergence loss between the encoded latent space distribution $q(\boldsymbol{z} \mid \boldsymbol{x};\phi)$ and the prior distribution $p(\boldsymbol{z};\theta)$.

2.2 Auto-correlation Based Transformer Architecture

Modeling long-term time series forecasting for TEC maps is not easy: we have to handle the intricate temporal patterns. The original transformer architecture(Vaswani et al., 2017) adopts self-attention modules to calculate the correlation between scattered points but ignores the dependencies among sub-series. In contrast, our approach leverages an auto-correlation-based transformer(Wu et al., 2021) as a prediction model which enables series-wise connections to model dependencies between each sub-series and raises the information utilization. The architecture of the auto-correlation-based transformer is shown in Figure 3.

Series Decomposition Block inherit the ideas from (Cleveland et al., 1990) separate the long time series into two parts: trend-cyclical part and seasonal part, for the former reflects overall trend and fluctuations and the latter reflects the repeating patterns(seasonality) of the series respectively. The series decomposition block is deployed along the model as it goes deeper to capture complex patterns. For the input, TEC maps series $X \in R^{I \times d}$, where I is the length of the input series, and d is the number of features of the TEC map series. The concrete inner operation(Cleveland et al., 1990) for gathering two composed series is:

$$X_t = AvgPool(Padding(X)) \; and \; X_s = X - X_t \tag{2}$$

where X_t is the trend-clynical part, and X_s is the seasonal part. Within the inner operation, AvgPool($\cdot$) is a moving average pooling with the padding operation to keep the series length unchanged.

Auto-correlation mechanism The auto-correlation mechanism is designed based on the periodicity of the time series and aims to conduct the discovery and representation aggregation of dependencies at the sub-series level. The calculation of auto-correlation involves shifting the time series by a certain tag and computing the correlation between the original time series and the shifted time series. From stochastic process theory autocorrelation $R_{xx}(\tau)$ is a time-delay similarity between original series $\{X_t\}$ and its τ lagged series $\{X_{t-\tau}\}$ which can be calculated by:

$$R_{xx}(\tau) = \lim_{L\to\infty} \sum_{t=1}^{L} X_t X_{t-\tau} \tag{3}$$

In the real-world application for TEC map predictions, we first project the embedding of TEC maps to get query Q, key K. A time delay aggregation block is then applied to

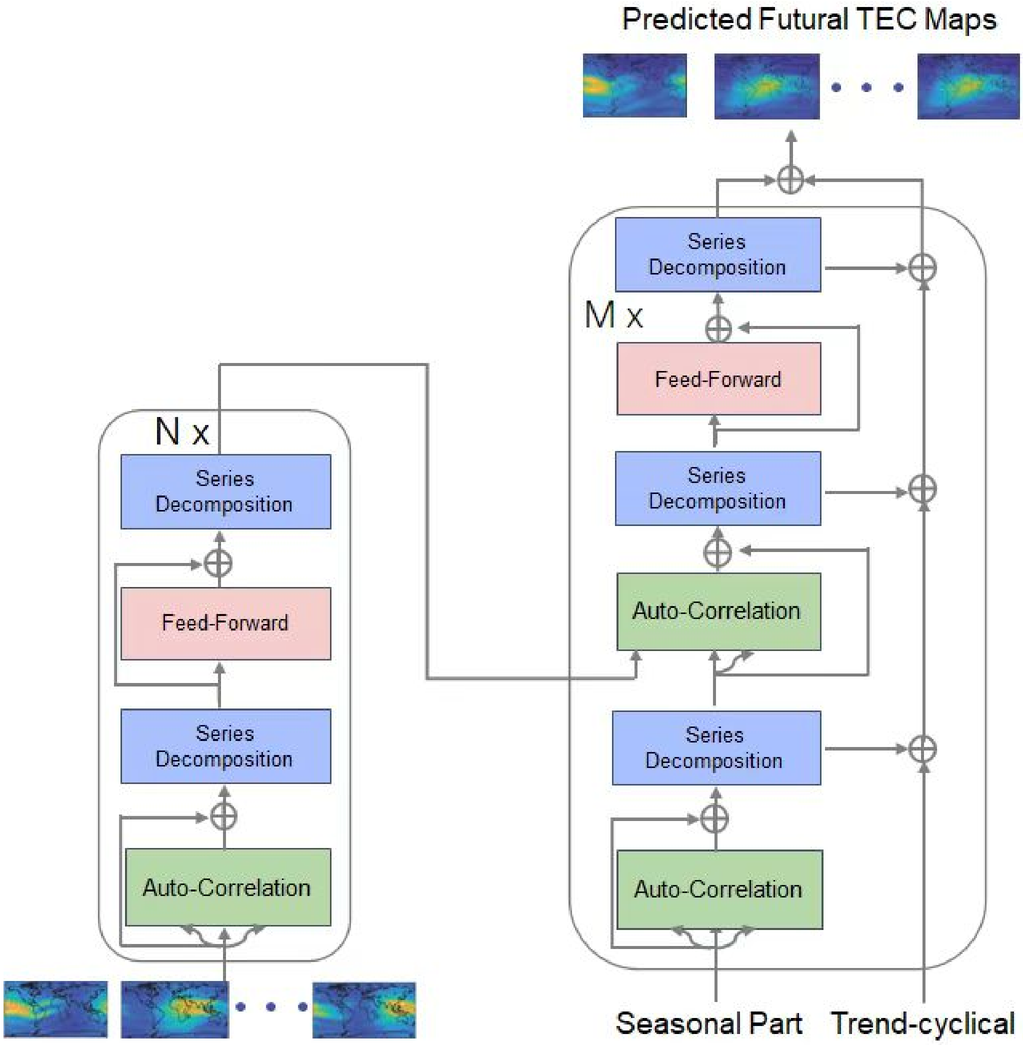

Figure 3. The architecture of auto-correlation-based transformer.

roll the series based on the selected time delay, then we are able to aggregate the sub-series by softmax. Concretely we first get the arguments of TopK autocorrelations:

$$\tau_1 ... \tau_k = argTopk_{\tau \in \{1,...,L\}}(R_{Q,K}(\tau)) \tag{4}$$

$R_{Q,K}$ is the autocorrelation between Q and K. After that, the series are fed into a softmax layer:

$$R_{Q,\hat{K}}(\tau_1), ..., R_{Q,\hat{K}}(\tau_k) = SoftMax(R_{Q,K}(\tau_1) ... R_{Q,K}(\tau_k)) \tag{5}$$

Then the Auto-correlation can be obtained by the relationship between the series and its time-delay shifted version. $Roll(X, \tau)$ the operation to shift the series $\{X_t\}$ with time delay τ. The expression of auto-correlation is:

$$Auto\_correlation(Q, K, V) = \sum_{i=1}^{k} Roll(V, \tau_i)(\hat{R}_{Q,K}(\tau_i)) \tag{6}$$

2.3 Transformer Architecture

The Transformer model consists of an encoder and a decoder. The encoder takes an input sequence and generates a sequence of hidden states, while the decoder takes the encoder output and generates a sequence of output tokens. Both the encoder and decoder consist of multiple layers of self-attention and feedforward neural networks. The self-attention mechanism allows the model to attend to different parts of the input sequence, while the feedforward neural networks enable the model to capture complex patterns in the data. We directly quote the model-building method mentioned in the article (Vaswani et al., 2017) to build the Transformer model.

Encoder: The encoder is composed of two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. The residual connection is around the two sub-layers, followed by layer normalization.

Decoder: The decoder is also composed of a stack of $N = 6$ identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Similar to the encoder, we employ residual connections around each of the sub-layers, followed by layer normalization. We also modify the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions. This masking, combined with the fact that the output embeddings are offset by one position, ensures that the predictions for position i can depend only on the known outputs at positions less than i.

Multi-headed Self-Attention (MSA): Multi-head self-attention allows the model to jointly attend to information from different representation subspaces at different positions. With a single attention head, averaging inhibits this. The basic structure is the same as self-attention, but when operating, it is divided into multiple heads, and then the self-attention calculation is carried out in parallel, and then the output vectors are finally spliced together, where different heads will learn different levels of knowledge.

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^O \tag{7}$$

Where $head = Attention(QW_i^Q, KW_i^K, VW_i^V)$, and the projections are parameter matrices $W_i^Q \in R^{d_{\text{model}} \times d_k}, W_i^K \in R^{d_{\text{model}} \times d_k}, W_i^V \in R^{d_{\text{model}} \times d_v}$ and $W^O \in R^{hd_v \times d_{\text{model}}}$ · .

Positional Encoding This part is used to inject some information about the relative or absolute position tokens from the TEC sequence. tokens in the sequence. To this end, we add "positional encodings" at the bottoms of the encoder and decoder stacks.

The positional encodings have the same dimension d_{model} as the embeddings so that the two can be summed. There are many choices of positional encodings, learned and fixed [9]. In this work, we use sine and cosine functions of different frequencies:

$$\begin{aligned} PE_{(pos,2i)} &= \sin\left(pos/10000^{2i/d_{\text{model}}}\right) \\ PE_{(pos,2i+1)} &= \cos\left(pos/10000^{2i/d_{\text{model}}}\right) \end{aligned} \tag{8}$$

where pos is the position and i is the dimension. That is, each dimension of the positional encoding corresponds to a sinusoid.

3 EXPERIMENTS

Our aim is to achieve two primary objectives. Firstly, we employ TimeVAE to enhance our capability of producing credible TEC time-series dataset samples when we have a shortage of actual TEC datasets for model training. Secondly, we seek to train advanced time forecasting models, specifically the Transformer and Auto-Correlation Transformer, using these generated samples. These two models, Transformer and Auto-Correlation Transformer are the most advanced models in long-series time forecasting. Finally, we will evaluate the accuracy of both prediction models and present our findings. To achieve this, we have prepared three different training datasets names ori_data, gen_once, and gen_twice. The advantage of TimeVAE is that it learns the distribution of TEC datasets, allowing us to generate multiple TEC datasets. This enables us to produce theoretically unlimited datasets even in the absence of sufficient real data.

3.1 Data Source and Processing

In this paper, we use the dataset called global ionospheric TEC data, which is generated by the standard Ionosphere map exchange format (IONEX) file format which is provided by International GNSS Service (IGS). TECU stands for "Total Electron Content Unit" and is a unit of measurement for the ionosphere TEC. TECU is defined as the number of free electrons that would be present in a one square meter column of unit cross section extending from the Earth's surface to the top of the ionosphere if all the free electrons were concentrated in a single point. We conduct experiments with three distinct training datasets, namely ori_data, gen_once, and gen_twice. Here is the description of the experiment datasets:

1) **ori_data:** The global ionospheric TEC data is generated by the standard Ionosphere map exchange format file format which is provided by IGS. We processed datasets into a 4-dimensional time series sequence $(number, 24, 71, 73)$. The longitude dimension consists of 73 points, spanning from 180° west longitude to 180° east longitude with a resolution of 5°. The latitude dimension consists of 71 points, ranging from 87.5° north latitude to 87.5° south latitude with a resolution of 2.5°. The scale of the global TEC map grid points is 71×73. 24 means the total amount hourly for the time resolution of 1 hr each day, and *number* represents the total number of days in the time range from January 1, 1998 to December 31, 2017.
2) **gen_once:** We generated synthetic data as the second dataset, which has been confirmed to have a high similarity to the original dataset ori_data with an identical size.
3) **gen_twice:** We generated ori_data twice repeatedly, each time we obtained an identical dataset with ori_data. Finally, the gen_data dataset consists of two synthetic data, which are twice the size of the original data ori_data.

During the real training model process, 90% of each of the above mentioned datasets are utilized for training while the remaining 10% is used for validation purposes.

3.2 Generated Samples

T-SNE (t-Distributed Stochastic Neighbor Embedding) is a machine learning algorithm for non-linear dimensional reduction developed by Laurens van der Maaten and Geoffrey Hinton(Van der Maaten & Hinton, 2008). It is used to reduce high-dimensional data into a lower-dimensional space for visualization or machine learning. t-SNE works on the principle of minimizing the divergence between a distribution of actual data points in high dimensional space and a distribution of corresponding points in a lower dimensional space. This is done by mapping the data points to a probability distribution in the lower dimensional space.

We use these comparison metrics proposed in (Yoon et al., 2019) to assess the quality of the synthetic data. The lower the discriminative score, the better performance. The abscissa (x-axis) and ordinate (y-axis) of t-SNE represent the positions of data points in a low-dimensional space. The t-SNE algorithm is designed to map high-dimensional data to a low-dimensional space, typically two or three dimensions, for visualization purposes. In t-SNE, each data point in the input dataset is represented as a point in the low-dimensional space, and the similarity between data points in the high-dimensional space is preserved as the distance or closeness between points in the low-dimensional space. The discriminator score of generating the TEC dataset is 0.0035±0.007, which demonstrates the high correlation between synthetic data and original data and TimeVAE performs well in generating synthetic data. Figure 4 displays the t-SNE charts of data generated from the generator TimeVAE. The TimeVAE generated data consistently shows heavy overlap with original data.

Figure 5 and figure 6 present the comparison of the TEC provided by IGS and generated by the model TimeVAE at six randomly selected time points. The top row displays TEC maps provided by IGS, while the bottom row shows TEC maps generated by the TimeVAE model. Six randomly selected time points are labeled at the top of each map, highlighting the comparison between the two sources. The comparison showcases the accuracy of the TimeVAE model in TEC mapping, as well as its potential for use in ionosphere research and satellite-based communication systems.

3.3 Forecasting deep learning models.

Evaluation metric. Root-mean-square error (RMSE) and percentage deviation (PD) following are used to estimate the forecasting performance of the model. The lower the RMSE value, the better the model's accuracy in prediction. In essence, RMSE represents the average magnitude of the errors in the predictions made by a model. The percentage deviation score is calculated by taking the absolute difference between the predicted value and the actual value, divided by the actual value, and multiplied by 100.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(TEC_{\text{ori}} - TEC_{\text{pred}}\right)^2} \tag{9}$$

$$PD = \frac{1}{N}\sum_{i=1}^{N}\frac{|TEC_{\text{ori}} - TEC_{\text{pred}}|}{TEC_{\text{ori}}} \tag{10}$$

where N is the total number of data samples, TEC_{ori} and TEC_{pred} are the observed value and forecasting value, respectively.

4 Forecasting Models Performance and Results Analysis

Our study trains two models, Auto-correlation-based Transformer and Transformer models, using three different datasets: original data, once-generated synthetic data, and

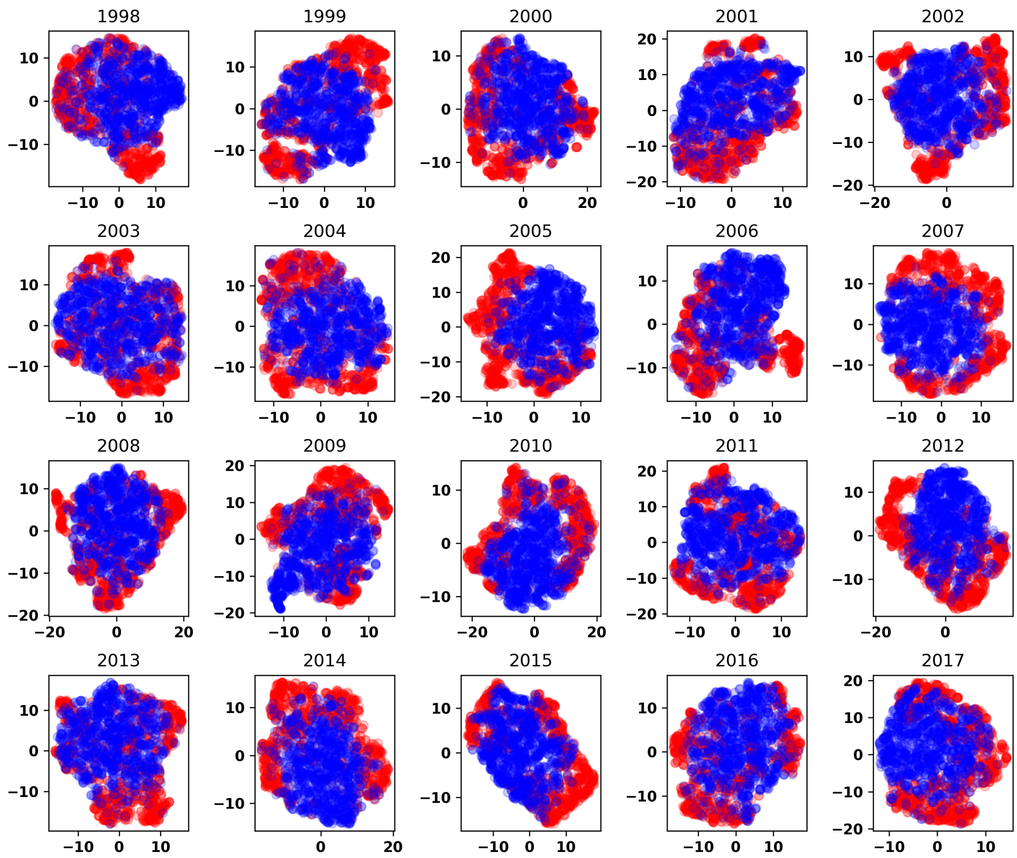

Figure 4. The visual t-SNE plots of generated and original data from 1998 to 2017. Red is for original data and blue is for synthetic data. Higher overlap rates represent higher similarity. The x-axis and y-axis of t-SNE represent the positions of data points($number, 24, 73, 71$) in a two-dimensional space.

twice-generated synthetic data (ori_data, gen_once, and gen_twice). We evaluate the Root Mean Squared Error (RMSE) of the total of six trained models in the years 2018, 2019, and 2020, as well as the RMSE of the IRI2016 model in 2018. Furthermore, we assess the RMSE and percentage deviation (PD) scores of the models in high, low, and middle latitudes to compare their predictive capabilities.

4.1 Traning Set and Testing set Results Analysis

In table 1, we compared the RMSE of IRI2016, 1-day BUAA, Transformer, and Auto-correlation-based transformer models during 2018. We find that our Transformer based models to be superior to the IRI2016 model and 1-day BUAA model. for example, the RMSE of the Auto-correlation-based transformer model is 1.55 TECU, which is 0.52 TECU less than 1-day BUAA and 1.33 TECU less than IRI2016 model. As presented in Table 2, the RMSE scores of all models (Auto-correlation-based Transformer, Transformer) trained on different datasets demonstrate that our models trained using synthetic data outperform the IRI2016 model and 1-day BUAA model. Additionally, it is observed that the more synthetic data used for model training, the better the model performance. For

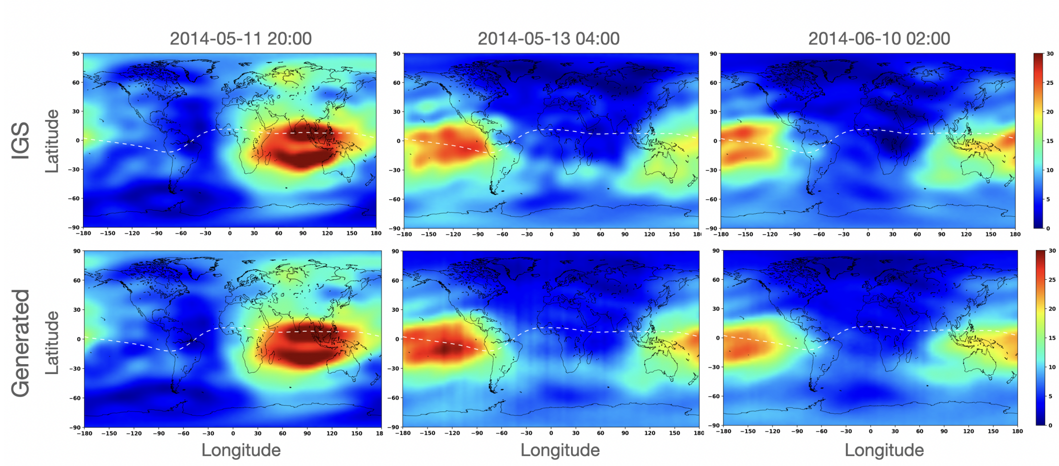

Figure 5. Comparison between the global total electron content map provided by International GNSS Service and generated by TimeVAE model for three stochastic times.

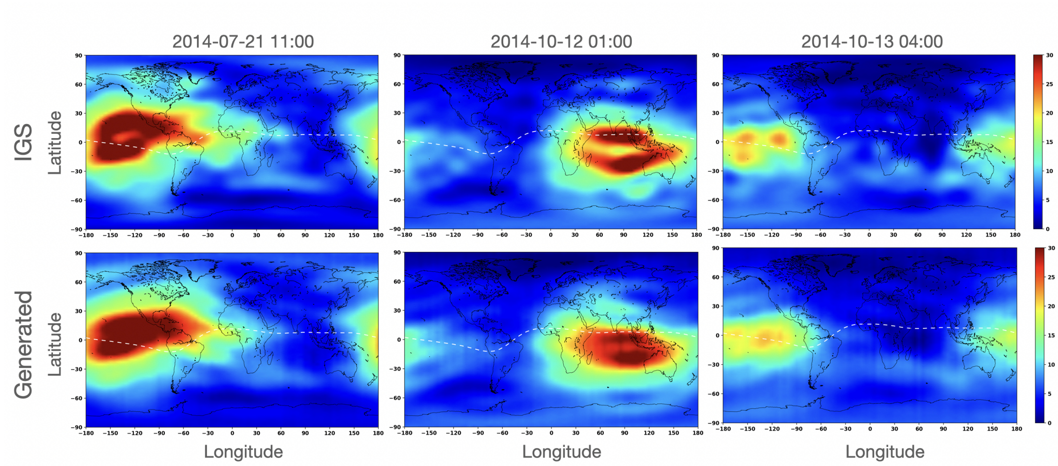

Figure 6. Comparison between the global total electron content map provided by International GNSS Service and generated by TimeVAE model for three stochastic time.

instance, the Auto-correlation-based Transformer trained using twice-generated synthetic data increases the RMSE of the IRI2016 model from 2.88 to 1.17 in 2018. In comparison to the Auto-correlation-based Transformer trained with once-generated synthetic data, the model trained with twice-generated synthetic data has a better performance with RMSE scores of 0.86 and 1.21 in 2019 and 2020 respectively. The Auto-correlation-based Transformer model trained with gen_twice data outperforms the other six models in forecasting TEC from 2018 to 2020. Compared to the model trained with original data, the Auto-correlation-based Transformer model trained with generated data significantly improves its RMSE from 1.55 to 1.17 in 2018, 1.48 to 0.86 in 2019, and 1.58 to 1.21 in 2020. Additionally, the Auto-correlation-based Transformer model demonstrates better performance than the Transformer model in certain years from 2018 to 2020.

Table 1. The accuracy RMSE scores on four different training models (IRI2016 model, 1-day BUAA, Auto-correlation-based transformer, Transformer) during 2018.

Models	IRI2016 model	1-day BUAA	Auto-correlation-based transformer	Transformer
RMSE	2.88	2.07	1.55	1.37

Table 2. Multivariate accuracy RMSE scores on three different training datasets with the pre-trained models. Training datasets include ori_data, gen_once, and gen_twice, and ori_data means IGS TEC datasets from 1998 to 2017, gen_once means the generated datasets from 1998 to 2017 and gen_twice means generated dataset twice from 1998 to 2017. The best performances are in bold, a lower RMSE score indicates a better prediction.

Models	Training datasets	RMSE (TECU)		
		2018	2019	2020
Anto-correlation-based transformerr	ori_data	1.55	1.48	1.58
	gen_once	1.22	0.91	1.25
	gen_twice	**1.17**	**0.86**	**1.21**
Transformer	ori_data	1.37	1.08	1.96
	gen_once	1.31	1.00	1.75
	gen_twice	1.29	1.00	1.73

4.2 Comparison of Predictional TEC Maps

We train the Auto-correlation-based transformer model and the Transformer model on multivariate training datasets and also evaluated their monthly root mean squared error (RMSE) during 2018 and 2020. As shown in Fig 8, we plotted the predicted TEC map and the original TEC map for January 3, 2020, with the top half coming from the original IGS data and the bottom half coming from our best-performing model Autoformer-correlation-based transformer. Fig 7 shows the TEC map on Five points in time on January 3, 2019, which also compared the original real TEC map with the predicted TEC map. We assessed the monthly RMSE of the IRI2016 model in 2018 and compared it with the original data trained Transformer model and the original data trained Auto-correlation-based Transformer model in figure 9. The results indicate that both the Auto-correlation-based Transformer model and the Transformer model offer improved predictive capabilities when compared to the IRI2016 model.

Figure 10 displays the monthly root mean squared error (RMSE) of the Auto-correlation-based Transformer model and the Transformer model trained on three different datasets in 2018. A_ori_data, A_gen_once and A_gen_twice represent Auto-correlation-based Transformer model trained on ori_data, gen_once and gen_twice respectively. And T_ori_data, T_gen_once, and T_gen_twice represent the Transformer model trained on ori_data, gen_once, and gen_twice respectively. The results presented in Figure 10 demonstrate that the model trained with twice the amount of synthetic data performed better than the model trained with only the original data in 2018. Specifically, the Auto-correlation-based transformer model trained with gen_twice performed more accurately than other models trained with only original data or gen_once for all 12 months in 2018, with the best predictions occurring in June and October, and the biggest RMSE value being 1.175 occurring in month 1. The worst model is the Transformer model trained by original data with an RMSE over 1.5 TECU but less than 2.0 TECU in every month of 2018.

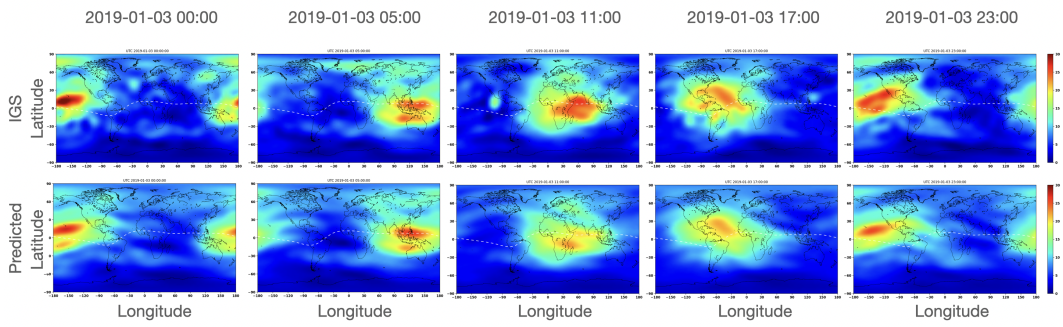

Figure 7. The comparison between the TEC map of Auto-correlation-based transformer trained by gen_twice dataset and the TEC map of the real IGS on January 3, 2019.

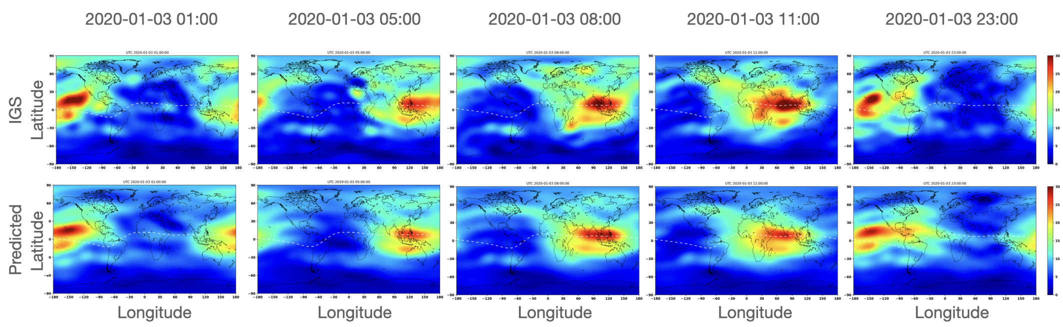

Figure 8. The comparison between the TEC map of Auto-correlation-based transformer trained by gen_twice dataset and the TEC map of the real IGS on January 3, 2020.

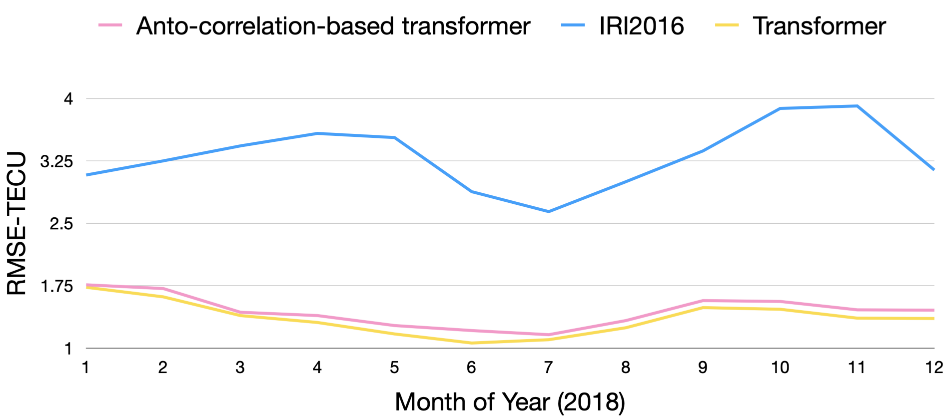

Figure 9. The monthly averaged RMSE scores of two different models(Auto-Correlation Transformer, Transformer) compared with the IRI2016 model with the same training datasets during 2018.

The results from the comparison of prediction accuracy in 2020, as shown in Figure 11, indicates that the Transformer model trained on the gen_twice dataset achieved the highest accuracy in predicting monthly TEC values. The RMSE values for all three models varied between approximately 1.4 TECU and 2.5 TECU throughout the year, with the best predictions occurring in October and November and the worst predictions occurring in July and August. In contrast, the Transformer model trained on the ori_data is expected to perform poorly in every month of 2020 when compared to the other two dataset training models, having a high level of RMSE error. Additionally, in each month of 2020, the performance of the Transformer model trained on the gen_twice dataset was slightly better than the model trained on the gen_once dataset. Overall, the Transformer models performed best in October and November, with the lowest RMSE values among the 12 months in 2020.

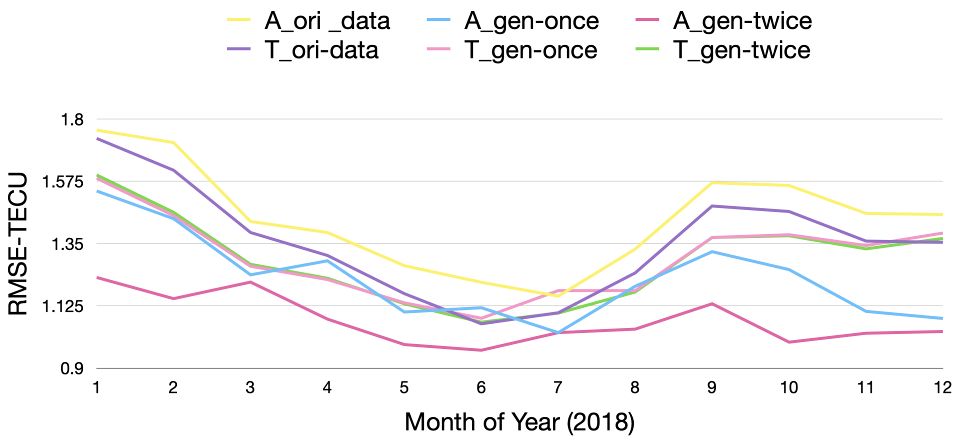

Figure 10. The monthly average RMSE scores for the Auto-Correlation Transformer models and Transformer models trained on three different datasets (ori_data, gen_once, gen_twice) during 2018. The labels for the figures begin with the letter 'A' to represent the Auto-correlation-based transformer model, while the letter 'T' is used to represent the Transformer model.

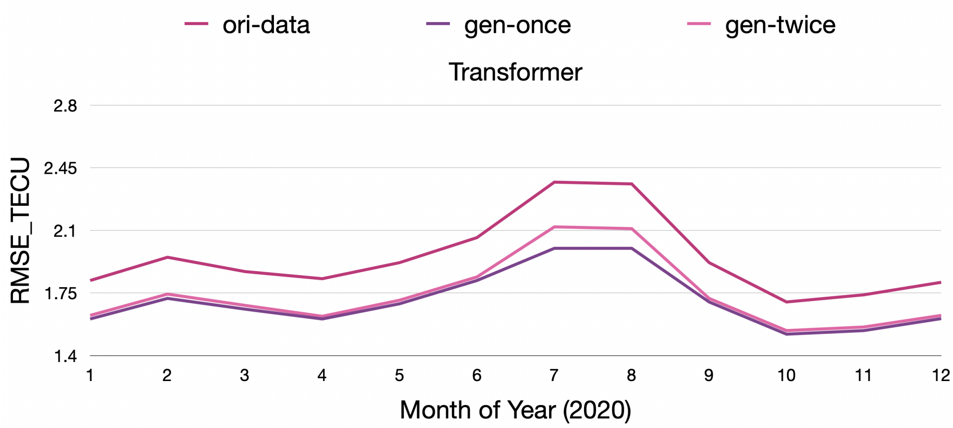

Figure 11. The monthly RMSE values of the Tramsformer model trained on 3 dataset (ori_data, gen_once, and gen_twice) in 2020.

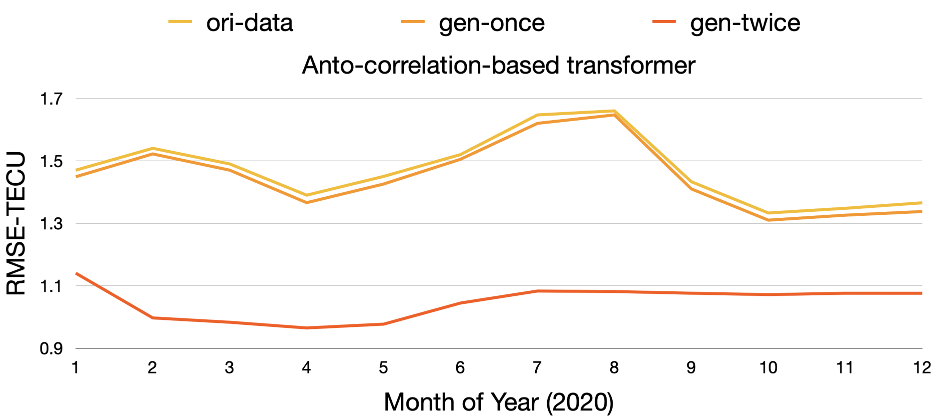

Figure 12. The comparison of monthly RMSE values of the Auto-correlation-based transformer models trained on 3 datasets (ori_data, gen_once, and gen_twice) in 2020.

The comparison of the prediction accuracy of the Auto-correlation-based transformer model in 2020 on three different training datasets in figure 12 shows that, similar to that of the Transformer model, the Auto-correlation-based transformer model trained on the gen_twice dataset has the highest accuracy in predicting monthly TEC values. The RMSE values of the three models fluctuate between approximately 0.9 and 1.3 TECU throughout the year, with the best predictions in April and March. On the other hand, the Auto-correlation-based transformer model trained on the ori_data performed the worst in every month of 2020, with slightly lower performance than gen_once every month, with the RMSE values fluctuating between approximately 1.3 and 1.7 TECU throughout the year.

4.3 Latitude Results Analysis

Figure 13 illustrates the comparison of RMSE scores in different latitudes of two different models trained on three datasets (ori_data, gen_once, gen_twice) during 2018, The first three models are Auto-correlation-based transformer models trained on ori_data, gen_once, and gen_twice, while the next three are Transformer models trained on the same three datasets. The last model represented is the IRI2016 model. Our analysis reveals that all six training models outperform the IRI2016 model, as they exhibit lower RMSE scores in high, low, and middle latitudes. In low latitudes, we have achieved a significant improvement in RMSE from around 4 TECU to approximately 2.5 TECU. Out of the six training models, five of them achieved a lower RMSE score of about 2.5 TECU in low latitude, with the remaining Auto-correlation-based transformer model trained on twice synthetic data presenting the lowest RMSE score of around 0.8 TECU. The RMSE scores of the six training models in high and middle latitudes are consistently around 1 TECU. It is noteworthy that the accuracy in high latitudes tends to be lower compared to that in middle latitudes for each of the models. The percentage deviation score is a measure of the deviation of a model's predictions from the actual values in percentage. A lower percentage deviation score indicates more accurate predictions. Figure 14 displays these results. As can be seen, all of the models perform best in the middle latitude, with a percentage deviation score of around 8% to 10%. This suggests that the models are generally more accurate in this latitude range Furthermore, Figure 14 indicates that the Auto-correlation-based Transformer model performed best in the middle latitude, with a percentage deviation score of 8.4%. In contrast, the Transformer model did not perform as well, with percentage deviation scores in the high latitude exceeding 17%. These

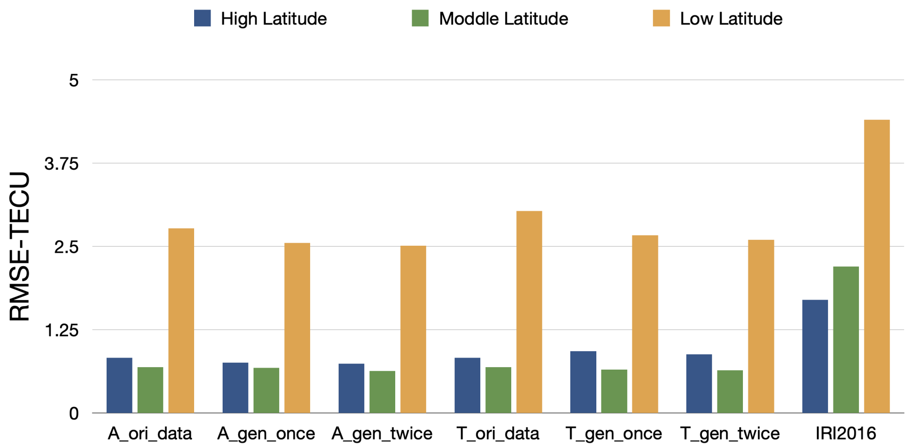

Figure 13. The averaged RMSE scores of all models with three different latitudes (high latitude, middle latitude, low latitude) during 2018.

results demonstrate that the Auto-correlation-based Transformer model may be a better choice for predicting TEC values in certain latitude ranges.

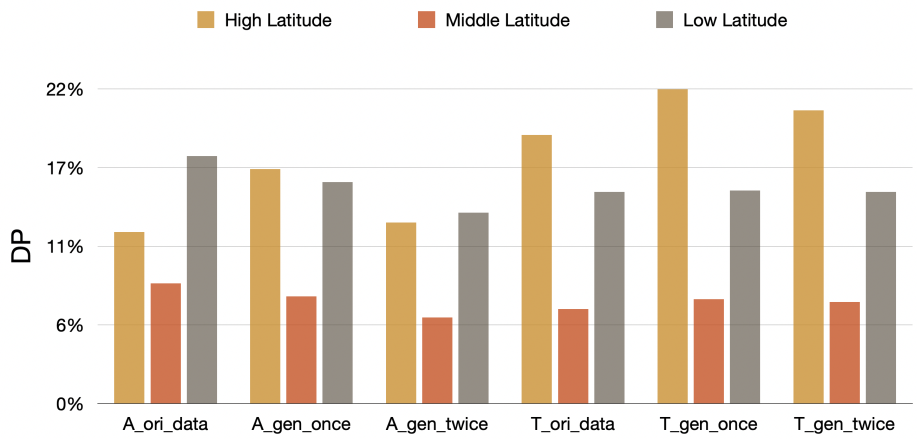

Figure 14. The DP scores of all models with three different latitudes (high latitude, middle latitude, low latitude) during 2018.

5 Conclusion

The Total Electron Content of the Earth's ionosphere plays a critical role in the satellite-based communications system. In this study, we compare the performance of two transformer models trained on synthetic data with the widely used IRI2016 model and BUAA model in TEC prediction. We applied data enhancement to TEC forecasting and found that it was effective in improving the quality of predictions.

Our model has shown significant improvement in predicting TEC at mid and high latitudes. However, at low latitudes, the RMSE accuracy of our model is still above the desired value of 2.5 TECU, indicating the need for further improvement in this area. On the other hand, both the monthly RMSE and Latitude RMSE of our model outperforms the IRI2016 model. Our results show that the transformer model trained on synthetic data achieves higher accuracy and reliability in TEC prediction, with a significant RMSE reduction of 1.17 compared to 2.88 for the IRI2016 model. This finding has important implications for the development of advanced TEC prediction models and highlights the potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

Furthermore, when tested on 2018 prediction data, the Auto-correlation-based transformer model using synthetic data exhibits an RMSE reduction of 0.38 TECU compared to the model trained on original data. We demonstrate that utilizing synthetic data can effectively enhance the prediction efficiency of the model, providing another avenue for enhancing the model's accuracy. Moreover, with the same amount of data, models trained on synthetic data offer more accurate predictions than those solely using actual data. The use of synthetic data has several contributions to the study. First, it can effectively enhance the prediction efficiency of the model, improving the accuracy of TEC prediction, but would gradually saturate as the amount of data increases. Second, it offers a way to generate more data without the need for additional data collection, which is particularly useful in cases where obtaining real data is difficult or expensive. Third, the study shows that models trained on synthetic data can outperform those trained on real data in terms of accuracy, indicating that synthetic data can provide a valuable alternative for training machine learning models. The results demonstrate the great potential of transformer models trained on synthetic data for a range of applications in ionospheric research and satellite communication systems.

6 Open Research

The TEC GIMs provided by IGS is available from the website at `"http://pub.ionosphere.cn/products/daily/"` and the BUAA 1-day prediction data can be obtained from the website (`"http://pub.ionosphere.cn/prediction/daily/"`).

References

Béniguel, Y., & Hamel, P. (2011). A global ionosphere scintillation propagation model for equatorial regions. *Journal of Space Weather and Space Climate*, *1*(1), A04.

Cesaroni, C., Spogli, L., Aragon-Angel, A., Fiocca, M., Dear, V., De Franceschi, G., & Romano, V. (2020). Neural network based model for global total electron content forecasting. *Journal of space weather and space climate*, *10*, 11.

Chen, Z., Jin, M., Deng, Y., Wang, J.-S., Huang, H., Deng, X., & Huang, C.-M. (2019). Improvement of a deep learning algorithm for total electron content maps: Image completion. *Journal of Geophysical Research: Space Physics*, *124*(1), 790–800.

Cherrier, N., Castaings, T., & Boulch, A. (2017). Forecasting ionospheric total electron content maps with deep neural networks. In *Proc. conf. big data space (bids), esa workshop.*

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). Stl: A seasonal-trend decomposition. *J. Off. Stat*, *6*(1), 3–73.

Desai, A., Freeman, C., Wang, Z., & Beaver, I. (2021). Timevae: A variational auto-encoder for multivariate time series generation. *arXiv preprint arXiv:2111.08095*.

Feng, J., Han, B., Zhao, Z., & Wang, Z. (2019). A new global total electron content

empirical model. *Remote Sensing*, *11*(6), 706.
Ha, D., & Schmidhuber, J. (2018). World models. *arXiv preprint arXiv:1803.10122*.
Iyer, S., & Mahajan, A. (2023). Short-term adaptive forecast model for tec over equatorial low latitude region. *Dynamics of Atmospheres and Oceans*, *101*, 101347.
Lastovicka, J., Urbar, J., & Kozubek, M. (2017). Long-term trends in the total electron content. *Geophysical Research Letters*, *44*(16), 8168–8172.
Li, Q., Yang, D., & Fang, H. (2022). Two hours ahead prediction of the tec over china using a deep learning method. *Universe*, *8*(8), 405.
Li, Z., Wang, N., Wang, L., Liu, A., Yuan, H., & Zhang, K. (2019). Regional ionospheric tec modeling based on a two-layer spherical harmonic approximation for real-time single-frequency ppp. *Journal of Geodesy*, *93*, 1659–1671.
Lin, M., Zhu, X., Tu, G., & Chen, X. (2022). Optimal transformer modeling by space embedding for ionospheric total electron content prediction. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1–14.
Lin, X., Wang, H., Zhang, Q., Yao, C., Chen, C., Cheng, L., & Li, Z. (2022). A spatiotemporal network model for global ionospheric tec forecasting. *Remote Sensing*, *14*(7), 1717.
Liu, L., Morton, Y. J., & Liu, Y. (2022). Ml prediction of global ionospheric tec maps. *Space Weather*, *20*(9), e2022SW003135.
Liu, L., Zou, S., Yao, Y., & Wang, Z. (2020). Forecasting global ionospheric total electron content (tec) using deep learning. In *Agu fall meeting abstracts* (Vol. 2020, pp. NG004–0017).
Mendoza, L. P. O., Meza, A. M., & Aragón Paz, J. M. (2019). A multi-gnss, multifrequency, and near-real-time ionospheric tec monitoring system for south america. *Space Weather*, *17*(5), 654–661.
Monte-Moreno, E., Yang, H., & Hernández-Pajares, M. (2022). Forecast of the global tec by nearest neighbour technique. *Remote Sensing*, *14*(6), 1361.
Prol, F. d. S., Camargo, P. d. O., Monico, J. F. G., & Muella, M. T. d. A. H. (2018). Assessment of a tec calibration procedure by single-frequency ppp. *GPS Solutions*, *22*, 1–11.
Ratnam, D. V., Vishnu, T. R., & Harsha, P. B. S. (2018). Ionospheric gradients estimation and analysis of s-band navigation signals for navic system. *IEEE Access*, *6*, 66954–66962.
Ruwali, A., Kumar, A. S., Prakash, K. B., Sivavaraprasad, G., & Ratnam, D. V. (2020). Implementation of hybrid deep learning model (lstm-cnn) for ionospheric tec forecasting using gps data. *IEEE Geoscience and Remote Sensing Letters*, *18*(6), 1004–1008.
Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, *9*(11).
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.
Wang, C., Xin, S., Liu, X., Shi, C., & Fan, L. (2018). Prediction of global ionospheric vtec maps using an adaptive autoregressive model. *Earth, Planets and Space*, *70*(1), 1–14.
Wang, H., Lin, X., Zhang, Q., Chen, C., Cheng, L., & Wang, Z. (2022). Global ionospheric total electron content prediction based on spatiotemporal network model. In *China satellite navigation conference (csnc 2022) proceedings: Volume ii* (pp. 153–162).
Wu, H., Xu, J., Wang, J., & Long, M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, *34*, 22419–22430.
Xia, G., Liu, M., Zhang, F., & Zhou, C. (2022). Caitst: Conv-attentional image time sequence transformer for ionospheric tec maps forecast. *Remote Sensing*,

14(17), 4223.
Xia, G., Zhang, F., Wang, C., & Zhou, C. (2022). Ed-convlstm: A novel global ionospheric total electron content medium-term forecast model. *Space Weather*, *20*(8), e2021SW002959.
Yang, K., & Liu, Y. (2022). Global ionospheric total electron content completion with a gan-based deep learning framework. *Remote Sensing*, *14*(23), 6059.
Yoon, J., Jarrett, D., & Van der Schaar, M. (2019). Time-series generative adversarial networks. *Advances in neural information processing systems*, *32*.