FWin Transformer for Dengue Prediction Under Climate and Ocean Influence

Nhat Thanh Tran(✉), Jack Xin, and Guofa Zhou

University of California, Irvine, Irvine, CA 92697, USA
{nhattt,jack.xin,zhoug}@uci.edu

Abstract. Dengue fever is one of the most deadly mosquito-born tropical infectious diseases. Detailed long range forecast model is vital in controlling the spread of disease and making mitigation efforts. In this study, we examine methods used to forecast dengue cases for long range predictions. The dataset consists of local climate/weather in addition to global climate indicators of Singapore from 2000 to 2019. We utilize newly developed deep neural networks to learn the intricate relationship between the features. The baseline models in this study are in the class of recent transformers for long sequence forecasting tasks. We found that a Fourier mixed window attention (FWin) based transformer performed the best in terms of both the mean square error and the maximum absolute error on the long range dengue forecast up to 60 weeks.

Keywords: Dengue forecasting · non-stationary time series · efficient transformers · Fourier mixing

1 Introduction

The modeling and forecasting of vector-borne diseases (VBD) are scientifically challenging and important to the society at large due to their complex non-local temporal dependencies in the data as well as external climate factors. VBD originated from tropical countries pose serious public health risks to both local and global communities. Among them, malaria and dengue fever are the two most deadly mosquito-borne tropical infectious diseases, with about 240 million malaria cases globally and 440,000 malaria deaths annually, and 50–100 million dengue cases. Currently, no tested vaccine or treatment is available to stop or prevent all types of dengue fever. Thus modeling dengue disease evolution is particularly significant.

The correlation of weather/climate with VBD evolution is well-documented ([2,9,12,15,27] among others). With global warming upon us, higher temperatures create more habitats for mosquitoes to infect unexposed human populations and spread diseases. Just in July 2023, about 80 million Americans experienced a heat index of at least 105° according to the National Weather Service. The extreme heat waves prompt irregular typhoons in Asia and flash floods in north America. Such intense precipitation and flooding events become

G. Nicosia et al. (Eds.): LOD 2024/ACAIN 2024, LNCS 15509, pp. 160–175, 2025.
https://doi.org/10.1007/978-3-031-82484-5_12

more frequent and longer (unless humans essentially stop adding carbon dioxide to the atmosphere), behaving as *strongly non-stationary* instead of traditional seasonal signals. They can favor mosquito breeding and survival to further complicate VBD evolution. Besides temperature, ocean currents can also influence the infection dynamics of VBD [2,9].

Knowledge of weather and population susceptibility cycles is known to help prediction, see [5,11] and references therein. A new challenge of VBD modeling is to extract critical information from complex correlations and multiple factors *in the presence of intense transients lasting for week long periods due to extreme climate events*. Ensemble machine learning (ensemble support vector machines [11]), recurrent neural networks (RNN) and regression [5], among other tools [18] have been utilized in the past to study the effects of climate and seasonal weather. However due to either stationary hypothesis or simple decision boundary or one time unit ahead recursion, these existing tools are not well-suited for predicting the aftermath of intense non-stationary behavior in the data.

In this paper, we study a *local-global attention* based *efficient transformer* [21] for *non-stationary VBD modeling and prediction in a specific spatial region*. For simplicity, we leave the spatial synchrony issue [1] (coupling among neighboring regions) to a future study. Transformer networks equipped with attention blocks [22] have powered the recent breakthroughs of natural language processing (NLP and Open AI's Chat-GPT) where long range temporal correlations exist. Since VBD data are not as abundant as in NLP, *light weight and efficient transformer networks* are more suitable and will be our main objects of study here. A successful strategy to reduce computational complexity of transformers is to approximate the full attention by a local attention (e.g. window attention [8]) globalized by a subsequent mixing step (e.g. through shifting [8] or shuffling [3] windows among other treatments). Recently, Fourier transform based mixing has been found competitive in accuracy and efficiency in both training and inference on long sequence time series forecasting tasks. On standard benchmarks [28] as well as highly non-stationary power grid data [26], Fourier-mixed window attention (FWin, [21]) out-performs prob-sparse attention [28] and other recent models such as Autoformer [24], FEDformer [29], ETSformer [23] and PatchTST [16]. We aim to continue this line of inquiry here on multi-variate dengue and climate data in Singapore which provide a real-world VBD data set to help understand and evaluate transformers as a new tool for advancing public health.

The innovations of this paper include a comprehensive transformer based generative model to encompass essential driving mechanisms of VBD evolution and make fast prediction of non-stationary dynamic behavior over a longer temporal duration than existing methods. The ideas of attention and multi-variate data fusion have been applied before in the context of infectious disease prediction, e.g. hand-foot-mouth disease and hepatitis beta virus [25], influenza and dengue [14,30] etc. However, 1) the prior approaches of using a composition of a linear projection and softmax normalization as attention [25,30] or a special form of attention in machine translation [10] adopted in [14] is not robust com-

pared to the canonical quadratic (Q,K,V) form [22]; 2) these methods have not been evaluated on the public long sequence time-series forecast benchmarks [28]; and 3) the models by design can only make one time unit predictions which is not enough for any early warning. Though the attention enhanced LSTM in [14] was extended from 1 month ahead to 2/3/6 month ahead predictions, the performance was increasingly worsening. Methods being a generative transformer can naturally make multi-time unit predictions.

The rest of our paper is organized as follows. In Sect. 2, we discuss the Singapore dengue data set and two prediction tasks. In Sect. 3, we give an overview of transformer models in this paper including FWin transformer. In Sect. 4, we verify the hypothesis of the FWin equivalence theorem. In Sect. 5, we compare results from the transformer models on the two prediction tasks and give our interpretations. In Sect. 6, we provide an ablation study on the choice of window lengths of FWin transformer, and effect of shorter inputs. Concluding remarks are in Sect. 7.

2 Dataset and Task

2.1 Data

The dataset contains 1000 weeks of Singapore's weekly dengue data spanning from 2000 to 2019. The dataset's features include the following variables: the cases number, average temperature, precipitation, Southern Oscillation Index (SOI), Oceanic Niño Index (ONI) total, ONI anomaly, Indian Ocean Dipole (IOD), IOD East, NINO1+2, NINO3, NINO4, NINO3.4, and the respective NINO anomaly. We provide descriptions of important features here: 1) SOI is the difference from average air pressure in the western Pacific, measured in Darwin, Australia, to the difference from average pressure in the central Pacific, measured at Tahiti [7], 2) ONI is running 3-month average sea surface temperatures in the east-central tropical Pacific between 120W-170W [6], 3) IOD is the surface temperature difference between the western and eastern tropical Indian ocean [17], 4) NINO1+2, 3, 3.4, 4 represent the sea surface temperature correponds with the region across the Pacific ocean with coordinates (0-10S, 90W-80W), (5N-5S, 150W-90W), (5N-5S, 170W-120W), (5N-5S, 160E-150W) respectively. Many of the features are reported in a daily or monthly interval. Whenever data are available at a coarser temporal resolution than weekly, we select the data corresponding to the month of the first day of that week. In cases where data is available at a finer temporal resolution than weekly, we calculate the average data for that week. The train/val/test split ratio is 6/2/2. We present a sample of the features of the dataset with the corresponding split ratio in Fig. 1. Data normalization applies to the entire dataset before passing it to the model. This means that each feature in the dataset will have zero mean and variance equal to 1. We label the data set as Singapore Dengue (SD). The processed data is available upon request.

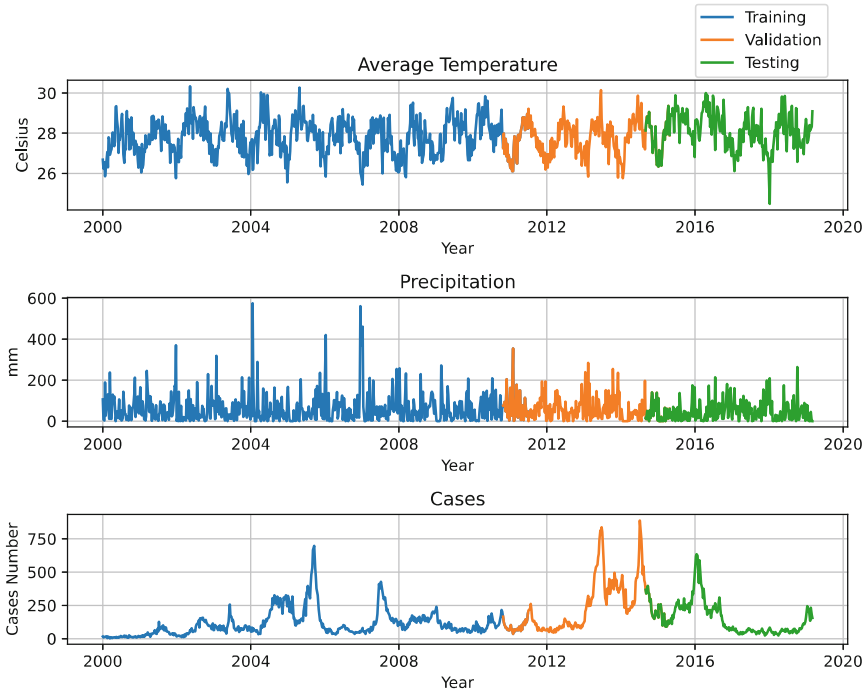

Fig. 1. Sample features of the dataset. The plot includes the average temperature and precipitation from 2000 to 2019 with the dengue cases number. Here blue, orange, green indicate training, validation and testing split respectively. (Color figure online)

2.2 Prediction Task

In this paper, we are interested in predicting the dengue cases number using all provided features. The appropriate task for this is Multivariate to Univariate (MS). For this prediction task, we utilize information from all features from the past m time steps, and predict the number of dengue cases for the next n time steps. In the models, we set m to be 36 by default, and $n \in \{24, 36, 48, 60\}$.

Since we do not need to predict the other features, e.g. precipitation, we introduce a new prediction task where the inputs consist of $m + n$ time steps of the predictor variables (PV). However, the input will only contain m time steps of the response variable (RV), with the RV at the remaining n time steps set to 0. We refer to this task as Modified Multivariate to Univariate (MM). The rationale behind this task is that if we have accurate forecasts for the PV, then we can leverage this information to improve the prediction of the RV. Later, we will show that this modification increases the prediction power of the models. Figure 2 provides an overview of the overall structure of these tasks.

3 Models

Time scale latency between the effect of weather features (such as the abundance of water for larvae, and symptoms of disease in the host) varies depending on the location. This latency could range from up to 6 months of delay to as short

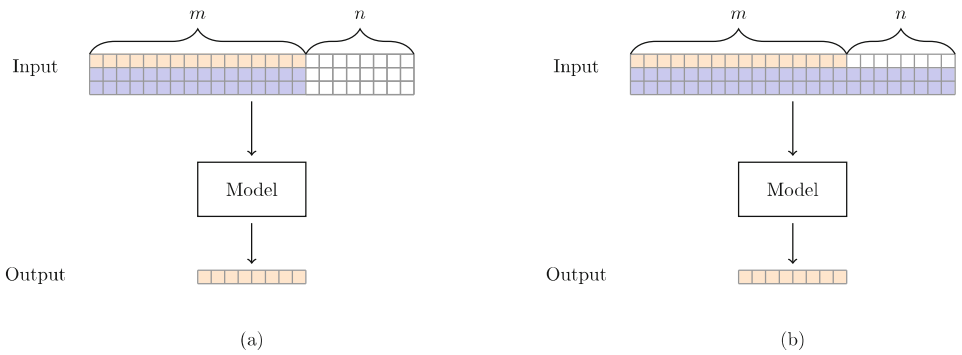

Fig. 2. (a) Input-Output structure of MS task. (b) Input-Output structure of MM task. Here orange shaded cells indicate non-zero value of the RV, blue shaded cells indicate non-zero value of the PV, and white cells indicate zero padded value. (Color figure online)

as just 6 weeks [20]. Thus choosing a lag order (an integer parameter) in a standard statistical model such as ARIMA or VAR is non-trivial for our dataset. Moreover, traditional methods require choosing significant features, i.e. Pearson's correlation, before passing them in to the model [13]. Also features are assumed to be linearly correlated, which may not be true in general. In cases where a nonlinear model is necessary, we must have some functional form to describe the relationship between the climate features and cases number. To address this issue, we will utilize recent deep neural networks, in particular attention based neural networks. We will discuss in detail the structure in the following sections. Below are some reasons why a neural network may resolve some of the issues we have with traditional methods. First, a deep neural network has the ability to extract important features through learning. Second, we treat each time step of the input as a token passing into the neural network. Given a certain time step information (token), the attention mechanism allows a particular token to focus on other tokens, this alleviates the need to pre-define a lag order. Of course, if one has a prior knowledge of a lag order, then it can be incorporated into the model such that one only allows tokens in that particular lag window to affect the final prediction. We will show this as an advantage of FWin. Lastly, other than choosing an architecture for a deep neural network, since in most cases it is a universal approximator, we do not need to have an exact equation to describe the nonlinear relationship between predictor and response variables.

In order to accomplish the dengue forecasting task, we will utilize some of the recently developed deep neural network models. We will compare the following models: FWin [21], Informer [28], FEDformer [29], Autoformer [24], ETSformer [23], and PatchTST [16]. We only include transformer based models in this paper, because these models are the current SOTAs for time series tasks. They demonstrated to out perform statistical models, RNN and CNN on multiple benchmarks [16,28,29].

3.1 Background

Many of the models presented in this paper utilize similar ideas from Transformer [22]. This was originally designed for natural language tasks. In recent years, it was demonstrated that similar approach can be used for time series. We will provide some background information on these models next.

Transformer. Transformer is a basis for many of the newly developed models. Transformer has an encoder-decoder structure. The encoder maps an input x to a representation y. Then the decoder accepts y as an input to generate an output z. Encoder composes of stacks of self attention and feed forward layers. Similarly, decoder composes of stacks of masked self attention, cross attention, and feed forward layers [22]. Attention is an essential component of the Transformer, thus we will discuss it in more detail in the next section.

Attention. Let $x \in \mathbb{R}^{L \times d}$ be the input sequence, where L is the sequence length and d is the feature dimension. Here d loosely can be understood as the number of features in the dataset, usually the input is passed through a linear layer to map the original number of features to a larger dimension. We compute queries (Q), keys (K), values (V):

$$Q = xW_Q + b_Q,\ K = xW_K + b_K,\ V = xW_V + b_V, \tag{1}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d \times d}$ are the weighted matrix, and $b_Q, b_K, b_V \in \mathbb{R}^{L \times d}$ are the bias matrix. Attention is defined as:

$$A(Q, K, V) = \text{softmax}(QK^T/\sqrt{d})V, \tag{2}$$

where A is the attention function and the softmax function applies along the row dimension [22]. We usually refer to this calculation as full self attention, because the queries, keys, and values are linear projections of the same input, and there is a full matrix multiplication of Q and K^T. However, for cross attention, the query comes from the linear projection of decoder input, while the keys and values come from encoder output. For causality, masked attention is a restriction to the self attention calculation where we prohibit the interaction of the current query to future keys.

For window attention [8,21], one computes the attention on sub-sequences then concatenate the results together. We first divide sequence x into N subsequences: $x^{(1)}, x^{(2)}, \ldots, x^{(N)}$, such that $x = [x^{(1)}, x^{(2)}, \ldots, x^{(N)}]^T$. Each $x^{(i)} \in \mathbb{R}^{L/N \times d}$ for $i = 1, 2, \ldots, N$, where $N = L/w$, w is a fixed window size. This implies we divide the queries, keys and values as follow $Q = [Q^{(1)}, Q^{(2)}, \ldots, Q^{(N)}]^T$, $K = [K^{(1)}, K^{(2)}, \ldots, K^{(N)}]^T$, $V = [V^{(1)}, V^{(2)}, \ldots, V^{(N)}]^T$. Thus we compute attention for each subsequence as follows:

$$A(Q^{(i)}, K^{(i)}, V^{(i)}) = \text{softmax}\left(Q^{(i)}K^{(i)T}/\sqrt{d}\right)V^{(i)}. \tag{3}$$

After computing the attention for each sub-sequence, we concatenate the sub-attentions to form the window attention:

$$A_w(Q, K, V) = \begin{bmatrix} A(Q^{(1)}, K^{(1)}, V^{(1)}) \\ \vdots \\ A(Q^{(N)}, K^{(N)}, V^{(N)}) \end{bmatrix}. \tag{4}$$

3.2 Fourier Mix

The FWin model [21] uses window attention in place of full attention to reduce the computational complexity. However, this limits the interactions between the tokens, which may lead to degradation in performance. To resolve this issue, FWin utilizes the fast Fourier transform as a way to mix tokens among the windows. Given input $x \in \mathbb{R}^{L \times d}$, one computes Fourier transform along the feature dimension (d), then along the time dimension (L), finally taking real part to arrive at:

$$y = \mathcal{R}(\mathcal{F}_{\text{time}}(\mathcal{F}_{\text{feature}}(x))), \tag{5}$$

where $\mathcal{F}$ is 1D discrete Fourier transform, and $\mathcal{R}$ is the real part of a complex number [21].

3.3 FWin Overview

FWin [21] performs the best among all of the models. In addition, the procedure is general across different models. We will present FWin model structure in more detail. See Fig. 3 for the model overview. FWin is an adaptation of Informer [28] by employing the window attention [8] mechanism to capture the local information and Fourier Mix layer [4] to mix tokens among the windows. FWin has an encoder-decoder structure. First, raw input passes into the Encoder Input layer to embed the time, and positional information. In the encoder, the input first passes through the window attention, then dimensional reduction layer of Distilling Operation. This layer composes of convolution and MaxPool operations. Lastly, the tokens mix by the Fourier Mix layer, and go toward to the decoder. Second, the raw input passes into the Decoder Input layer with the time and positional information added into the input. In the decoder, the input passes through a masked window attention to respect causality. Then the tokens mix by the Fourier Mix layer before passing through the cross attention block, and finally through a Fully Connected Layer (i.e. a linear projection to return an output with correct dimension) to produce the output. The input and output dimensions are the same as shown in Fig. 2.

FWin employs window attention instead of full attention. This led to computational savings as token interactions are localized in a window. This implies that selecting a window size has a similar effect to choosing a lag order in a vector autoregression (VAR) model. We will delve into this effect in more detail in the ablation study section.

3.4 Other Models

FWin is derived from the Informer whose structure is similar to Fig. 3. The main difference is that Informer uses the so called Probsparse attention instead of window attention and Fourier mix. Probsparse attention [28] relies on a sparse query measurement function (an analytical approximation of Kullback-Leibler divergence) so that each key attends to only a few top queries.

FEDformer uses an encoder-decoder structure as well, however, instead of canonical attention, it uses the frequency enhanced attention. It applies Fourier transform to the input, then select a few modes to compute attention. In addition, it incorporates a seasonal-trend decomposition layers to capture the global properties of time series [29]. Similarly, Autoformer uses Series Decomposition and Auto-Correlation instead [24]. ETSformer uses similar structure with frequency attention and exponential smoothing attention to extract growth and seasonal information from the inputs. Here the attention is weighted with an exponential term that favors nearby tokens. Then it uses such information selection in the decoder to forecast the future horizon [23]. Lastly, PatchTST only uses the encoder structure of the Transformer. It first divides the input into patches, and passes each feature independently through the Transformer's encoder, then concatenate the outputs to form the final prediction [16].

Some of the models only make multivariate prediction. Therefore, to generate univariate predictions, we simply extract the response feature from the model's output. This approach is standard for these types of problems.

3.5 Models Hyperameters

For all of the models in this paper, we used their default hyper-parameters. For FWin we use a window size of 12, and cross attention window number is 3. The total number of epochs is 6 with early stopping. We used the Adam optimizer, and the learning rate starts at $1e^{-4}$, decaying two times smaller every epoch. For ETSformer we used the initial learning rate of $1e^{-3}$ in the exponential with a warm up learning rate schedule. For PatchTST we used the constant learning rate of $2.5e^{-3}$ and 100 training epochs with early stopping.

4 FWin Equivalency Condition

FWin comes with an equivalence theorem for its estimator that is summarized below. For the detailed proof, we refer to [21].

Theorem 1. *Let $Q, K, V \in \mathbb{R}^{L\times d}$ and $w \in \mathbb{N}$ be a factor of L. If $Attn(Q, K)$ is block diagonally invertible, then there exists a matrix $C \in \mathbb{C}^{L\times L}$ such that*

$$A(Q, K, V) = C \cdot \mathcal{F}(A_w(Q, K, V)). \tag{6}$$

Here $Attn(Q, K) = softmax(QK^T/\sqrt{d})$, $\mathcal{F}$ is the discrete Fourier transform, $\cdot$ denotes matrix multiplication, $A(\cdot)$ and $A_w(\cdot)$ are functions define in Eq. 2 and Eq. 4 respectively.

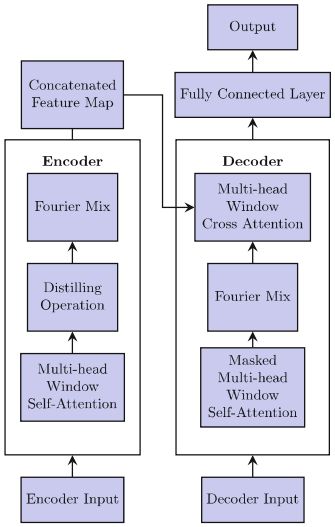

Fig. 3. FWin model overview [21].

A matrix B is block diagonally invertible if each block diagonal sub-matrix is invertible [21].

To verify the condition of the theorem we will follow the procedure: 1) Run simulations using the Informer model with full attention instead of probsparse on the testing set. 2) Collect the full attention matrix of the first encoder block of the Informer. 3) For a fixed window size w, compute the condition number of each block sub-matrix of size $w \times w$. 4) If the condition number is finite, then the sub-matrix satisfies the condition.

From Fig. 4, we verify that dengue dataset satisfies Theorem 1. All of the condition numbers are finite under various window sizes, most of them are relatively small (less than 10^4). We noted that as window size increases, the condition numbers increase.

5 Results and Discussion

We present a summary of all the prediction tasks on the Singapore dataset in Table 1. MAE $= \frac{1}{n}\sum_{i=1}^{n}|y-\hat{y}|$ and MSE $= \frac{1}{n}\sum_{i=1}^{n}(y-\hat{y})^2$ serve as evaluation metrics. The results are the average of five independent simulations. The best results are highlighted in boldface, and the total count at the bottom of the table indicates how many times a particular method outperforms the others per metric per task. From Table 1, we observe that FWin has the best performance on both tasks.

In addition, we also demonstrate that including future PV information (e.g. climate and ocean current features) in the model increases FWin performance (on dengue cases) significantly. We observed that for longer time forecasting, i.e. metrics of 36, 48, 60, the error for MM task is lower than MS task. However, for Informer and PatchTST, the models' performances decrease with additional information. An explanation for the degradation in performance is as follows: 1)

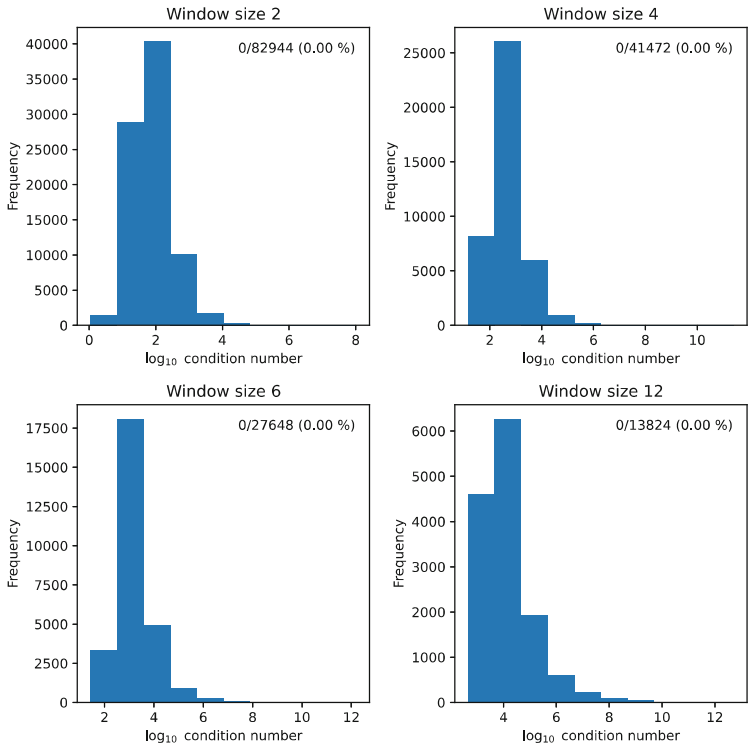

Fig. 4. Condition numbers of block sub-matrices with various sizes of attention matrix obtained from the first encoder block of Informer model using full attention on the testing data. On the top right corner of each subplot, there is a label "n/m (k%)"; here m is the total number of condition numbers, n is the infinite condition numbers, and k is the corresponding percentage.

the Informer's probsparse attention only chooses a number of queries to compute attention, which means if the input length increases then a few top queries may not capture the majority of the contribution. 2) For PatchTST, it initially patches the input with overlaps, thus increasing the sequence length may cause too much overlap, leading to an overflow of information. In addition, it treats each feature independently, thus adding additional information for the PV may not affect the RV. In case of Autoformer and ETSformer, including future predictor (e.g. climate and ocean current) information is beneficial to model prediction power. In particular, ETSformer's error reduces significantly for MM task compared to the MS task. However, FEDformer was not able to perform the MM task because the requirement of the input length of the decoder need to be half of the input of the sequence.

Furthermore, we compared the vector autoregression (VAR) to DNN models. VAR depends on a choice of lag order, which is non-trivial. Opting for a large lag order can lead to the model making wild predictions, resulting in large errors. On the other hand, selecting a small lag order can significantly reduce errors, but the prediction may appear too smooth, which is unrealistic given the behavior of the dataset. Additionally, the overall error was higher than FWin, thus we do

not present the full VAR results obtained from calling the VAR library of the package statsmodels [19].

We present a sample of the prediction of various models in Fig. 5 for the MM tasks. The $x-$axis represents the timescale in weeks, and $y-$axis is the normalized number of cases. FWin performs the best in terms of metrics presented and visual. It was able to predict a large drop in number of cases while many other models were unable to do so. We noted that in the MM task, the model input is richer, containing the most information. This confirm the results we obtained in Table 1.

Table 1. Accuracy comparison on Singapore data with input length of 36, best results highlighted in bold. Here MS, MM are multivariate to univariate, and modified multivariate to univariate respectively. SD is short for Singapore Dengue, and "-" indicates that the method is inapplicable for the task.

Methods		FWin		Informer		FEDformer		Autoformer		ETSformer		PatchTST	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
SD (MS)	24	**1.170**	**0.684**	1.842	0.877	2.090	1.085	2.237	1.127	1.611	0.856	1.273	**0.684**
	36	**1.450**	**0.725**	1.974	0.942	2.455	1.187	2.441	1.198	1.823	0.938	1.664	0.815
	48	**1.782**	**0.830**	2.043	0.958	2.912	1.334	2.927	1.370	2.271	1.063	1.887	0.915
	60	**1.518**	**0.826**	1.784	0.910	2.938	1.358	3.015	1.395	2.379	1.128	2.390	1.107
SD (MM)	24	**1.303**	**0.787**	2.137	1.002	-	-	1.951	1.055	1.480	0.885	1.734	0.833
	36	**1.156**	**0.680**	2.043	0.968	-	-	2.136	1.105	1.305	0.774	1.994	0.937
	48	**1.251**	**0.693**	2.228	1.033	-	-	2.238	1.152	1.355	0.825	2.352	1.062
	60	**1.124**	**0.710**	1.911	0.946	-	-	2.680	1.272	1.295	0.781	2.579	1.117
Count		16		0		0		0		0		1	

6 Ablation Study

In this section, we will examine the effect of window size of the FWin model on its performance. We can think of the window size of FWin as prior knowledge of the time delay effect of weather/climate on the dengue cases. Due to the restricted interaction within each window, the cases number in a particular window only attends to local weather information. For this experiment, we utilize FWin with window sizes of 1, 2, 4, 6, 12, and 18.

From Table 2, we observe that the biggest window size of the task gives the best performance in most cases. This is intuitively consistent with the design of FWin where we expect that the larger the window size, the better the overall performance. In addition, we observe that for the smallest window size of 1, the errors are significantly lower than naively expected. In particular, under the MAE metric, the window size of 18 performs the best, while under the

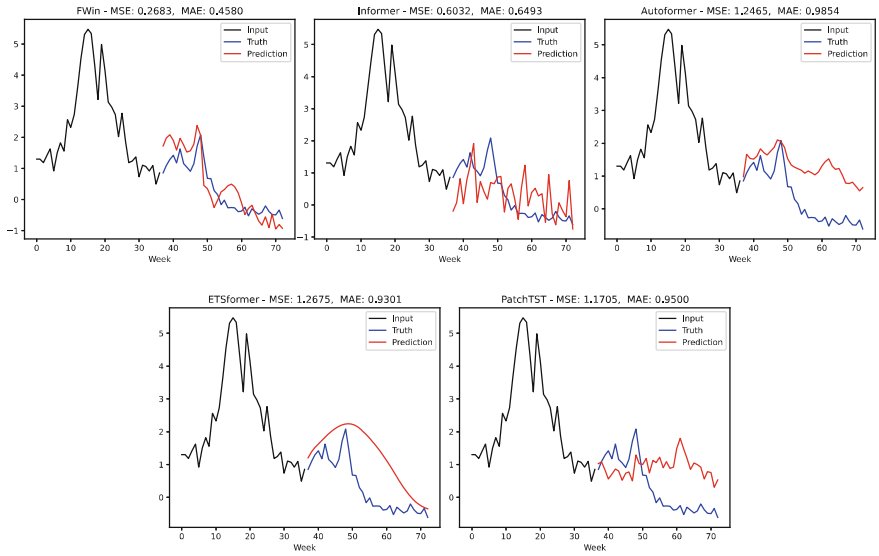

Fig. 5. Sample models' prediction for MM task. The x-axis represents time in weeks, and y-axis is the normalized cases number. The suptitle includes the model name and the prediction errors (MSE and MAE). In black is the case number input, blue is the ground truth and red is the prediction of the model.

MSE metric, a window size of 1 is the best for MS task. MSE amplifies the effects of large errors while suppressing those from the small errors. Thus under this metric, the smallest window size of 1 does not result in many large errors overall. On the other hand, MAE penalizes all error types equally, thus implying that the window size of 18 mostly results in small errors. We observe from the proof of FWin attention equivalency with the full attention (Theorem 5.6 in [21]) that the theorem is true without any assumption on the structure of the attention matrix if the window size is 1. This provides an explanation for why we encountered low MSE error for window size of 1, i.e. the model does not make many large errors in the prediction. On the other hand for MM task, since we incorporate more information in the input, the largest window size of 12 performs the best. Moreover, except for the metric (prediction length) of 24, MM task performs better than MS task. This indicates that additional information is useful. An explanation for this phenomena is if the dependency between the RV (dengue cases) and the PV (precipitation) is short, i.e. 6 weeks, then the model performance may degrade at later time steps.

In addition, we performed further experiments to understand the effect of shorter input sequence length to model performance. For this purpose, we reduced the input sequence length of the model from the default value of 36 to 18. From Table 3 we observe that for the shortest prediction length, PatchTST performs the best. However, for longer prediction lengths, FWin exhibits the best results. In general, the errors remain similar to their counterparts when the input length is 36. Therefore FWin is robust under the variation to shorter

Table 2. Accuracy comparison of FWin model using different window sizes with input length of 36. Here MS, MM are multivariate to univariate, and modified multivariate to univariate respectively. Best results highlighted in bold. "-" denotes window size is inapplicable.

Window Size		1		2		4		6		12		18	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
SD (MS)	24	1.195	0.705	1.282	0.720	1.284	0.877	1.223	0.703	**1.181**	0.678	1.196	**0.667**
	36	**1.432**	0.742	1.507	0.754	1.517	0.755	1.492	0.747	1.458	0.725	1.440	**0.708**
	48	**1.673**	0.828	1.854	0.866	1.821	0.859	1.836	0.862	1.765	0.824	1.756	**0.817**
	60	**1.538**	0.848	1.552	**0.805**	1.555	0.810	1.622	0.830	1.592	0.813	1.598	0.809
SD (MM)	24	**1.354**	**0.795**	1.367	0.802	1.368	0.802	1.351	0.798	**1.354**	0.807	-	-
	36	1.306	0.726	1.326	0.740	1.331	0.738	1.286	0.726	**1.185**	**0.695**	1.216	0.718
	48	1.523	0.815	1.502	0.808	1.504	0.807	1.534	0.839	**1.270**	**0.719**	-	-
	60	1.339	0.802	1.338	0.805	1.328	0.800	1.312	0.796	**1.140**	**0.724**	-	-
Count		5		1		0		0		8		3	

Table 3. Accuracy comparison on Singapore data with input length of 18, best results highlighted in bold. Here MS, MM are multivariate to univariate, and modified multivariate to univariate respectively.

Methods		FWin		Informer		FEDformer		Autoformer		ETSformer		PatchTST	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
SD (MS)	24	1.388	0.673	1.899	0.888	1.650	0.890	1.734	0.937	1.476	0.839	**1.040**	**0.621**
	36	**1.444**	**0.698**	1.903	0.921	2.169	1.021	2.267	1.064	1.722	0.908	1.542	0.772
	48	**1.564**	**0.771**	2.045	0.983	2.863	1.241	2.930	1.275	2.147	1.048	1.972	0.932
	60	**1.281**	**0.666**	1.811	0.928	2.992	1.282	3.030	1.297	2.169	1.073	2.282	1.050
SD (MM)	24	1.669	0.863	2.250	1.017	-	-	2.075	1.189	1.807	0.944	**1.340**	**0.765**
	36	**1.479**	**0.839**	2.225	1.004	-	-	2.451	1.309	1.995	1.009	1.650	0.872
	48	**1.511**	**0.822**	2.307	1.058	-	-	2.687	1.364	1.726	0.890	1.800	0.940
	60	**1.187**	**0.696**	1.952	0.966	-	-	2.803	1.312	1.462	0.844	1.570	0.864
Count		12		0		0		0		0		4	

input lengths. Moreover, the MM tasks perform better than their counterparts on longer prediction lengths for all models except Informer.

7 Conclusions

We evaluated various attention-based deep neural networks for predicting dengue cases in Singapore. The dataset contains multiple features, including average temperature, precipitation, and global climate indices such as Southern Oscillation Index, Oceanic Niño Index, and Indian Ocean Dipole. Among the models

investigated in the study, FWin demonstrated the highest prediction accuracy overall. Moreover, incorporating future climate/weather/ocean information in the modified multivariate to univariate task generally improved dengue case prediction for many models considered.

For subsequent work, we plan to design and embed a more explicit climate and disease correlation layer (with help of certain prior knowledge) into the FWin model to enhance its performance. In addition, we plan to develop spatial-temporal transformer models that take into account geographical information and predict disease cases over multiple regions in countries bordering the Indian and Pacific oceans.

Acknowledgements. The work was partially supported by NSF grants DMS-2219904 and DMS-2151235.

Declaration of Competing Interest. The authors declare that there are no competing interest.

References

1. Bjørnstad, O., Ims, R., Lambin, X.: Spatial population dynamics: analyzing patterns and processes of population synchrony. Tree **14**(11), 427–432 (1999)
2. Chuang, T.W., Chaves, L.F., Chen, P.J.: Effects of local and regional climatic fluctuations on unprecedented dengue outbreaks in southern Taiwan. PLoS ONE **12**(7), e0181638 (2017)
3. Huang, Z., Ben, Y., Luo, G., Cheng, P., Yu, G., Fu, B.: Shuffle transformer: rethinking spatial shuffle for vision transformer. arXiv:2106.03650 (2021)
4. Lee-Thorp, J., Ainslie, J., Eckstein, I., Ontanon, S.: Fnet: mixing tokens with Fourier transforms. In: Proceedings of Conference North American Chapter of Association Computational Linguistics: Human Language Technologies, pp. 4296 – 4313 (2022)
5. Li, Z., Xin, J., Zhou, G.: An integrated recurrent neural network and regression model with spatial and climatic couplings for vector-borne disease dynamics. In: Proceedings of International Conference on Pattern Recognition Applications Methods, pp. 505–510 (2022)
6. Lindsey, R.: Climate variability: oceanic niño index (2009). https://www.climate.gov/news-features/understanding-climate/climate-variability-oceanic-nino-index. Accessed 01 Mar 2024
7. Lindsey, R.: Climate variability: Southern oscillation index (2009). https://www.climate.gov/news-features/understanding-climate/climate-variability-southern-oscillation-index. Accessed 01 Mar 2024
8. Liu, Z., et al.: Swin transformer: hierarchical vision transformer using shifted windows. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12009–12019 (2022)
9. Liyanage, P., Tozan, Y., Overgaard, H., Tissera, H., Rocklöv, J.: Effect of El Niño-southern oscillation and local weather on Aedes vector activity from 2010 to 2018 in Kalutara district, Sri Lanka: a two-stage hierarchical analysis. Lancet Planetary Health **6**, e577–e585 (2022)

10. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421 (2015)
11. McGough, S., Clemente, L., Kutz, J.N., Santillana, M.: A dynamic, ensemble learning approach to forecast dengue fever epidemic years in Brazil using weather and population susceptibility cycles. J. R. Soc. Interface **18**, 20201006 (2021)
12. Morin, C., Comrie, A., Ernst, K.: Climate and dengue transmission: evidence and implications. Environ. Health Perspect. **121**, 1264–1272 (2013)
13. Nejad, F., Varathan, K.: Identification of significant climatic risk factors and machine learning models in dengue outbreak prediction. BMC Med. Inform. Decis. Making **21** (2021)
14. Nguyen, H., et al.: Deep learning models for forecasting dengue fever based on climate data in Vietnam. PLOS Neglected Tropical Diseases (2022). https://doi.org/10.1371/journal.pntd.0010509
15. Nguyen, L., Le, H., Nguyen, D., Ho, H., Chuang, T.W.: Impact of climate variability and abundance of mosquitoes on dengue transmission in central Vietnam. J. Environ. Res. Public Health **17**(7), 2453 (2020)
16. Nie, Y., Nguyen, N.H., Sinthong, P., Kalagnanam, J.: A time series is worth 64 words: long-term forecasting with transformers. In: ICLR (2023). https://openreview.net/forum?id=Jbdc0vTOcol
17. Saji, N., Goswami, B., Vinayachandran, P., Yamagata, T.: A dipole mode in the tropical Indian ocean. Nature (1999)
18. Santangelo, O., Gentile, V., S. Pizzo, D.G., Cedrone, F.: Machine learning and prediction of infectious disease: a systematic review. Mach. Learn. Knowl. Extr. **5**(1), 175–198 (2023)
19. Seabold, S., Perktold, J.: Statsmodels: econometric and statistical modeling with python. In: 9th Python in Science Conference (2010)
20. Titus Muurlink, O., Stephenson, P., Islam, M.Z., Taylor-Robinson, A.W.: Long-term predictors of dengue outbreaks in Bangladesh: a data mining approach. Infectious Dis. Model. **3**, 322–330 (2018)
21. Tran, N.T., Xin, J.: Fourier-mixed window attention: accelerating informer for long sequence time-series forecasting. arXiv:2307.00493 (2023)
22. Vaswani, A., et al.: Attention is all you need. In: NeurIPS, vol. 30 (2017)
23. Woo, G., Liu, C., Sahoo, D., Kumar, A., Hoi, S.: Etsformer: exponential smoothing transformers for time-series forecasting. arXiv:2202.01381 (2022)
24. Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. In: NeurIPS (2021)
25. Zhang, P., Wang, Z., Chao, G., Huang, Y., Yan, J.: An oriented attention model for infectious disease cases prediction. In: Theory and Practices in Artificial Intelligence; Lecture Notes in Computer Science, vol. 13343 (2022)
26. Zheng, Y., Hu, C., Lin, G., Yue, M., Wang, B., Xin, J.: Glassoformer: a query-sparse transformer for post-fault power grid voltage prediction. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 3968–3972 (2022)
27. Zhou, G., Minakawa, N., Githeko, A., Yan, G.: Association between climate variability and malaria epidemics in the east African highlands. Proc. Natl. Acad. Sci. **101**(8), 2375–2380 (2004)
28. Zhou, H., et al.: Informer: beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of AAAI, vol. 35, pp. 11106–11115 (2021)

29. Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R.: Fedformer: frequency enhanced decomposed transformer for long-term series forecasting. In: ICML (2022)
30. Zhu, X., et al.: Attention-based recurrent neural network for influenza epidemic prediction. BMC Bioinform. 20-S(18):art. no. 575, 1–10 (2019)