Interpreting County-Level COVID-19 Infections using Deep Learning for Time Series

Abstract—**Deep Learning for Time-series plays a key role in AI for healthcare. To predict the progress of infectious disease outbreaks and demonstrate clear population-level impact, more granular analyses are urgently needed that control for important and potentially confounding county-level socioeconomic and health factors. We forecast US county-level COVID-19 infections using the Temporal Fusion Transformer (TFT). We focus on heterogeneous time-series deep learning model prediction while interpreting the complex spatiotemporal features learned from the data. The significance of the work is grounded in a real-world COVID-19 infection prediction with highly non-stationary, finely granular, and heterogeneous data. 1) Our model can capture the detailed daily changes of temporal and spatial model behaviors and achieves better prediction performance compared to other time-series models. 2) We analyzed the attention patterns from TFT to interpret the temporal and spatial patterns learned by the model. 3) We collected around 2.5 years of socioeconomic and health features for 3142 US counties, such as observed cases, and a number of static (age distribution and health disparity) and dynamic features (vaccination, disease spread, transmissible cases, and social distancing). Using the proposed framework, we have shown that our model can learn complex interactions. Interpreting different impacts at the county level would be crucial for understanding the infection process that can help effective public health decision-making.**

Index Terms—**Time Series Deep Learning, Interpretability, Temporal Fusion Transformer, Spatiotemporal, Attention, County-Level COVID-19 prediction.**

I. Introduction

The rapid spread of coronavirus disease has had a profound impact on the human population, making accurate forecasting of infection cases critical for administrative planning and resource allocation for public health. To achieve this, it is necessary to conduct a detailed study of trends and develop short- and long-term prediction tools that can evaluate infection processes at different speeds and scales in various geographic regions. Several features of the current pandemic make it challenging to develop successful time-series forecasting models. Firstly, there is no prior epidemiological knowledge of the disease dynamics to base models. Secondly, the data generation process and disease influences, policies, and individual behaviors are non-stationary, making it difficult to generate accurate predictions. Additionally, the data sources are often noisy due to reporting issues or undocumented infections. Finally, beyond accuracy, it is crucial to have explainable models that can help interpret the results meaningfully for the healthcare system, policymakers, and the public.

Many forecasting models such as SEIR [1] [2], and machine learning autoregressive models including ARIMA [3] [4] [5] have been used for COVID-19 spread forecasting. Additionally, deep learning models, such as Long Short Term Memory (LSTM) networks [6], Gated Recurrent Unit (GRU) [7], CNN [8], and attention-based networks, such as Transformer [9] [10], have been applied to further improve COVID-19 forecasting. To our knowledge, however, no prior studies have utilized an interpretable attention mechanism to quantitatively analyze both spatial and temporal patterns of infection cases at the US county level. Although many state-level, county-level and Points of Interest (POI) [11] studies have been reported, prior works have mostly focused on forecasting. However, forecasting COVID-19 at a fine-grained level, such as the county level, is challenging due to the diverse population sizes, socio-economic differences, and lack of data availability. Furthermore, non-stationary time series (with their distribution drifting over time) [12] or time series with extreme events [13] or unknown events like COVID variants are particularly challenging to model and interpret.

To address the aforementioned gap, we propose the use of the Temporal Fusion Transformer (TFT) model [14] to forecast COVID-19 infections and interpret the model's predictions by extracting its attention weights. This study focuses on forecasting COVID-19 infections at the daily level for 3,142 US counties. To achieve this, we collected a comprehensive set of county-level input features spanning 2.5 years. Our results demonstrate that the proposed TFT model outperforms related works in terms of prediction accuracy and can extract interesting temporal and spatial patterns from raw data. In summary, the main contributions of our research are:

- We collected *static covariates*, *observed inputs*, and *known future inputs*, for 3,142 US counties from February 29, 2020, to May 17, 2022. We performed data cleaning to remove outliers and ensure the quality of the dataset.
- We compared our proposed Temporal Fusion Transformer (TFT) Model with four other deep learning time-series models in a multivariate multi-horizon setting. Our experimental results showed that the TFT model outperforms the other models across all five evaluation metrics used in this work, demonstrating its effectiveness in predicting COVID-19 infection cases at the county level.
- To gain deeper insights into the spatiotemporal patterns learned by the TFT model, we analyzed its predictions and self-attention weights. We demonstrated that the model can capture temporal patterns, including infection trends, seasonality, and holiday effects in a meaningful way. Moreover, the model's attention mechanism en-

ables it to focus on more infection-affected geographical regions and perform well across different population demographics.

- Moreover, to facilitate reproducibility and further research in this area, we will make our code available on GitHub.

The rest of this paper is organized as follows. Section II presents details on the data collection process, feature descriptions, and preprocessing. In Section III, we define the problem statement and provide the necessary mathematical formulations. Next, Section IV provides an overview of the TFT model architecture and its interpretable self-attention mechanism. We then describe the experimental setups in Section V and provide a comprehensive comparison of our proposed TFT model with related works in Section VI. The spatiotemporal patterns and important insights learned by the model are analyzed in Sections VII and VIII, respectively. Section IX presents the input feature importance from TFT. Related works are discussed in Section X, and we conclude this paper with Section XI, where we provide concluding remarks, lessons learned, and discuss the possible impact on future work.

II. Input Data and Features

In this section, we describe the data collection process, feature description, and data preprocessing steps.

A. *Data Collection*

We collected our dataset from multiple sources, including CDC (Centers for Disease Control and Prevention), USA Facts [15], and Unacast [16]. The collected data covers a time frame from February 29, 2020, to May 17, 2022, and includes a wide range of county-level features. However, for the vaccination feature, we used data from the CDC [17] starting from December 14, 2020, when the US initiated a nationwide COVID-19 vaccination campaign, as it was the earliest available data.

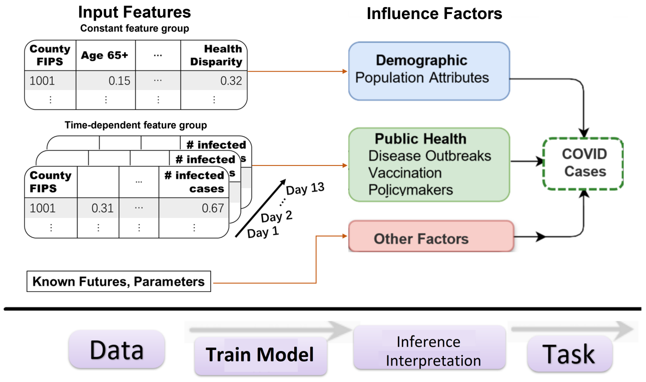


Fig. 1: The feature groups and influencing factors.

Fig.1 summarizes the feature groups with the influencing factors they capture and the county characteristics they represent. In all associated features, the county FIPS codes are used as unique identifiers for those geographic areas. Table I lists the features with respective sources and descriptions.

B. *Data Preprocessing*

To ensure the quality of our data, we removed outliers caused by rare events or human errors during the data collection process. We applied the following thresholds to identify outliers:

$$\begin{aligned} \text{lower} &= Q1 - (7.5 * IQR) \\ \text{upper} &= Q3 + (7.5 * IQR) \end{aligned} \tag{1}$$

where $Q1$ and $Q3$ represent the first and third percentiles, and IQR is the interquartile range. The data statistics before and after removing the outliers are presented in Table II, which demonstrates the effectiveness of our outlier removal process.

TABLE II: Statistics of input features.

Feature	**Original**		**Cleaned**	
	Mean	**Std**	**Mean**	**Std**
Case (Target)	**31.67**	**337.4**	**27.18**	**174.2**
Age Distribution	0.576	0.094	0.576	0.094
Health Disparities	0.368	0.198	0.368	0.198
Vaccination	**20.61**	**22.92**	**20.61**	**22.92**
Disease Spread	0.150	0.194	0.150	0.193
Social Distancing	0.784	0.228	0.795	0.229
Transmissible Cases	0.492	0.210	0.491	0.210

The large standard deviation of the ”Cases” feature, as shown in Table II, indicates that the data is highly volatile, with a 10x and 5x difference for raw and outlier-removed data, respectively. It is worth noting that we opted not to apply a moving average to smooth the dataset further, as doing so would filter out any seasonal patterns present in the daily raw data. Prior to fitting the data to our models, we normalized both the input and target features using standard scaling techniques.

C. *Ground Truth*

The ground truth data in our dataset exhibits three distinct waves, which are depicted in Fig. 2. These waves were identified by a surge in the number of COVID-19 cases caused by new variants of the virus.

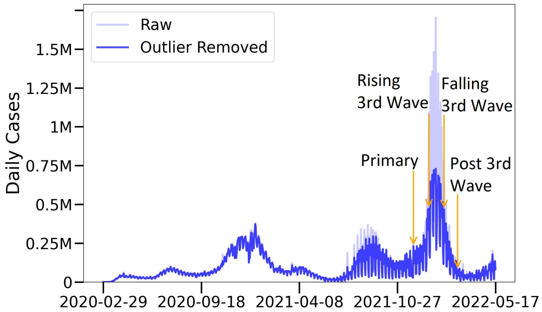


Fig. 2: Ground truth of reported COVID-19 cases [15] along with dataset splits in the three waves.

The first wave occurred between October 2020 and March 2021, followed by a second wave between July and October

TABLE I: Description of input feature groups and targets.

Input Type	Feature	Description	Data Source
Target	Cases	Daily COVID-19 cases	USA Facts [15]
Static	Age Distribution Health Disparities	Percentage of population aged 65 or older Uninsured population percent and socioeconomic status	SVI [18]
Observed	Vaccination Disease Spread Transmissible Cases Social Distancing	Percentage of population fully vaccinated Fraction of total cases from the last 13 days (one incubation period) Population size divided by cases from the last 13 days Change in distance travelled relative to baseline(previous year), based on cell phone mobility data	CDC [17] USA Facts [15] USA Facts [15] Unacast [16]
Known Future	SinWeekly CosWeekly	sin (day of the week/7) cos (day of the week/7)	Date

2021. The third wave began in December 2021 and lasted until March 2022, reaching its peak around January 15, 2022, due to the emergence of more virulent strains such as Delta and Omicron [15]. We refer to this period as the third wave, consistent with previous studies [19]. To evaluate the performance of our model, we divided the data into different phases near the third wave, as illustrated in Fig. 2, and described in detail in Table III. Demonstrating the model's ability to perform well in the face of changing trends such as these is essential to establishing its generalizability and robustness.

III. Problem Statement

Our goal is to create a deep learning model that can accurately predict daily COVID-19 cases for each of the 3142 counties in the United States. To achieve this, we use a multivariate multi-horizon approach that integrates heterogeneous types of inputs for each county. Our prediction model denoted as $f(.)$, is defined as follows:

$$\hat{y}_i(t,\tau) = f(\tau, y_{i,t-k:t}, \mathbf{z}_{i,t-k:t}, \mathbf{x}_{i,t-k:t+\tau}, \mathbf{s}_i) \qquad (2)$$

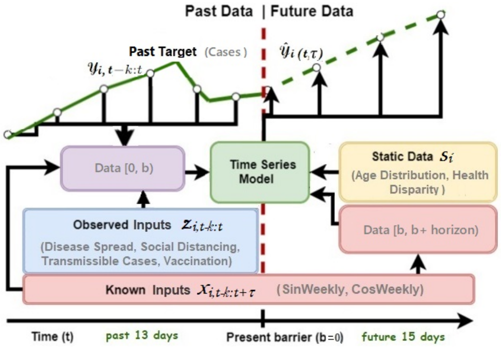


Fig. 3: Time-series forecasting with static covariates, observed inputs, and known future events.

where $\hat{y}_i(t,\tau)$ represents the predicted number of cases at time $t \in [0, T_i]$ for county i, τ days into the future. T_i is the length of the time series period, which for our case is the same for each county. For instance, we use the previous 13 days of data to forecast the future 15 days. Our approach employs the Temporal Fusion Transformer (TFT) as the primary time-series model. Fig. 3 provides a high-level overview of data model preparation for our research. Moreover, we seek to understand the model's inner workings and interpretability through attention-based analysis.

For each unique county i in our dataset, we associate a time series model that takes three types of covariates as inputs:

1) **Static Inputs**: Each county i is associated with a set of static inputs $\mathbf{s}_i$, which do not vary over time and are specific to that county's demographics.
2) **Observed or Past Inputs**: Observed inputs are time-varying features known at each timestamp $t \in [0, T_i]$ (e.g., Vaccination, Disease Spread, Social Distancing, Transmissible Cases), but their future values are unknown. We incorporate all past information within a look-back window k (past 13 days), using target (cases) and observed inputs upto the forecast start time t ($y_{i,t-k:t} = \{y_{i,t-k}, \cdots, y_{i,t}\}$ and $\mathbf{z}_{i,t-k:t} = \{\mathbf{z}_{i,t-k}, \cdots, \mathbf{z}_{i,t}\}$).
3) **Known Future Inputs**: These inputs $\mathbf{x}_{i,t}$ can be measured beforehand (e.g., sine and cosine of the day of a week at a given date) and are known at the time of prediction. We add known future inputs across the entire range for TFT ($\mathbf{x}_{i,t-k:t} = \{\mathbf{x}_{i,t-k}, \cdots, \mathbf{x}_{i,t}, \cdots, \mathbf{x}_{i,t+\tau}\}$). Other models which don't exclusively support known future inputs incorporated this feature up to the forecast start time t.

The time series model outputs daily COVID-19 case forecasts $\hat{y}_i(t,\tau)$ for τ_{max} time steps, where $\tau \in 1, ..., \tau_{max}$ is the daily prediction interval in the future (up to 15 days).

IV. Temporal Fusion Transformer

To understand the rationale behind choosing TFT [14] for this study, we give a theoretical background of TFT and its self-attention weights, which we later extract to interpret the spatiotemporal patterns of COVID-19 infection.

A. Model Architecture

Fig. 4 shows a brief overview of the TFT model architecture for three types of input covariates and the target output. We highlighted four key components of the model as follows:

1) **Embedding and input transformation** are performed on static metadata, time-varying past inputs, and time-varying known future inputs. The model inputs are passed through a Variable Selection Network (VSN) to select the most salient features and filter out noise.
2) **LSTM layer** enhances learning significant points in the surrounding (e.g. anomalies, cyclical patterns) by leveraging local context. Past inputs are fed into the encoder,

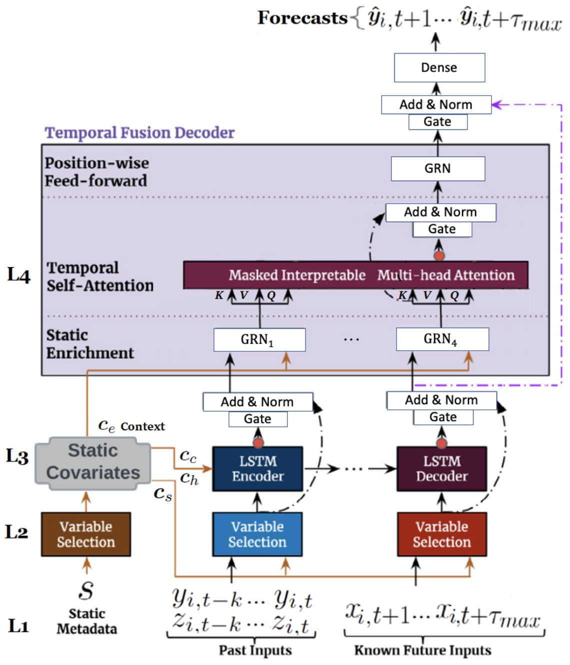


Fig. 4: TFT architecture [14]. TFT effectively builds feature representation from static covariates, observed inputs, and known future events. The transformer adopts four key layers from the bottom: (L1) Embedding & Input Transformation, (L2) Variable Selection, (L3) LSTM, (L4) Self-Attention.

whereas known future inputs are fed into the decoder. The outputs go through a static enrichment layer. For example, the static covariate features (e.g., Age distribution, Health Disparities) provide the context vectors (c_s, c_c, c_h, c_e) to conditions for temporal dynamics: (a) c_s being fed to the temporal VSN blocks, (b) (c_c, c_h) setting up the initial *cell state* and *hidden state* vectors of LSTM for local processing of temporal features, and (c) c_e enriching of temporal features at the later static enrichment phase.

3) **Static enrichment** layer uses Gated Residual Network (GRN) to enhance temporal features with static metadata, as static features often influence temporal dynamics. Gated Residual Network (GRN) works as its building block.
4) **Interpretable multi-head self-attention** takes static-enriched temporal features as inputs and learns long-range temporal dependencies. The self-attention can access all previous states and weigh them according to a learned measure of relevance.

B. Attention Weight in TFT

TFT uses the self-attention mechanism to learn long-term time-dependent relationships. This is modified from the multi-head attention in transformers [20] to enhance explainability. In this section we explain, how the attention works before using them to interpret different spatiotemporal patterns later in Section VII & VIII.

Given, values, keys, and queries as $\mathbf{V}$, $\mathbf{K}$, and $\mathbf{Q}$, attention can be defined as below where $A()$ is a normalization function:

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = A(\mathbf{Q}, \mathbf{K})\mathbf{V} \tag{3}$$

Multi-head attention [20] improves the learning capacity of this standard attention by employing different heads ($\mathbf{H}_h$) for different representations and then combining them:

$$\begin{aligned} MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= [\mathbf{H}_1, \cdots, \mathbf{H}_{m_H}]\mathbf{W}_H \\ \mathbf{H}_h &= Attention(\mathbf{Q}\mathbf{W}_Q^{(h)}, \mathbf{K}\mathbf{W}_K^{(h)}, \mathbf{V}\mathbf{W}_V^{(h)}) \end{aligned} \tag{4}$$

TFT [14] improved this multi-head attention by sharing values in each head and additively integrating as follows:

$$\begin{aligned} InterpretableMultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= \tilde{\mathbf{H}}\mathbf{W}_H \\ \tilde{\mathbf{H}} &= \tilde{A}(\mathbf{Q}, \mathbf{K})\mathbf{V}\mathbf{W}_V \\ &= \frac{1}{H}\sum_{h=1}^{m_H} Attention(\mathbf{Q}\mathbf{W}_Q^{(h)}, \mathbf{K}\mathbf{W}_K^{(h)}, \mathbf{V}\mathbf{W}_V) \end{aligned} \tag{5}$$

After the static-enrichment layer, the enriched temporal features are grouped into a matrix $\Theta(t) = [\theta(t, -k), \ldots, \theta(t, \tau)]^T$, where k is the encoder length and τ is the forecast horizon. At each forecasting time, t, the self-attention layer $\tilde{A}$ calculates a matrix of attention weights. The multi-head attention at each forecast horizon τ is then defined as an attention-weighted sum of lower-level features at each position $n \in (-k, \tau_{max})$, given by the following equation:

$$\beta(t, \tau) = \sum_{n=-k}^{\tau_{max}} \alpha(t, n, \tau)\tilde{\theta}(t, n) \tag{6}$$

where $\alpha(t, n, \tau)$ is the (τ, n)-th element of $\tilde{A}$ and $\tilde{\theta}(t, n)$ is a row of $\tilde{\Theta}(t) = \Theta(t)\mathbf{W}_v$. For each forecast horizon τ, the importance of a previous time point $n < \tau$ (e.g. prior day) can be calculated by analyzing the $\alpha(t, n, \tau)$ values across time steps (e.g. days) and entities (e.g. counties).

In our study, we utilized the PyTorch implementation of TFT [21]. The interpretable multi-head attention weight is a $(N_s, \tau, \boldsymbol{H}, k + \tau)$ matrix, where N_s is the total number of sequences in the dataset, τ is the forecasting horizon (15 days), $\boldsymbol{H}$ is the number of attention heads, and k is the number of prior days (13 days). N_s can be computed by $N_d \times (k + \tau - 1)$, where N_d is the number of time steps per county. The upper right half of the attention matrix is masked since $\alpha(t, i, j) = 0, \forall i > j$. The mean is then taken over by the attention heads to obtain the attention weight for each day at each county level. An illustration of this process is shown in Fig. 5.

V. Experimental Setup

In this section, we describe the design of the experiments with the runtime environments we used, our train/test/validation splits, and how we performed the hyper-parameter tuning.

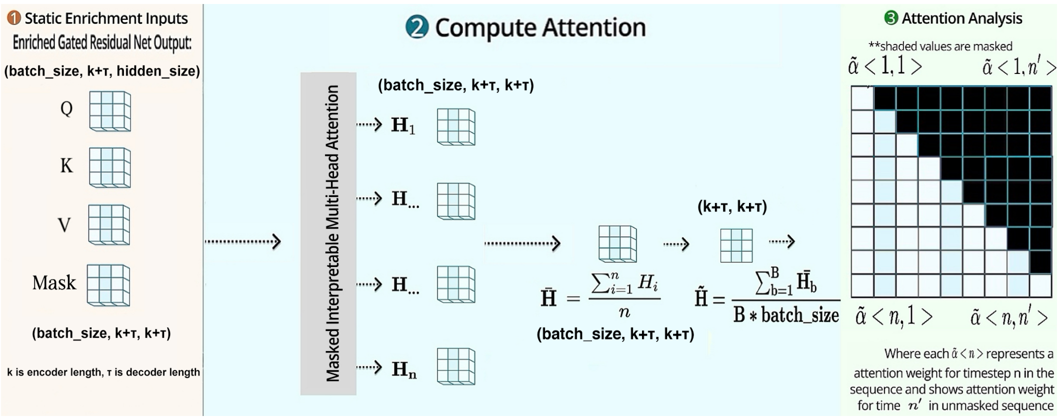

❶ Static Enrichment Inputs
Enriched Gated Residual Net Output:
(batch_size, k+τ, hidden_size)
Q
K
V
Mask
(batch_size, k+τ, k+τ)
k is encoder length, τ is decoder length
❷ Compute Attention
Masked Interpretable Multi-Head Attention
(batch_size, k+τ, k+τ)
$\mathbf{H}_1$
$\mathbf{H}_{\dots}$
$\mathbf{H}_{\dots}$
$\mathbf{H}_n$
$\bar{\mathrm{H}} = \frac{\sum_{i=1}^{n} H_i}{n}$
(batch_size, k+τ, k+τ)
(k+τ, k+τ)
$\tilde{\mathrm{H}} = \frac{\sum_{b=1}^{B} \bar{\mathrm{H}}_b}{B * \text{batch\_size}}$
❸ Attention Analysis
**shaded values are masked
$\tilde{\alpha} < 1, 1 >$
$\tilde{\alpha} < 1, n' >$
$\tilde{\alpha} < n, 1 >$
$\tilde{\alpha} < n, n' >$
Where each $\tilde{\alpha} < n >$ represents a attention weight for timestep n in the sequence and shows attention weight for time n' in unmasked sequence

Fig. 5: Flow of aggregation and selection for TFT attention weights.

A. Data Splits

Table III illustrates the partitioning of the dataset into training, validation, and test sets using different dataset splits. Unless explicitly mentioned, all experiments in this study used the **Primary split**. In Section VII-A, we used other splits to evaluate our model's ability to learn and perform well on different trends of the COVID-19 infection. The validation set in each split comprises the next 15 days after the training period, and the test set comprises the following 15 days after the validation period.

TABLE III: Dataset splits and dates.

Split	Start	End	Dataset
Primary	02-29-2020 11-30-2021 12-15-2022	11-29-2021 12-14-2021 12-29-2022	Train Validation Test
Rising 3rd Wave	02-29-2020 01-01-2022 01-16-2022	12-31-2021 01-15-2022 01-30-2022	Train Validation Test
Falling 3rd Wave	02-29-2020 02-01-2022 02-16-2022	01-31-2021 02-15-2022 03-02-2022	Train Validation Test
Post 3rd Wave	02-29-2020 03-01-2022 03-16-2022	02-28-2021 03-15-2022 03-30-2022	Train Validation Test

B. Hyperparameter Tuning

We evaluated the prediction performance of our TFT model and compared it with four other deep learning-based models, namely LSTM, Bidirectional-LSTM (Bi-LSTM), NBEATS, and NHiTS. The LSTM and Bi-LSTM models were implemented using TensorFlow, while for NBEATS and NHiTS we used the Darts framework [22]. The PyTorch implementation of TFT [21] was used in our experiments. We tuned the models' hyperparameters using Optuna [23], with 25 trial runs for each model and selected the best configuration based on the validation loss. Table IV summarizes the models' parameters and the tuning results. All models are optimized using *Adam* optimizer and *MSE* loss. We used the mean squared error (MSE) as the loss function, consistent with prior works on COVID-19 forecasting [12] [24].

C. Computational Resources

We conducted our model experiments on HPC clusters including the GPU nodes in Table V. The minimum memory requirement is 32GB of RAM. We maintained both docker and singularity containers for the full reproducibility of our work.

TABLE V: Runtime environment and hardware specification.

Driver	CUDA	Processor	NVIDIA GPU
470.82.01	11.4	Intel Xeon	A100-SXM4-40GB
			Tesla P100-PCIE
			Tesla V100-SXM2

VI. Performance Benchmarks

A. Evaluation Metrics

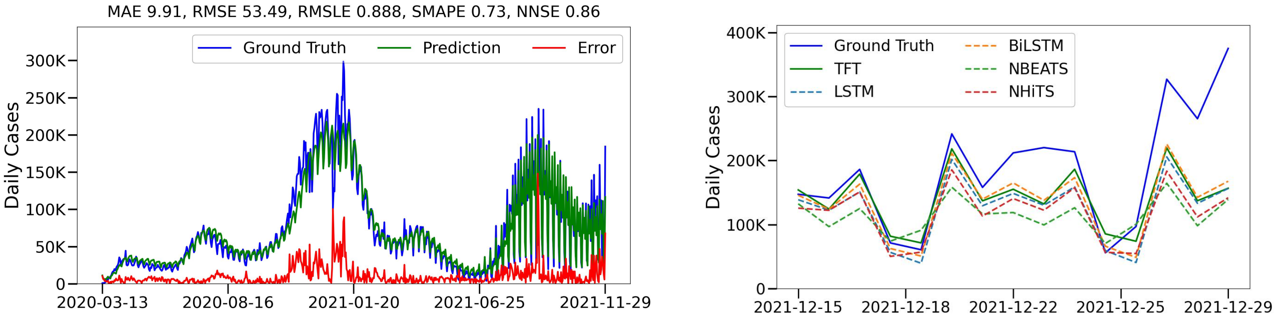
Our forecasting models are evaluated using a range of metrics commonly used in time-series forecasting and COVID-19 infections prediction, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Root Mean Square Logarithmic Error (RMSLE), Symmetric Mean Absolute Percentage Error (SMAPE), and Normalized Nash-Sutcliffe Efficiency (NNSE) [25] with definitions given in the Appendix A. These metrics have been widely adopted to evaluate the performance of time-series forecasting models [26] [27] [28] [29]. We use the scikit-learn [30] implementation of these metrics.

Normalized Nash-Sutcliffe Efficiency (NNSE) is used as an overall measure of model performance. It is defined as $1/(2 - NSE)$, where NSE is equivalent to the coefficient of determination ($\mathbf{R}^2$). Unlike other metrics, NNSE is robust to error variance and has a range of $[0, 1]$. A model with NNSE = 1 represents a perfect fit, while a model with NNSE = 0.5 has the same error variance as the observed time series. When the error variance is larger, NNSE will be in the range of $(0, 0.5)$.

MAE is preferred when we want to penalize the model equally irrespective of the error magnitude, whereas RMSE is used to penalize more for larger outliers. Since SMAPE measures the proportional error, it is more appropriate for comparing the performance of models across counties with different population sizes. RMSLE measures the difference

TABLE IV: Model training and network parameters (optimal values are in bold).

Model	Parameters	Value	Parameters	Value	Parameters	Value
TFT	learning rate hidden layer	**1e-3**, 1e-4 **16**, 32, 64	batch size dropout rate	64 0.2	attention head gradient clip	1, **4** 0.01, **1.00**
LSTM	learning rate hidden size	(1e-5, 1e-3), **1.26e-5** 32, **64**, 128	batch size dropout	32, 64, **128** **0**, 0.1, 0.2, 0.3	layers	**2**, 3, 4
Bi-LSTM	learning rate hidden size	(1e-5, 1e-3), **2.46e-5** 32, 64, **128**	batch size dropout	32, 64, **128** 0, **0.1**, 0.2, 0.3	layers	**2**, 3, 4
NBEATS	learning rate layers	(1e-5, 1e-3), **1e-5** **2**, 3, 4	batch size dropout	32, **64**, 128 0, 0.1, **0.2**, 0.3	layer width num stacks	256 30
NHiTS	learning rate layers	(1e-5, 1e-3), **4.26e-5** **2**, 3, 4	batch size layer width	**32**, 64, 128 512	dropout num stacks	0.1 3



(a) Train results of TFT

(b) Test results: TFT predictions (green line) outperform related work with high accuracy

Fig. 6: Performance comparison of TFT with other four time-series models (on the Primary split).

between the logarithmic predictions and the logarithmic true values, and is useful when the error distribution is skewed.

B. Comparison with Related Works

The test results for the hyperparameter-tuned models on the Primary split using are presented in Table VI. A lower score is better for MAE, RMSE, and SMAPE. Higher is better for NNSE.

TABLE VI: Prediction performance comparison on the test set. The best results are in bold.

Model	MAE	RMSE	RMSLE	SMAPE	NNSE
TFT	**35.68**	**221.3**	**1.347**	**0.842**	**0.679**
LSTM	40.27	267.1	1.434	1.054	0.616
Bi-LSTM	40.36	261.8	1.465	1.022	0.626
NHiTS	36.79	247.5	1.366	1.066	0.628
NBEATS	41.22	244.8	1.649	1.134	0.633

Table VI demonstrates that the TFT model outperforms the other models across all evaluation metrics. Fig. 6 presents the aggregated prediction plots. During training, there were two significant error spikes for the TFT predictions on Christmas day (Dec 25, 2020) and US Labor Day (Sep 06, 2021), which corresponded with substantial drops in reported cases. While our outlier removal step partially mitigated these two significant drops, they still resulted in large errors in the predictions.

VII. Interpreting Temporal Patterns

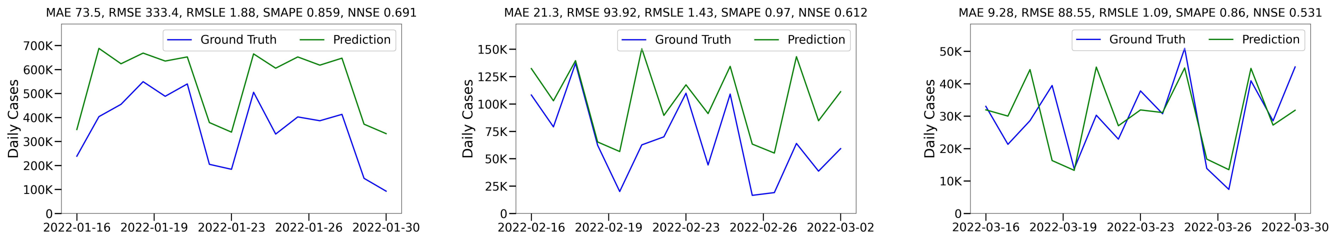
Time series data typically exhibit various temporal patterns, such as trend, seasonal, and cyclic patterns. 1) *Trend* is a long-term increase or decrease in the data, which can be linear or non-linear. 2) *Seasonal* patterns are affected by a fixed known frequency, such as year, month, or week. 3) *Cyclic* patterns, on the other hand, exhibit rise and fall, but not with a fixed frequency. In this section, we investigate how well our TFT model can learn and interpret these patterns by conducting experiments on data with these patterns.

A. Infection Trends

Since the start of the COVID-19 pandemic, various factors such as lockdown measures and the emergence of different variants have resulted in multiple waves of infection with distinct temporal patterns. To evaluate our model's ability to learn and generalize to different trends, we conducted experiments on the third wave of COVID-19 [19]. Specifically, we tested the model's ability to predict the surge of infection, after the peak infection is reached and the post-peak period. To this end, we extended our dataset to more recent dates as reported in Table III and performed experiments on three additional splits (rising, falling, and post 3rd wave). The results, shown in Fig. 7, indicate that our TFT model performs consistently well across different trends of COVID-19 infection waves.

B. Seasonal Patterns

Periodic patterns in the incidence of COVID-19 cases can reflect the contribution of various societal and epidemiological factors that influence the spread, testing, and reporting of the disease. Auto-correlation analysis is a common method used to identify periodicity in time-series datasets. Auto-correlation measures the degree of similarity between a time series and

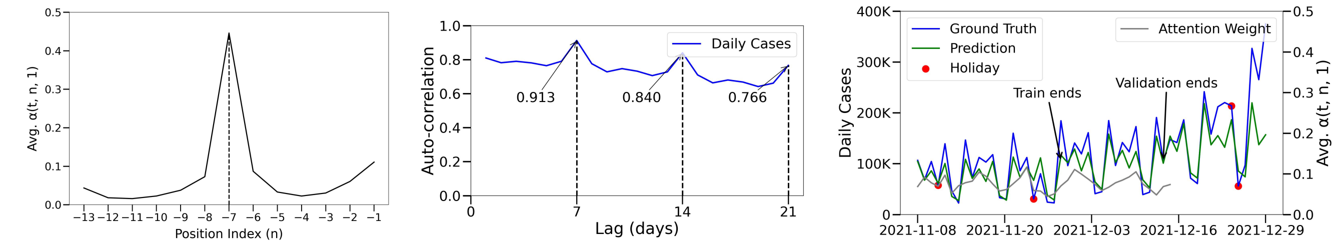


(a) Rising 3rd Wave

(b) Falling 3rd Wave

(c) Post 3rd Wave

Fig. 7: Trend: TFT test performance on all US counties for additional data splits for different infection trends.



(a) Attention weights aggregated by past time index showing high importance in the same day the previous week.

(b) Weekly seasonality due to reporting.

(c) Cyclic holiday patterns (Thanksgiving, Christmas).

Fig. 8: Persistent temporal patterns in infection forecasting and model attention weights.

a lagged version of itself at different time lags. Given a time series $Y_1, Y_2, \cdots, Y_N$ at times $t_1, t_2, \cdots, t_N$, the *lag k* auto-correlation function is defined as:

$$r_k = \frac{\sum_{i=1}^{N-k}(Y_i - \bar{Y})(Y_{i+k} - \bar{Y})}{\sum_{i=1}^{N}(Y_i - \bar{Y})^2}, \quad (7)$$

where $\bar{Y}$ is the mean of the time series. The auto-correlation function returns a value between -1 and 1, with values close to 1 indicating a strong positive correlation between the time series and its lagged version, values close to -1 indicating a strong negative correlation, and values close to 0 indicating no correlation.

We aggregated the daily COVID-19 case incidence data by dates and then plotted the auto-correlation for lags $k \in [1, 21]$ in Fig. 8b. Our analysis shows a clear weekly periodicity, where the correlation peaks at lag day $k = 7$. This finding is consistent with previous studies that have observed weekly oscillations in infection and death rates during the first wave of the COVID-19 pandemic [31] [32]. The weekly fluctuations can be attributed to the way the health sector handled COVID-19 test and case reporting [31], rather than any underlying societal or biological factors. By learning and incorporating such patterns, an epidemic model can make more accurate forecasts of future incidence.

The TFT model's ability to predict weekly patterns and trends for both the train and test periods is demonstrated in Fig. 6, as well as in a zoomed-in version with holidays shown in Fig. 8c. This learning can be further analyzed by looking at the attention weights ($\alpha(t, n, \tau)$) of the TFT model. For a given time t and encoder position index n, the attention weight assigned to a forecasting horizon τ can be determined. By plotting the average attention weights for $\tau = 1$ only, as shown in Fig. 8a, we observe a clear weekly periodicity that peaks at position index -7. This indicates that the input data from the same day in the previous week received the most attention from the model when forecasting for the next day ($n = 0$).

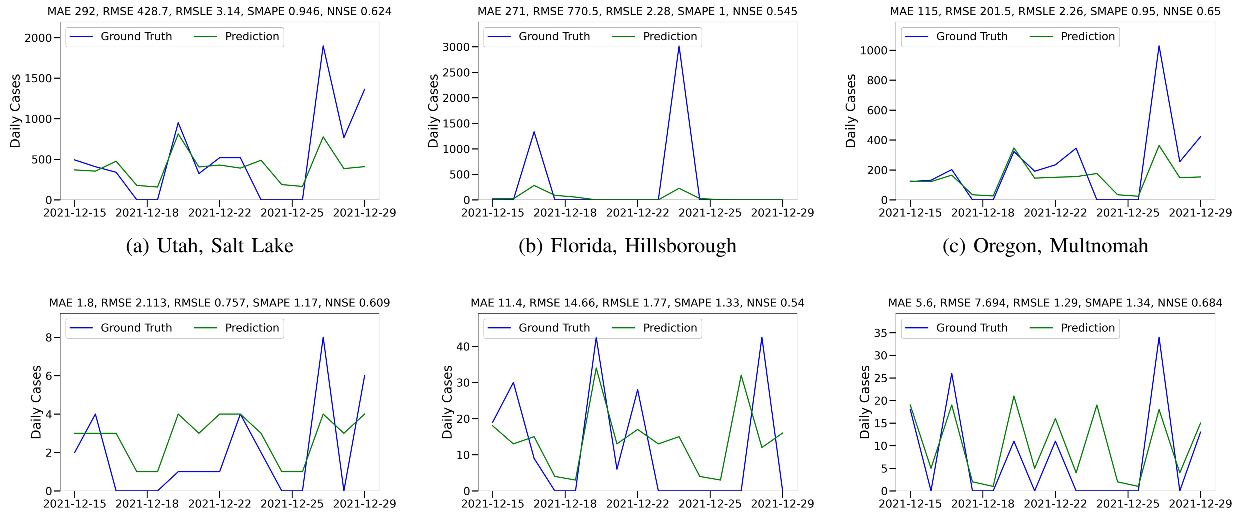
C. Cyclic Patterns

We also observed a drop in reported cases on holidays, which can be attributed to the same reasons as the weekly patterns. During holidays, hospitals and COVID-19 test centers often have reduced staffing and operating hours, leading to fewer tests and reported cases [31]. However, since holiday occurrences are relatively infrequent (e.g., yearly), it is more challenging for the model to learn and incorporate this information with only 2.5 years of data.

To evaluate the model's performance in predicting holiday effects, we plotted the ground truth and predicted cases from some weeks before the end of the training period, till the end of the test period in Fig. 8c, annotated with Veterans Day (Nov 11, 2021), Thanksgiving (Nov 25, 2021), and Christmas (Dec 24-25, 2021). Fig. 8c shows that the TFT model can correctly predict the dip in reported cases during the Christmas holidays in the test period. The attention weights also show drops on those days, implying less focus from the model on holidays.

VIII. Interpreting Spatial Patterns

Understanding the spatial dynamics of the forecasting model is crucial in comprehending the spread of COVID-19 across different regions with diverse socio-economic backgrounds.



(a) Utah, Salt Lake

(b) Florida, Hillsborough

(c) Oregon, Multnomah

(d) Virginia, Brunswick

(e) North Carolina, Yancey

(f) Michigan, Ogemaw

Fig. 9: Spatial Trend: Cases prediction performance for six randomly selected US counties. The top row contains counties selected from the top 100 US counties by population. The bottom row of counties is selected from the rest.

Spatial analysis can provide insights into the local factors that influenced the transmission of the disease, leading to variations in infection rates [33]. To shed light on this important aspect, we posed the following research questions:

1) Can our model accurately predict COVID-19 infection trends in diverse geographical locations?
2) How do the attention weights of the model vary across different geographic regions?

By answering these research questions, we can gain a better understanding of the model's behavior in different demographics. Additionally, these insights can help further research to explore ways to optimize the model's performance for specific regions and demographic groups to improve the accuracy of COVID-19 infection forecasts.

A. Infection Trends at Different Locations

The spread and transmission of COVID-19 can vary significantly across counties due to local factors such as population density, housing conditions, available medical facilities, and the effectiveness of lockdown measures. In particular, large cities tend to have more heterogeneous populations and greater potential for the spread of infections. Therefore, it is crucial that the forecasting model can accurately predict changing infection trends, despite the geospatial differences between these locations.

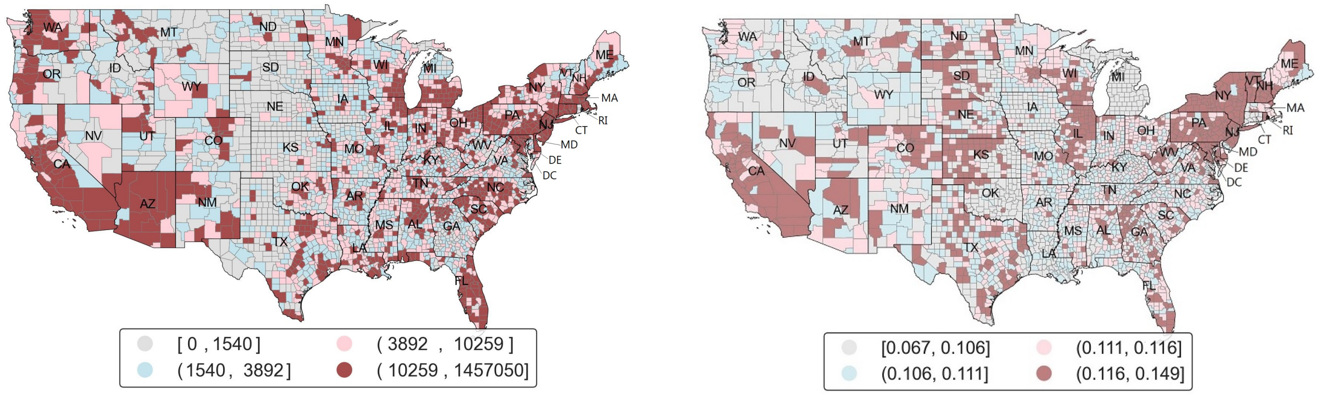
To demonstrate the effectiveness of our Temporal Fusion Transformer (TFT) model, we selected six US counties at random and plotted their predicted infection trends in Fig. 9. The upper three counties were randomly chosen from the top 100 US counties by population, while the lower three counties were randomly selected from the rest. The figure shows that our model was able to accurately predict the infection trends in each of these counties, despite their diverse characteristics and infection rates.

In Fig. 9, we can observe that the population differences across the counties are reflected in the reported COVID-19 cases. The smaller counties often have a ground truth of either zero or very few infections, while the larger counties have a higher number of infections as well as larger fluctuations. Despite these scale differences, our model can accurately predict the trends in all the counties. This indicates that our model can generalize well across different demographic groups and geographical locations, which is crucial for making accurate forecasts and taking informed decisions.

B. Attention Focus across Counties

To further understand the impact of the TFT model's attention mechanism on forecasting COVID-19 infection cases, we examine the variation in attention weights across different counties. As discussed earlier, counties differ in their population, socio-economic factors, and geographical location, which can impact the spread of the epidemic. Therefore, attention weights may vary depending on these factors. To quantify this, we calculate the average attention weights for each county on the previous day (position index -1 in Fig. 8a) for the next day ($\tau = 1$) prediction ($\alpha(t, -1, 1)$). This allows us to observe how attention weights differ across counties from the previous day to the next day's prediction. Analyzing these results provides a more detailed interpretation of the relationship between the attention weight of a particular day and the ground truth. Although we present this analysis for one case, it can be extended to other position indexes (n) and forecast horizons (τ) as well.

Fig. 10 provides the density maps of the cumulative COVID-19 cases and their corresponding average attention weights for each county using the Primary split outlined in Table III. We utilized the US census bureau's [34] shape geometry and GeoPandas to plot the map, excluding geo-



(a) Cumulative COVID-19 cases across US counties

(b) Avg. attention weights across US counties from TFT

Fig. 10: Spatial distribution of COVID-19 cases in US counties and corresponding attention weights from TFT.

graphically distant states such as Alaska and Puerto Rico for convenience. To enhance visual clarity, we divided the values into four quantiles to highlight the distinct clusters. The map is useful in identifying counties with similar COVID-19 trends and attention weights, providing insight into the model's performance across various regions. This can aid in devising tailored interventions and mitigation strategies in response to the pandemic.

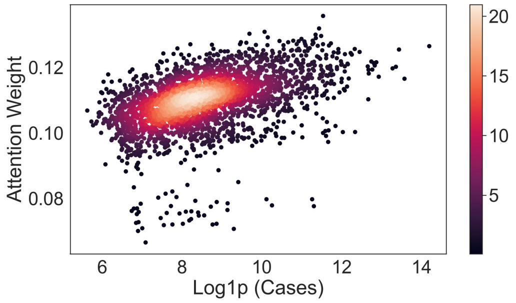


Fig. 11: Correlation density between COVID-19 cases predictions and TFT model's attention weights. Each point represents a county summed over the full time duration.

As shown in Fig. 10, neighboring counties often have similar densities in COVID-19 cases and attention weights. For instance, the states of California (CA) and Pennsylvania (PA) have a larger number of counties with higher case rates and attention weights, while Idaho (ID) and Wyoming (WY) exhibit lower infection counts and attention weights. Such non-homogeneous distribution would be overlooked in state-level analysis and underscores the importance of local heterogeneity in this interpretation. Fig. 11 presents the correlation scatter plot between log1p of cases and attention weight, revealing a mostly positive linear relationship.

IX. Interpreting Feature Importance

To analyze the variable importance, we computed the sum of the weights assigned to each variable from the Variable Selection Network (VSN) across the training set. We then normalized the weights for each input feature to percentage and presented them in Table VII. For the static covariates, we observed that the age distribution has the highest weight. Among the observed inputs, the past target values (cases) are found to be the most important, which is expected since they are directly related to the current infection rate.

TABLE VII: Feature importance from variable selection weights (the highest values are in bold).

Feature	Static	Observed	Known
Cases		**38.26%**	
Age Distribution	**54.45%**		
Health Disparities	45.55%		
Vaccination		11.28%	
Disease Spread		16.32%	
Transmissible Cases		3.26%	
Social Distancing		3.35%	
SinWeekly		16.87%	**72.85%**
CosWeekly		10.66%	27.15%

X. Related Work

In this section, we provide an overview of the literature related to predicting COVID-19 infections using various approaches, including statistical, machine learning, and deep learning methods. Additionally, we summarize related work on interpreting such models.

A. Statistical and Machine Learning Models

Many efforts have been made to COVID-19 forecasting using statistical learning, epidemiological, and machine learning models [35]. Different statistical models such as Susceptible Infectious Recovery (SIR), and Susceptible Exposed Infectious Recover (SEIR), have been used to simulate and forecast the COVID-19 spread [1] [36]. These models provide useful insights and are often easier to interpret. However, their performance is limited by the number of complex influencing factors and relationships they can capture, which is where the machine learning-based models excel [4] [3] [37]. Models like Auto-Regressive Integrated Moving Average (ARIMA) [5]

[4] [3], Seasonal Auto-Regressive Integrated Moving Average (SARIMA) [3] and XGBoost [38] have been used to forecast COVID-19 cases and deaths. These models can capture nonlinear relationships between variables and have shown improved performance compared to statistical models in some cases. However, they may have limitations in dealing with high-dimensional and temporal data.

B. Deep Learning Models

Deep learning has demonstrated remarkable performance in time series forecasting [28] [39] and has been widely adopted for predicting COVID-19 infections [35]. LSTM and Bi-LSTM-based models are often superior to other statistical and machine learning approaches in time-series forecasting [7] [24] [40] [41]due to the ability of RNN-based models to learn from sequential data. Moreover, [26] found that Variational Auto Encoder (VAE) outperforms RNN-based deep learning models in predicting daily confirmed and recovered COVID-19 cases. New research using AI-based predictions also discusses several issues, including the discovery of new virus variants [42], limited data quality and quantity used for model training, and the possibility that ML-based models may not incorporate socioeconomic, cultural, and demographic factors while learning the data [43]. However, interpreting these models can be challenging due to their black-box nature, which has led to efforts to develop methods for explaining their predictions

C. Interpreting the Forecasting

As deep learning models are becoming more prevalent in COVID-19 infection prediction, there is a growing need to interpret the models and understand how they arrive at their decisions [28]. For instance, DeepCOVID [44] utilized RNN with auto-regressive inputs to predict COVID-19 cases and provide insight into the input features' contribution to the prediction performance. Meanwhile, DeepCOVIDNet [45] analyzed the features and their interactions to predict the range of infected cases increase at the US county level, albeit addressing this as a classification task instead of forecasting the infection. Self-Adaptive Forecasting [12] adapts models to non-stationary time-series data and provides explanations with TFT. This approach was used to interpret state-level COVID-19 death forecasts [12]. Additionally, [46] proposed a rule-based local explainer that interprets predicted electricity consumption based on aggregated features.

XI. Conclusions and Future Work

Interpretation of deep learning models has gained significant attention in recent years for their applications in various domains, including social impact [47] [48]. In this paper, we utilize the Temporal Fusion Transformer (TFT), an attention-based deep learning time-series model, to achieve state-of-the-art performance in forecasting US county-level infections while enabling new forms of interpretability [49] through analyzing complex spatiotemporal patterns. The proposed model (1) outperforms other popular deep learning models such as LSTM, Bi-LSTM, NBEATS, and NHiTS in all evaluation metrics for multivariate multi-horizon forecasting, (2) exhibits robust performance in predicting non-stationary trends of the infections at different waves of the COVID-19 pandemic, (3) interprets temporal patterns, such as weekly and holiday seasonality in reported cases, through multi-head attention weights, and (4) reveals spatial patterns using attention weights that are correlated to the infection spread. The model performs consistently across different counties, despite the large variation in infection rates due to diverse community-level characteristics (e.g. population, health status, and socioeconomic factors).

Our results show that the model learns multiple long-range dependencies in highly dynamic forecasting problems. By interpreting such spatiotemporal patterns at the individual county level, we provide quantitative explanations of the impact of the infection trends in a meaningful way. Future work can focus on adaptively optimizing the model for dynamic data while remaining robust against extreme events and analyzing the sensitivity of input features to understand the impact on communities. Finally, our work enables disentangling the complex learning of TFT and similar analysis will be needed in the future for making deep learning interpretable in many other health and financial applications.

References

[1] X. Zhou, X. Ma, N. Hong, L. Su, Y. Ma, J. He, H. Jiang, C. Liu, G. Shan, W. Zhu *et al.*, "Forecasting the worldwide spread of covid-19 based on logistic model and seir model," *MedRxiv*, pp. 2020–03, 2020.

[2] F. Amaral, W. Casaca, C. M. Oishi, and J. A. Cuminato, "Towards providing effective data-driven responses to predict the covid-19 in são paulo and brazil," *Sensors*, vol. 21, no. 2, p. 540, 2021.

[3] P. Kumar, H. Kalita, S. Patairiya, Y. D. Sharma, C. Nanda, M. Rani, J. Rahmani, and A. S. Bhagavathula, "Forecasting the dynamics of covid-19 pandemic in top 15 countries in april 2020: Arima model with machine learning approach," *MedRxiv*, pp. 2020–03, 2020.

[4] K. ArunKumar, D. V. Kalaga, C. M. S. Kumar, G. Chilkoor, M. Kawaji, and T. M. Brenza, "Forecasting the dynamics of cumulative covid-19 cases (confirmed, recovered and deaths) for top-16 countries using statistical machine learning models: Auto-regressive integrated moving average (arima) and seasonal auto-regressive integrated moving average (sarima)," *Applied soft computing*, vol. 103, p. 107161, 2021.

[5] H. Alabdulrazzaq, M. N. Alenezi, Y. Rawajfih, B. A. Alghannam, A. A. Al-Hassan, and F. S. Al-Anzi, "On the accuracy of arima based prediction of covid-19 spread," *Results in Physics*, vol. 27, p. 104509, 2021.

[6] H. Abbasimehr and R. Paki, "Prediction of covid-19 confirmed cases combining deep learning methods and bayesian optimization," *Chaos, Solitons & Fractals*, vol. 142, p. 110511, 2021.

[7] F. Shahid, A. Zameer, and M. Muneeb, "Predictions for covid-19 with deep learning models of lstm, gru and bi-lstm," *Chaos, Solitons & Fractals*, vol. 140, p. 110212, 2020.

[8] C.-J. Huang, Y.-H. Chen, Y. Ma, and P.-H. Kuo, "Multiple-input deep convolutional neural network model for covid-19 forecasting in china," *MedRxiv*, pp. 2020–03, 2020.

[9] R. Wang, D. Maddix, C. Faloutsos, Y. Wang, and R. Yu, "Bridging physics-based and data-driven modeling for learning dynamical systems," in *Learning for Dynamics and Control*. PMLR, 2021, pp. 385–398.

[10] S. Er, S. Yang, and T. Zhao, "County aggregation mixup augmentation (courage) covid-19 prediction," *Scientific Reports*, vol. 11, no. 1, p. 14262, 2021.

[11] S. Chang, E. Pierson, P. W. Koh, J. Gerardin, B. Redbird, D. Grusky, and J. Leskovec, "Mobility network models of covid-19 explain inequities and inform reopening," *Nature*, vol. 589, no. 7840, pp. 82–87, 2021.

[12] S. O. Arik, N. C. Yoder, and T. Pfister, “Self-adaptive forecasting for improved deep learning on non-stationary time-series,” *arXiv preprint arXiv:2202.02403*, 2022.

[13] B. Peng, J. Li, S. Akkas, T. Araki, O. Yoshiyuki, and J. Qiu, “Rank position forecasting in car racing,” in *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. Los Alamitos, CA, USA: IEEE Computer Society, may 2021, pp. 724–733.

[14] B. Lim, S. Ö. Arık, N. Loeff, and T. Pfister, “Temporal fusion transformers for interpretable multi-horizon time series forecasting,” *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.

[15] USA Facts, “(us covid-19 cases and deaths).” [Online]. Available: https://usafacts.org/visualizations/coronavirus-covid-19-spread-map/

[16] Unacast, “Social Distancing Scoreboard.” [Online]. Available: https://www.unacast.com/post/unacast-updates-social-distancing-scoreboard

[17] Centers for Disease Control and Prevention, “COVID-19 Vaccinations in the United States Counties.” [Online]. Available: https://data.cdc.gov/Vaccinations/COVID-19-Vaccinations-in-the-United-States-County/8xkx-amqh

[18] C. for Disease Control and Prevention, “Social vulnerability index,” 2020. [Online]. Available: https://www.atsdr.cdc.gov/placeandhealth/svi/data_documentation_download.html

[19] R. M. El-Shabasy, M. A. Nayel, M. M. Taher, R. Abdelmonem, K. R. Shoueir, and E. R. Kenawy, “Three waves changes, new variant strains, and vaccination effect against covid-19 pandemic,” *International Journal of Biological Macromolecules*, vol. 204, pp. 161–168, 2022.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[21] PyTorch, “Temporal Fusion Transformer.” [Online]. Available: https://pytorch-forecasting.readthedocs.io/en/stable/api/pytorch_forecasting.models.temporal_fusion_transformer.TemporalFusionTransformer.html

[22] J. Herzen, F. Lässig, S. G. Piazzetta, T. Neuer, L. Tafti, G. Raille, T. Van Pottelbergh, M. Pasieka, A. Skrodzki, N. Huguenin *et al.*, “Darts: User-friendly modern machine learning for time series,” *Journal of Machine Learning Research*, vol. 23, no. 124, pp. 1–6, 2022.

[23] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[24] M. D. Hssayeni, A. Chala, R. Dev, L. Xu, J. Shaw, B. Furht, and B. Ghoraani, “The forecast of covid-19 spread risk at the county level,” *Journal of big data*, vol. 8, no. 1, pp. 1–16, 2021.

[25] J. Nossent and W. Bauwens, “Application of a normalized nash-sutcliffe efficiency to improve the accuracy of the sobol’sensitivity analysis of a hydrological model,” in *EGU General Assembly Conference Abstracts*, 2012, p. 237.

[26] A. Zeroual, F. Harrou, A. Dairi, and Y. Sun, “Deep learning methods for forecasting covid-19 time-series data: A comparative study,” *Chaos, Solitons & Fractals*, vol. 140, p. 110121, 2020.

[27] G. R. Shinde, A. B. Kalamkar, P. N. Mahalle, N. Dey, J. Chaki, and A. E. Hassanien, “Forecasting models for coronavirus disease (covid-19): a survey of the state-of-the-art,” *SN Computer Science*, vol. 1, pp. 1–15, 2020.

[28] B. Lim and S. Zohren, “Time-series forecasting with deep learning: a survey,” *Philosophical Transactions of the Royal Society A*, vol. 379, no. 2194, p. 20200209, 2021.

[29] J. Nayak, B. Naik, P. Dinesh, K. Vakula, B. K. Rao, W. Ding, and D. Pelusi, “Intelligent system for covid-19 prognosis: A state-of-the-art survey,” *Applied Intelligence*, vol. 51, pp. 2908–2938, 2021.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[31] A. Bergman, Y. Sella, P. Agre, and A. Casadevall, “Oscillations in us covid-19 incidence and mortality data reflect diagnostic and reporting factors,” *Msystems*, vol. 5, no. 4, pp. e00 544–20, 2020.

[32] T. Pavlíček, P. Rehak, and P. Král, “Oscillatory dynamics in infectivity and death rates of covid-19,” *Msystems*, vol. 5, no. 4, pp. e00 700–20, 2020.

[33] S. Kim and M. C. Castro, “Spatiotemporal pattern of covid-19 and government response in south korea (as of may 31, 2020),” *International Journal of Infectious Diseases*, vol. 98, pp. 328–333, 2020.

[34] United States Census Bureau, “Cartographic Boundary Files.” [Online]. Available: https://www.census.gov/geographies/mapping-files/time-series/geo/cartographic-boundary.2021.html

[35] J. C. Clement, V. Ponnusamy, K. Sriharipriya, and R. Nandakumar, “A survey on mathematical, machine learning and deep learning models for covid-19 transmission and diagnosis,” *IEEE reviews in biomedical engineering*, vol. 15, pp. 325–340, 2021.

[36] L. López and X. Rodo, “A modified seir model to predict the covid-19 outbreak in spain and italy: simulating control scenarios and multi-scale epidemics,” *Results in Physics*, vol. 21, p. 103746, 2021.

[37] M. M. Rahman, M. M. Islam, M. M. H. Manik, M. R. Islam, and M. S. Al-Rakhami, “Machine learning approaches for tackling novel coronavirus (covid-19) pandemic,” *SN Computer Science*, vol. 2, pp. 1–10, 2021.

[38] J. Luo, Z. Zhang, Y. Fu, and F. Rao, “Time series prediction of covid-19 transmission in america using lstm and xgboost algorithms,” *Results in Physics*, vol. 27, p. 104462, 2021.

[39] P. Wang, X. Zheng, G. Ai, D. Liu, and B. Zhu, “Time series prediction for the epidemic trends of covid-19 using the improved lstm deep learning method: Case studies in russia, peru and iran,” *Chaos, Solitons & Fractals*, vol. 140, p. 110214, 2020.

[40] R. Chandra, A. Jain, and D. Singh Chauhan, “Deep learning via lstm models for covid-19 infection forecasting in india,” *PloS one*, vol. 17, no. 1, p. e0262708, 2022.

[41] G. C. Fox, G. von Laszewski, F. Wang, and S. Pyne, “Aicov: An integrative deep learning framework for covid-19 forecasting with population covariates,” *Journal of Data Science*, vol. 19, no. 2, pp. 293–313, 2021.

[42] E. A. Rashed and A. Hirata, “Infectivity upsurge by covid-19 viral variants in japan: Evidence from deep learning modeling,” *International journal of environmental research and public health*, vol. 18, no. 15, p. 7799, 2021.

[43] C. Sáez, N. Romero, J. A. Conejero, and J. M. García-Gómez, “Potential limitations in covid-19 machine learning due to data source variability: A case study in the ncov2019 dataset,” *Journal of the American Medical Informatics Association*, vol. 28, no. 2, pp. 360–364, 2021.

[44] A. Rodriguez, A. Tabassum, J. Cui, J. Xie, J. Ho, P. Agarwal, B. Adhikari, and B. A. Prakash, “Deepcovid: An operational deep learning-driven framework for explainable real-time covid-19 forecasting,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 17, 2021, pp. 15 393–15 400.

[45] A. Ramchandani, C. Fan, and A. Mostafavi, “Deepcovidnet: An interpretable deep learning model for predictive surveillance of covid-19 using heterogeneous features and their interactions,” *Ieee Access*, vol. 8, pp. 159 915–159 930, 2020.

[46] D. Rajapaksha and C. Bergmeir, “Limref: Local interpretable model agnostic rule-based explanations for forecasting, with an application to electricity smart meter data,” 2022.

[47] C. Molnar, G. Casalicchio, and B. Bischl, “Interpretable machine learning – a brief history, state-of-the-art and challenges,” in *ECML PKDD 2020 Workshops*. Springer International Publishing, 2020, pp. 417–431. [Online]. Available: https://doi.org/10.1007\%2F978-3-030-65965-3_28

[48] A. Feng, C. You, S. Wang, and L. Tassiulas, “Kergnns: Interpretable graph neural networks with graph kernels,” 2022.

[49] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

Appendix

In this section, we present the mathematical formulations of the evaluation metrics used in our work. At a given point $y, \hat{y}$, and $\bar{y}$ stand for the target ground truth, model prediction, and average ground truth respectively. When calculating these metrics at the US country level $n = (i, t, \tau), N = (I, T, \tau_{max})$, and $|N| = IT\tau_{max}$. Similarly, when evaluating at the US county level $(n = t, \tau), N = (T, \tau_{max}$, and $|N| = T\tau_{max})$. Here I is the set of counties, T is the length of that time series period, τ_{max} is the prediction horizon.

$$RMSE = \sqrt{\frac{1}{|N|}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2} \tag{8}$$

$$MAE = \frac{1}{|N|}\sum_{n=1}^{N}|y_n - \hat{y}_n| \tag{9}$$

$$RMSLE = \sqrt{\frac{1}{|N|}\sum_{n=1}^{N}(\log(1 + y_n) - \log(1 + \hat{y}_n))^2} \tag{10}$$

$$SMAPE = \frac{2}{|N|}\sum_{n=1}^{N}\frac{|y_n - \hat{y}_n|}{|y_n + \hat{y}_n|} \tag{11}$$

$$NNSE = \frac{1}{2 - NSE}, NSE = 1 - \frac{\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{\sum_{n=1}^{N}(y_n - \bar{y}_n)^2} \tag{12}$$