TSEQPREDICTOR: SPATIOTEMPORAL EXTREME EARTHQUAKES FORECASTING FOR SOUTHERN CALIFORNIA

Anonymous ICME submission

ABSTRACT

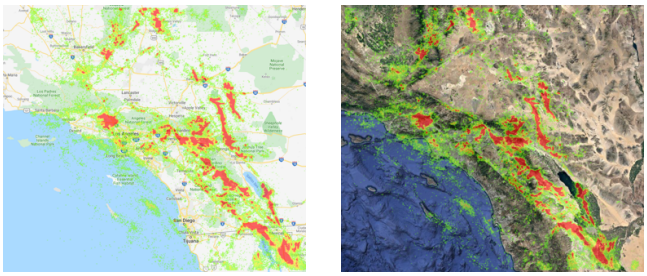
Seismology from the past few decades has utilized the most advanced technologies and equipment to monitor seismic events globally. However, forecasting disasters like earthquakes is still an underdeveloped topic from the history. Recent researches in spatiotemporal forecasting have revealed some possibilities of successful predictions, which becomes an important topic in many scientific research fields. Most studies of them have many successful applications of using deep neural networks. In the geoscience study, earthquake prediction is one of the world's most challenging problems, about which cutting edge deep learning technologies may help to discover some useful patterns. In this project, we propose a joint deep learning modeling method for earthquake forecasting, namely TSEQPREDICTOR. In TSEQPREDICTOR, we use comprehensive deep learning technologies with domain knowledge in seismology and exploit the prediction problem using encoder-decoder and temporal convolutional neural networks. Comparing to some state-of-art recurrent neural networks, our experiments show our method is promising in terms of predicting major shocks for earthquakes in Southern California.

Index Terms— Earthquake prediction; Spatiotemporal dynamics; Convolution

1. INTRODUCTION

Over the past few decades, large volumes of data have been collected by the seismological community. This drives high demand for seismology data processing and analysis, which also provides opportunities to predict future dynamics from history. Spatiotemporal forecasting is an important topic in many scientific research fields, in which there are a plethora of successful applications. Recent studies using deep neural networks have shown various successful applications, including car traffic forecasting [1], ride-hailing forecasting [2], rain/weather forecasting [3], etc.

Earthquake forecasting is a worldwide challenging problem. Scientists around the world have built an enormous number of detectors for picking up earthquake signals. It is a general belief that earthquakes are predictable under some assumption that quakes are formed underneath the Earth are accumulated stresses in a gradual process over a long time. In this case, it would be possible to predict earthquake shocks for future activities of quakes by learning patterns from historical seismic events.



(a) (b)

Fig. 1. Dataset overview of earthquakes in Southern California. (a) Earthquake events mapped on Maps. (b) Earthquake events mapped on satellite images.

Conventionally, earthquakes are located through a process of detecting signals, picking up arrival time, and estimating epicenters of events using a velocity model. Efforts have been made to filter P-waves and S-waves from the original waveform signals of earthquakes and seismic noise [4]. In this project, our goal is to utilize the preprocessed seismic signals forming epicenters (location labels) to forecast the probabilities of the next earthquakes in an area.

Earthquake forecasting consist of three major tasks in machine learning. The first task is to predict when the next seismic event will happen in a specific region. The second task is to predict whether or not the next seismic event will come. The third task is to predict the level of magnitude of the upcoming seismic events so that a major shock can be predicted.

Deep learning neural networks have presented a widely successful approach to capture spatial-temporal dependencies of problems to achieve accurate forecasting results. Convolutional neural networks have achieved convinced success in computer vision, image object recognition, etc [5]. Here we test the hypothesis that earthquake patterns can be perceived by learning historical seismic events. However, epicenters' prediction is learned from annotated seismograms. Due to the uncertainties of earthquakes, even the ground truth labels that are annotated by domain experts may be biased. Locations and magnitude of epicenters are maybe adjusted after the seismic event happened a long while.

In this project, we propose joint modeling of using self-supervised autoencoder and temporal convolutional (TCN) neural networks for earthquake prediction by modeling spatiotemporal dependencies in Southern California. Additionally TSEQPREDICTOR comprehensively improves the autoencoder and TCN by incorporating skip connections and local temporal attention mechanisms. Compared to conventional recurrent neural networks or a single model, our joint modeling presents some advantages in predicting major shocks in the area of study. In summary:

- We study the earthquake dataset for Southern California and reconstruct the time series events into a sequence of 2D images.
- We model the spatiotemporal dependencies of earthquakes in Southern California with an improved autoencoder and TCN neural networks and show some preliminary but promising results for forecasting events.

2. RELATED WORK

Convolutional Methods in Predicting Epicenters. Estimating and predicting the epicenters of earthquakes has a long history. Scientists from Geophysics, Geology and Seismology have developed a variety of tools and analytical functions to predict epicenters from datasets. In 1997, Bakun and Wentworth suggested using Modified Mercalli intensity datasets for southern California earthquakes to bound the epicenter regions and magnitudes [6]. In 1998, Pulinets proposed predicting epicenters of strong earthquakes with the help of satellite sounding systems. Scientists from Greece had illustrated a successful project which predicted the large aspects of earthquakes using seismic electric signals [7]. Recently, Guangmeng *et al.* attempted to predict earthquakes with satellite cloud images and revealed some possibilities of predicting earthquakes using geophysics data [8]. Zakaria *et al.* presented their work of predicting epicenters by monitoring precursors, such as crustal deformation anomalies and thermal anomalies, with remote sensing techniques [9]. These studies either used only too little data or too simple analytical models.

Spatiotemporal Dynamic Capture and Generative Models. Most recently, it is a prevailing method to do predictions by modeling the spatiotemporal dynamics for domain science problems. This is because large volumes of data are increasingly collected in the vast majority of domains including, social science, epidemiology, transportation, and geosciences. Cui *et al.* proposed to use graph convolutional long short-term memory neural networks to predict traffic via capturing spatial dynamics from the car traffic patterns [10]. Li *et al.* utilized a seq2seq neural network architecture to capture spatial and temporal dependencies for traffic forecasting by incorporating a diffusion filter in convolutional recurrent layers [11]. FUNNEL was a project proposed by Matsubara [12]. It was designed to use an analytical model and a fitting algorithm for discovering spatial-temporal patterns of epidemiological data.

3. TSEQPREDICTOR

The proposed prediction model consists of two major components, an autoencoder which learns the latent space distribution from the image like view of the earthquakes and a prediction network which learns to predict the likelihood of the next main shock happening within the same area.

3.1. Data Considerations

The earthquake catalog is a tablet formatted dataset. In this project, we focus on time and geo location shocks. The dataset contains all earthquake events in Southern California ranging from the year 1990 to 2019. Figure 1 shows all events plotted in 2D maps, in which hot spots are areas where earthquakes frequently happened or big earthquakes happened in history.

Energy-based data models. Seismometers record seismic events from calibrating vibrations of waves. Magnitude in the dataset represents measured amplitude as measured seismogram. While they are discrete data points, accumulating all magnitudes by summing them up by averaging makes the temporal information loss, and deemphasizes large earthquakes. In contrast to magnitude, earthquakes release energy can help mitigate this issue by two folds: 1) accumulated energy value in a region can represent the energy released by the stress of Earth over time; 2) energy data model naturally highlights large events since the energy of large events can be an order of magnitude higher than that of small events. The formula of converting earthquake magnitude to energy is defined as

$$\mathbf{E} = (10^{\mathbf{M}})^{3/2} \tag{1}$$

in which the magnitude $0 \leq \mathbf{M} \in \mathbb{R} \leq 10$. [1]

Location-aware data weaving. As a time-series prediction task, the earthquake catalog contains locations and magnitudes, which could be used as target properties. However, it could be more nature to reorganize the 1D time-series dataset into a 2D sequence dataset by dividing a map region into small boxes according to longitudes and latitudes and aggregating the released energy within a small box per specific time frequency. So each element of the sequence becomes a summation of all energy released at the location (i, j): $X^t_{i,j} = \sum((10^{\mathbf{M}})^{3/2}), i \in [0, M)$ and $j \in [0, N)$, which means X^t has a shape $M \times N$ for M boxes along the latitude and N boxes along the longitude.

3.2. AutoEncoder for Effective Spatial Modeling

Main shocks with large magnitudes are rare in terms of statistics and nature physics. In addition, earthquakes are full of stochastic processing, resulting in seismic signals are very

[1] Earthquake magnitude can be even negative for very small events that are negligible. This scale is also open-ended, but events larger than 10 are extremely unlikely to happen. So these are out of the scope of this project.

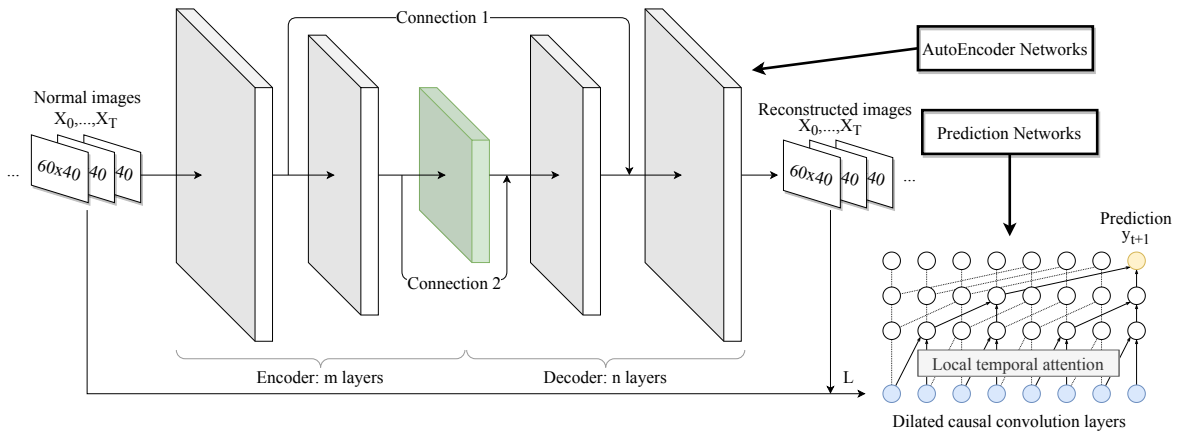


Fig. 2. TSEQPredictor: overview of earthquake prediction networks.

noisy. To predict the future main shocks, we first model the spatial patterns within the southern California area.

We use an autoencoder to recognize the spatial pattern changes under normal circumstances and abnormal circumstances. Compared to variational autoencoders (VAE), we do not assume Gaussian distribution or any other kinds of distributions for the latent space. In addition, the reconstructed results from VAE are tended to be more noisy. We also make some experiments for full comparison in Section 4. This is a semi-supervised process of pretraining a model that learns the representation of earthquake images. We train this model by using the following equation.

$$L(\mathbf{X_{normal}}, g(f(\mathbf{X_{normal}})) + \Omega(h, \mathbf{X_{normal}}) \quad (2)$$

, where $\mathbf{X_{normal}}$ are images of earthquakes with magnitudes less than a threshold, f is an encoder function, g is an decoder function, and Ω is a function that regularizes or penalizes the cost.

Spatial modeling. After the seismic events are parsed and transformed to image-like sequences in Section 3.1, we can utilize the spatial dependencies between pixels. Convolutional operations are common image feature extraction means. Pixel relationship can be easily mapped to geology locations of events.

Skip connections. We incorporate skip connections in the AutoEncoder architecture. Skip connections are forward shortcuts in networks. They symmetrically connect layers from the encoder and decoder as shown in Figure 2. This strategy allows long skip connections to pass features from the encoder path to the decoder path directly, which can recover spatial information lost due to downsampling, according to [13].

Bottleneck layer. The bottleneck layer in the AutoEncoder is deliberately set to a small vector of a size k feature map. This design is effective for two reasons. Firstly, it regularize the model from overfitting all samples. Secondly, a small feature map can better differentiate the abnormal cases from normal cases.

3.3. TCN Model for Effective Temporal Modeling

In this work, the goal of forecasting earthquakes is to predict the future probability of a major shock happening in Southern California. This can be done in a prediction network, which is fed in the information gained from the AutoEncoder. A long short-term memory (LSTM) model can predict well on this task. However, in TSEQPREDICTOR we incorporate an enhanced TCN (Figure 2), which can outperform LSTM. This situation is similar in predicting other physics related fields of study. For example, TCN is used to predict climate changes [14]. This is further analyzed in the following sub sections.

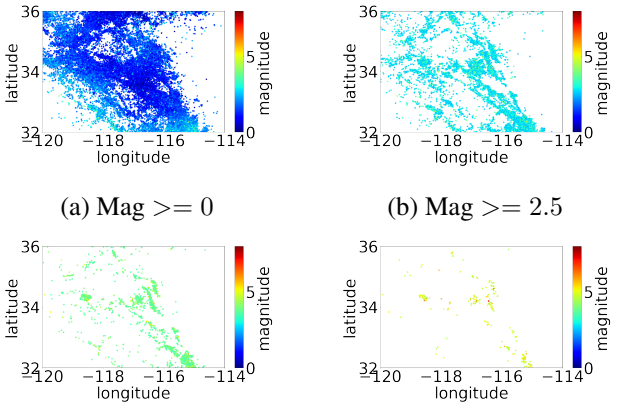
3.3.1. Conditional Temporal Convolution

Temporal convolution neural networks are used to improve the temporal locality prediction over time. Temporal convolutional layers are layers containing causal convolution with varied dilation rate in 1D convolutional layers [15, 16]. A typical configuration of temporal convolution layers is set the dilation rate corresponding to the i-th of layers, for example 2^i.

$$p(y|\theta) = \prod_{t=1}^{T} p(y_{t+1}|y_1, \ldots, y_t, \theta) \quad (3)$$

3.3.2. Local Temporal Attention

A localized attention process to enhance temporal information passing is inspired by self-attention structure from Transformer [17], and Hao *et al.* work for sequence modeling [18]. The process incorporates functions f, g, and h to calculate d dimensional vector of keys $\mathcal{K}$, queries $\mathcal{Q}$, and values $\mathcal{V}$ respectively. Then, we calculate the weight matrix by $W = \frac{\mathcal{K} \cdot \mathcal{Q}}{\sqrt{d}}$. Finally, we apply a softmax function to the lower triangle of W to get a normalized attention weight $W_{attention} = softmax(W)$ and the final out of this layer can be calculated via this attention weighted summary: $\sum_{t=1}^{T} W_{attention} \cdot y_t$.



(a) Mag >= 0 (b) Mag >= 2.5

(c) Mag >= 3.5 (d) Mag >= 4.5

Fig. 3. Dataset overview: (a) $444,589$ events with magnitude ≥ 0.0, (b) $24,822$ events with magnitude ≥ 2.5, (c) $2,489$ events with magnitude ≥ 3.5, (d) 237 events with magnitude ≥ 4.5

3.3.3. Smooth joint Nash–Sutcliffe efficiency: NSE

Nash–Sutcliffe model efficiency coefficient (NSE) is a commonly used metric to evaluate a predictive model. NSE is widely used to evaluate predictive skills in scientific studies, such as hydrology [19]. The value range of NSE is $(-\infty, 1)$. NSE can become negative when the mean error in the predictive model is larger than one standard deviation of the variability. Its equation is defined as follows.

$$NSE = 1 - \frac{\sum_{t=0}^{T}(\hat{y_t} - y_t)}{\sum_{t=0}^{T}(y_t - \bar{y})} \tag{4}$$

4. EXPERIMENTS AND EVALUATION

4.1. Earthquake Prediction: Dataset and Preprocessing

The earthquake dataset is a tablet formatted dataset in which each record is an earthquake epicenter with a timestamp, a GEO location, a magnitude, and depth. We preprocess the catalog according to the analysis in Section 3.1.

Data augmentation. We divide the Southern California (Longitude: -120~-140, Latitude: 32~36) into a grid with 60×40 cells, each of which has 1 degree of longitude and latitude (1 degree in kilometers is about 111km). Firstly, it is easy to group events into daily intervals. Then, let x, y denote the longitude and latitude location of an event. All events are accumulated in corresponding cell where x, y fall into. The value of each cell is the mean of magnitudes of all events within the cell. As a result, each day is represented by a 2D image-like 60×40 matrix.

Table 1. TSEQPREDICTOR AutoEncoder vs. VAE.

Model	MSE	Accuracy	Variance
TSEQPREDICTOR	0.148	0.968	1.432
VAE [20]	0.157	0.971	1.986

Table 2. Varying the latent space dimension.

Latent space dimension	MSE	Accuracy
16	0.148	0.968
64	0.140	0.968
128	0.138	0.972
1024	0.137	0.984

4.2. Experimental setup

Our TSEQPredictor model and other baseline models are implemented with Tensorflow in Python. All experiments are conducted on a machine with 8 NVidia K80 GPUs. All models, include TSEQPREDICTOR and baseline models are trained using Adam or SGD optimizers with a fine-tuned learning rate and mean squared error as training loss. All model weights are checkpointed and we select the best model weights for testing. Events with magnitudes ≥ 4.5 are labeled as extreme major shocks.

4.3. Experimental Results

In these set of experiments, we aim to demonstrate the performance of TSEQPREDICTOR compared to a series of baseline models. Firstly, we show the performance differences between autoencoder in TSEQPREDICTOR and a VAE. Then, we compare the prediction network with a LSTM. Finally, we illustrate the comprehensive results from using TSEQPREDICTOR comparing with a series of methods.

4.3.1. AutoEncoder

As we mention an antoencoder is used in TSEQPREDICTOR in Section 3 as opposed to a variational autoencoder, we compare the results of using TSEQPREDICTOR autoencoder with a common VAE. The performance results are summarized in Table 1. Even though VAE can achieve almost the same performance in terms of accuracy, it has higher mean squared loss and variance for the final output. Higher MAE loss and variance affect the performance of the prediction network.

4.3.2. Prediction

We analyze the TCN in TSEQPREDICTOR in Section 3 comparing with a LSTM model. For this time series forecasting, the prediction network in TSEQPREDICTOR can outperform the LSTM network. Due to the stochastic nature of shocks, the output series from the autoencoder is denoised by the LOESS

Table 3. Results comparison between TSEQPREDICTOR and baseline models. Some models adopt the same architecture of using an autoencoder and a prediction network. These models are named with a '+' sign.

Models	MAE	Precision	Recall	F-1	F-0.2	NSE
MLP	-	0.2631	0.2845	0.2096	0.2494	-1.4739
LSTM	-	0.4596	0.5186	0.3801	0.4058	-0.2059
Conv2D-FC	-	0.4589	0.3963	0.4340	0.4394	-0.1867
Conv2D-LSTM	-	0.4299	0.4069	0.4217	0.4243	-0.4022
ConvLSTM2D-FC	-	0.4633	0.3289	0.3763	0.3801	-0.1714
MLP+MLP	0.2570	0.7525	0.6338	0.6652	0.7113	0.6778
MLP+LSTM	0.1637	0.8420	0.7085	0.7599	0.8021	0.7890
MLP+Conv1D	0.1484	0.8571	0.9351	0.8029	0.8342	0.8133
Conv2d+MLP	0.1484	0.8577	0.7944	0.7887	0.8098	0.8108
Conv2D+LSTM	0.1410	0.8640	0.8776	0.8609	0.8683	0.8222
Conv2D+Conv1D	0.0588	0.9420	0.9115	0.8998	0.8688	0.9293
TSEQPREDICTOR	**0.0483**	**0.9563**	0.9016	**0.9251**	**0.9341**	**0.9323**

Table 4. Ablation study by removing core components in TSEQPREDICTOR.

Models	F-1	NSE
W/O skip connections	0.9001	0.9233
W/O local temporal attention	0.9247	0.9289
TSEQPREDICTOR	**0.9251**	**0.9323**

smoothing method [21]. We summarize the experimental results in Table 3.

4.3.3. Comprehensive Analysis

In this set of experiments, we list several commonly used models for predicting the future main shocks. The results are summarized in Table 3. In this table, MLP represents a three-layer of fully connected neural networks. LSTM represents a two-layer of stateful LSTM neural networks. Conv2D, Conv1D represent a neural network consisting of one 2D-convolutional and 1D-convolutional layer, respectively. From this table, we illustrate TSEQPREDICTOR can outperform a single model significantly and other combination of models for this task.

4.4. TSEQPREDICTOR Ablation Study

In the following two sets of experiments, we demonstrate the two major techniques that can improve the autoencoder and the prediction network: skip connections and local temporal attention. In the first set, we remove the skip connections in the autoencoder and keep the remaining parts the same. In the second set, we remove the local temporal attention in the prediction network and use the same autoencoder as the TSEQPREDICTOR. Table 4 shows the results of these two sets of experiments.

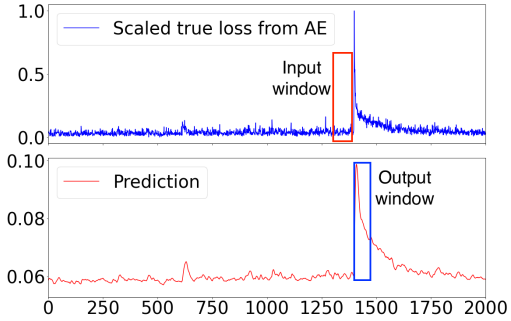


Fig. 4. TSEQPREDICTOR prediction.

4.5. Discussion and Empirical Study

We build joint models as shown in Figure 2, in which the autoencoder can learn the spatio pattern and the predictor can forecast future event. Figure 4 shows a prediction example. Given an input sequence window, the predictor can output a future sequence window, from which a major shock can be detected. There are two aspects in the consideration of this model:

- During the training period, a sequence of T 2D matrices are the input: $X_{t_1}, X_{t_2}, \ldots, X_{t_T}$, and the output is another sequence:$y_{t_2}, y_{t_3}, \ldots, y_{t_{T+1}}$. In this way, the $y_{t_{T+1}}$ is the predicted result. This means that the model can be trained on rolling basis as the data stream in.
- In Southern California, the model can be trained and predict a novelty score which represents the probability of the next major shock. For example, if the input is X_t at t time, the output from the model is X_{t+1} at $t+1$ time. The predicted probability of this area can be told from y_{t+1}.

5. CONCLUSIONS AND FUTURE WORK

In this project, we dissect the problem settings for forecasting earthquakes, discuss how we model spatial temporal forecasting problems using deep neural networks, and propose joint modeling to address this problem. In experiments, we demonstrate some preliminary results of using TSEQPREDICTOR to predict earthquakes in Southern California. According our experiments, we show some promising when proper thresholds are chosen to filter out noisy. In future, we need to consider other physics quantities like seismicity, electric field, magnetic field, deformation which are highly possible correlated to earthquake events.

Code and data availability

The earthquake events dataset used in the paper is available to download from the USGS website at `https://www.usgs.gov/`. Model codes and preprocessed data used in the paper will be published upon acceptance of this manuscript.

6. REFERENCES

[1] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.

[2] Lingxue Zhu and Nikolay Laptev, "Deep and confident prediction for time series at uber," in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2017, pp. 103–110.

[3] Senzhang Wang, Jiannong Cao, and Philip Yu, "Deep learning for spatio-temporal data mining: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[4] S Mostafa Mousavi, William L Ellsworth, Weiqiang Zhu, Lindsay Y Chuang, and Gregory C Beroza, "Earthquake transformer—an attentive deep-learning model for simultaneous earthquake detection and phase picking," *Nature communications*, vol. 11, no. 1, pp. 1–12, 2020.

[5] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[6] W. H. Bakun and C. M. Wentworth, "Estimating earthquake location and magnitude from seismic intensity data," *Bulletin of the Seismological Society of America*, vol. 87, no. 6, pp. 1502–1521, Dec. 1997, Publisher: GeoScienceWorld.

[7] S. A Pulinets, "Strong earthquake prediction possibility with the help of topside sounding from satellites," *Advances in Space Research*, vol. 21, no. 3, pp. 455–458, Jan. 1998.

[8] G. Guangmeng and Y. Jie, "Three attempts of earthquake prediction with satellite cloud images," *Natural Hazards and Earth System Sciences*, vol. 13, no. 1, pp. 91–95, Jan. 2013.

[9] Zahra Alizadeh Zakaria and Farshid Farnood Ahmadi, "Possibility of an earthquake prediction based on monitoring crustal deformation anomalies and thermal anomalies at the epicenter of earthquakes with oblique thrust faulting," *Acta Geophysica*, vol. 68, no. 1, pp. 51–73, Feb. 2020.

[10] Zhiyong Cui, Kristian Henrickson, Ruimin Ke, and Yinhai Wang, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," *IEEE Transactions on Intelligent Transportation Systems*, 2019, Publisher: IEEE.

[11] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu, "Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting," *arXiv:1707.01926 [cs, stat]*, July 2017, arXiv: 1707.01926.

[12] Yasuko Matsubara, Yasushi Sakurai, Willem G. van Panhuis, and Christos Faloutsos, "FUNNEL: automatic mining of spatially coevolving epidemics," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, Aug. 2014, KDD '14, pp. 105–114, Association for Computing Machinery.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*. Springer, 2016, pp. 630–645.

[14] Jining Yan, Lin Mu, Lizhe Wang, Rajiv Ranjan, and Albert Y Zomaya, "temporal convolutional networks for the advance prediction of enso," *Scientific Reports*, vol. 10, no. 1, pp. 1–15, 2020.

[15] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[16] Anastasia Borovykh, Sander Bohte, and Cornelis W Oosterlee, "Conditional time series forecasting with convolutional neural networks," *arXiv preprint arXiv:1703.04691*, 2017.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, pp. 5998–6008, 2017.

[18] Hongyan Hao, Yan Wang, Yudi Xia, Jian Zhao, and Furao Shen, "Temporal convolutional attention-based network for sequence modeling," *arXiv preprint arXiv:2002.12530*, 2020.

[19] Daniel N Moriasi, Jeffrey G Arnold, Michael W Van Liew, Ronald L Bingner, R Daren Harmel, and Tamie L Veith, "Model evaluation guidelines for systematic quantification of accuracy in watershed simulations," *Transactions of the ASABE*, vol. 50, no. 3, pp. 885–900, 2007.

[20] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[21] Jesús Rojo, Rosario Rivero, Jorge Romero-Morte, Federico Fernández-González, and Rosa Pérez-Badia, "Modeling pollen time series using seasonal-trend decomposition procedure based on loess smoothing," *International journal of biometeorology*, vol. 61, no. 2, pp. 335–348, 2017.