

Deep diffusion-based forecasting of COVID-19 by incorporating network-level mobility information

Padmaksha Roy,^{1*} Shailik Sarkar,^{2*} Subhodip Biswas,^{2*} Fanglan Chen,² Zhiqian Chen,³
Naren Ramakrishnan⁴ and Chang-Tien Lu⁴

¹Department of Electrical and Computer Engineering, Virginia Tech, Greater Washington DC area, USA
E-mail: padmaksha@vt.edu

²Department of Computer Science, Virginia Tech, Greater Washington DC area, USA
E-mail(s): shailik@vt.edu, subhodip@cs.vt.edu, fanglanc@vt.edu

³Department of Computer Science and Engineering, Mississippi State University, Starkville, Mississippi, USA
Email: zchen@cse.msstate.edu

⁴Sanghani Center for Artificial Intelligence and Data Analytics, Virginia Tech, Greater Washington DC area, USA
E-mail(s): naren@cs.vt.edu, ctlu@cs.vt.edu

*Author names arranged in alphabetical order of their first names.

Abstract—Modeling the spatiotemporal nature of the spread of infectious diseases can provide useful intuition in understanding the time-varying aspect of the disease spread and the underlying complex spatial dependency observed in people’s mobility patterns. Besides, the county level multiple related time series information can be leveraged to make a forecast on an individual time series. Adding to this challenge is the fact that real-time data often deviates from the unimodal Gaussian distribution assumption and may show some complex mixed patterns. Motivated by this, we develop a deep learning-based time-series model for probabilistic forecasting called Auto-regressive Mixed Density Dynamic Diffusion Network (ARM3Dnet), which considers both people’s mobility and disease spread as a diffusion process on a dynamic directed graph. The Gaussian Mixture Model layer is implemented to consider the multimodal nature of the real-time data while learning from multiple related time series. We show that our model, when trained with the best combination of dynamic covariate features and mixture components, can outperform both traditional statistical and deep learning models in forecasting the number of Covid-19 deaths and cases at the county level in the United States.

I. INTRODUCTION

The worldwide outbreak of the COVID-19 pandemic in 2020 has restricted people mobility both in micro and macro levels in an unprecedented manner. Although the origin of the disease still remains unclear but the unpredictable nature of the spread of the disease depending on various factors needs further investigation. A reliable disease forecasting of the outbreak can aid not only in preparing for adopting

containment measures but also in identifying areas with a high risk of spread for better resource allocation [1].

The scientific community has tackled this problem with different types of disease spread models. While in the beginning, most of the efforts were focused on traditional mathematical modeling and simulation [2], the application of data-driven methods increased as more data became widely available. While the majority of the works focused on forecasting the exact number of active cases [3], [4], some focused on predicting the number of newly infected cases, rate of infection, and a number of incoming cases from different geographical regions [5]. Our work proposes a data-driven method to forecast the exact number of positive cases and deaths at a multitude of different geographically smaller regions using spatiotemporal information of multiple correlated time series and people mobility information across those regions.

Most of the work on COVID-19 forecasting has so far exploited the temporal nature of the data and to the best of our knowledge have not considered both the spatial dependency among different region and the correlation among different time series across counties together to achieve better forecasting results. Adding to this is the uncertainty aspect of such forecasting in presence of various static and dynamic features.

In the early part of the pandemic, several factors like race and gender, [6], underlying factors—socioeconomic status [7], lock-down policy enforcement [8], social distancing measures [9], change in community mobility in different regions—that have proven to contribute more towards the nature of the spread [10]. Our work mainly focuses on the observation how people’s mobility at granular levels in between neighboring counties can have a direct correlation with the spread of the disease. The infectious and contagious nature of COVID-19 makes us hypothesize that understanding the population diffusion process can be significant in understanding the spread of disease.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ASONAM '21, November 8-11, 2021, Virtual Event, Netherlands

© 2021 Association for Computing Machinery.

ACM ISBN 978-1-4503-9128-3/21/11...\$15.00

<http://dx.doi.org/10.1145/3487351.3488334>

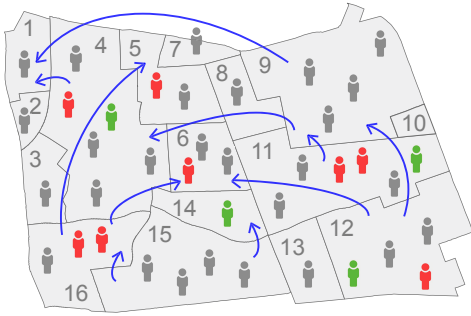


Fig. 1: A simulation depicting our assumption that infected people’s mobility (depicted in red in the diagram) within and across counties is correlated with the disease spread due to contagious nature of the disease.

In our work, we model the dynamics of people mobility as a diffusion process and use the diffusion convolution operation to capture the strong spatial dependency. We further propose the ARM3Dnet that integrates the diffusion convolution operation and an auto-regressive recurrent mixed density network and use sampling techniques to capture the true statistical nature of the data. We formulate our problem on Graph Neural Networks (GNNs) that can be utilized to get more accurate forecasts as they integrate graph structure with node attributes [11], [12]. The dynamic data collected by us, can be leveraged for implementing a graph structure with every node representing a county in the United States and edges denoting the people’s mobility between the counties.

The proposed model is a global model on multiple sequences of time series data, each corresponding to a county in the US [13]. A challenge with forecasting any infectious disease is that, with the ROIs (Region of Interest) becoming smaller, forecasting across different geographical regions becomes less consistent, making it poorer for generalization. Again, with an increased number of time series sequences, both the magnitudes and the distribution of the magnitude differ widely [14]. In fact, the majority of the existing works focus on forecasting of COVID-19 infection rate at a state level or country level [15], [16] and very few works have focused on proposing a global model that leverages the spatial as well as temporal dependency between geographical locations, and the joint training of multiple correlated time series data.

Based on the aforementioned observations, we develop our ARM3Dnet model in an attempt to address them. The contributions of this work can be summarized as follows:

- **We show that there is a direct correlation of inter and within county people’s mobility with the spread of the disease and associated deaths and infections at granular level.** We model the spatial dependency of people’s mobility as a diffusion process on a directed graph formed with a certain number of counties. The diffusion convolution intuitively captures both spatial and temporal dependencies when fed as time-dependent covariate to a sequence-sequence learning framework.

- **We try to prove that by considering different county-level time series which can be correlated and via sharing this information, one can actually improve the forecasting accuracy and the uncertainty.** In order to achieve it, we created an auto-regressive model that can jointly train multiple time series together and as a result at every instant the model learns from the time series information of neighboring counties.
- **We propose a model which can capture the multi-modal nature of the COVID real-time data that shows multiple peaks over time.** We introduced the Gaussian Mixture Model on top of the auto-regressive model to model the real time multi-modal nature of the COVID data as reported by the CDC and found improvements both in terms of accuracy and uncertainty.

II. BACKGROUND

Influenza-like illness (ILI) has been a subject of epidemiological research for several years. Flu forecasting, for example, has used everything from a mechanistic model that leverages epidemiological insights [17], [18] to predefined statistical models for disease modeling based on just historical data [19], [20]. Forecasting models using ARIMA and Prophet time series model [21] have also been proposed. But the mechanistic models require a lot of calibration, and they usually struggle to generalize well and accurately fit the data.

Cooper *et al.* [2] studied the effectiveness of the modeling approach on the pandemic due to the spreading of the novel COVID-19 disease. They developed a susceptible-infected-removed (SIR) model that provides a theoretical framework to investigate its spread within a community. On the other hand, Liu *et al.* [22] combined “COVID-19 case data with mobility data to estimate a modified susceptible-infected-recovered (SIR) model in the United States”. They observed that in contrast to a standard SIR model, the incidence of COVID-19 spread is concave if people have inter-related social networks. Recently, Goel *et al.* [23] proposed a mobility-based SIR model for epidemics, which especially takes into account the population distribution and connectivity of different geographic locations across the globe. Although these methods are based on traditional disease simulation models, they often fail to fit the real-world data successfully.

Bhouri *et al.* [24] used Google’s mobility data to use as features for their proposed DL-based model for doing county-wise forecasting of positive cases for over 200 counties. However, they do not address the correlation among different time series or the spatial dependency among different counties. Works like Adhikari *et al.* [25] addressed these issues and proposed a data-driven DL-based model that leveraged the seasonal similarities of historical data for flu forecasting tasks and introduced model interpretability. Although COVID-19 had a similar spreading pattern to other ILIs, the complete lack of historical seasonal data adds to the challenge of accurate forecasting. The data-driven methods suffer from exploiting seasonal similarities and providing enough interpretability and

fail to provide an uncertainty estimation of the forecasting over a long period of time.

The Deep-AR model [14] has successfully modeled multiple related time series data for traffic and electricity forecasting of households. Mixed Density Networks have been used to model complex data distributions that deviate from single-mode Gaussian patterns to an arbitrary conditional density function. [26], [27]. In many of such cases, the time series of the neighboring nodes are correlated and through joint training of multiple correlated time series, a substantial amount of data on past behavior of similar, related time series can be leveraged for making a forecast of an individual time series. Again in the spatial domain, if one considers the mobility flow between neighboring counties as a diffusion process that can capture the stochastic nature of the population dynamics. [28]

Motivated by the above, we propose the ARM3Dnet algorithm to model the spread of COVID-19 at multiple geographical regions simultaneously while also modeling the correlation among different time series sequence and the spatial dependency among these regions.

III. METHODOLOGY

In this section, we outline the problem of forecasting COVID-19 disease at a county level in the US. This is followed by a step by step description of our model.

A. Problem Formulation

Graphs are used to represent a wide variety of complex systems with interacting entities. We create a county level graph network where the relationship between the county locations is represented with a directed graph \mathcal{G} , where $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ is a set of N nodes representing each county in the US, \mathcal{V} is a set of nodes $|\mathcal{V}| = N$ and \mathcal{E} is a set of directed edges representing the connection between the nodes. $\mathcal{A} \in \mathbb{R}^{N \times N}$ is the weighted adjacency matrix representing the strength of connectivity between the nodes. The edges of the graph are connected on the basis of aggregated visits of people from an origin to a destination county on a particular day as gathered from the SafeGraph Community. Our model is based on two fundamental assumptions, firstly, the diffusion pattern across neighboring nodes has a fundamental statistical property where, after many time steps, such Markovian process converges to a stationary distribution $\rho \in \mathbb{R}^{N \times N}$, whose i^{th} row, $\rho_i \in \mathbb{R}^N$, represents the likelihood of diffusion from node $v_i \in \mathcal{V}$ and therefore proximity with respect to node v_i . Secondly, due to the extremely contagious nature of the disease, the people's mobility at the county level and the population diffusion in between the neighboring counties is bound to have some correlation with the spread of the disease in general. Therefore, given the historical covariates, like aggregated number of infection cases $X_{i,t}$, the target value at the previous time step $Z_{i,t-1}$, and the previous network output $h_{i,t-1}$, the goal of the network is to learn the following function:

$$h_{i,t} = \mathbf{h}(h_{i,t-1}, Z_{i,t-1}, X_{i,t}, \Theta)$$

In our model, the time covariate feature of the AR-LSTM is replaced by a diffusion convolution operation which is actually a graph convolution between county level people mobility information and the adjacency matrix which carries the spatial information. The overall architecture of our ARM3Dnet model is shown in Figure 2. It utilizes a graph-based encoder-decoder architecture that models the spatial correlation by a diffusion convolution operation on the graph and the temporal patterns by an auto-regressive long short-term memory (AR-LSTM) network.

B. Capturing the spatio-temporal correlation

The fundamental idea of a convolution layer in a neural network is to extract localized features from the input data by convolving with filters of various dimensions. Similarly, the core idea of a graph convolution layer is to extract the localized features from input data which is being represented as a graph structure. Therefore, the product of a neighborhood matrix \mathcal{A} , input data \mathcal{X}_t , and a trainable weight matrix \mathcal{W} , is considered as a graph convolution operation to extract features from a single hop neighborhood.

$$\mathcal{W}_G * \mathcal{X} = \mathcal{W}(\mathcal{D}^{-1}\mathcal{A})\mathcal{X} \quad (1)$$

Here, \mathcal{D} is the degree matrix, $\mathcal{D}^{-1}\mathcal{A}$ is a transition matrix of the diffusion process, and \mathcal{W} is a learnable filter for the diffusion process. Since the people's mobility across the counties are dynamic in nature, therefore our ARM3Dnet model uses a dynamic weighted adjacency matrix. We keep the adjacency matrix as dynamic by computing the pearson correlation of the aggregated visits among the counties on daily basis. Therefore, the elements in the adjacency matrix $\tilde{\mathcal{A}}$ at time T are given by,

$$\mathcal{A}_{i,j} = \rho_{x_i, x_j} = \frac{\text{cov}(x_i, x_j)}{\sigma_{x_i} \sigma_{x_j}}$$

where, $\mathcal{A}_{i,j}$ represents the edge weight between the nodes i and j ; x_i and x_j denotes the aggregated visits in between nodes i and j . The Pearson correlation co-efficient between x_i and x_j is represented by ρ ; cov is the covariance, σ_{x_i} and σ_{x_j} represents the standard deviation of the nodes x_i and x_j , respectively.

We found that a sparse and weighted adjacency matrix truly captures the correlation between the people diffusion, disease spread and the number of death forecasts. In total, our graph has 300 nodes and we have connected an edge between two counties whenever the aggregated visits people made between them crossed a threshold of 200 on a particular day. The graph convoluted output is fed to an auto-regressive recurrent encoder and decoder model as shown in Figure 3.

C. Modeling the likelihood function

Given, the value of time series i at time t by $Z_{i,t}$, our goal is to model the conditional distribution:

$$\mathcal{P}(Z_{i,t_0:T} | Z_{i,1:t_0-1}, \mathcal{W}_G * \mathcal{X}_{i,1:T})$$

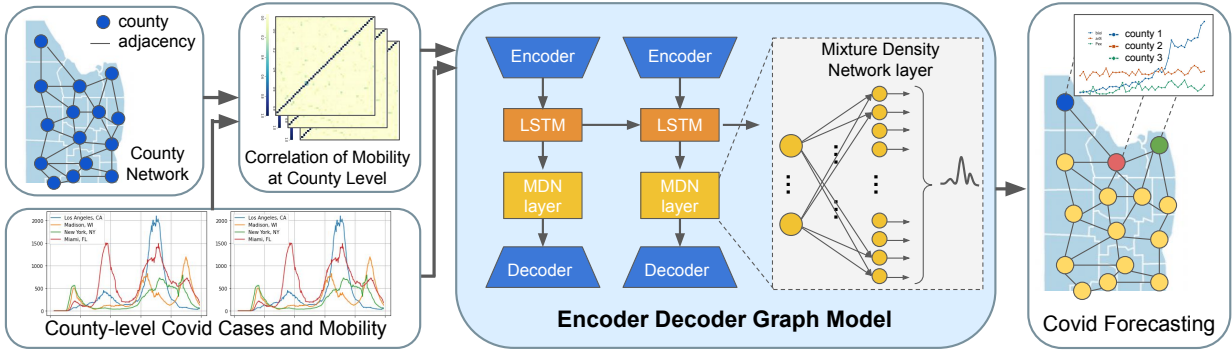


Fig. 2: An outline of the ARM3Dnet forecasting framework. The input to the network are the graph convoluted covariates, the target value of the previous step and the previous network output. The encoder network is followed by a MDN layer to capture the multimodal data. The output of the network is used to compute the parameters of the likelihood function.

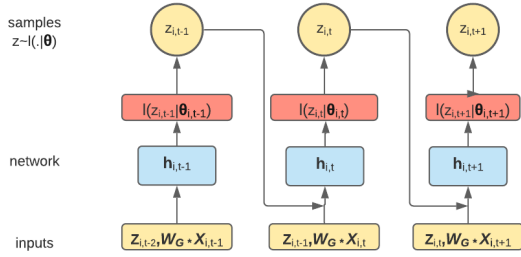


Fig. 3: A block diagram of the auto-regressive LSTM encoder-decoder network to jointly model the correlated time series information.

where, $Z_{i,t_0:T}$ and $Z_{i,1:t_0-1}$ represents the future and the past time series, respectively. Here, the dynamic covariates $W_G * \mathcal{X}$ are the graph convolution between aggregated people's mobility at the county level on daily basis and the county level adjacency matrix. During training, the data for the entire conditioning range $[1, t_0 - 1]$ and prediction range $[t_0, T]$ are observed but during the forecasting period $Z_{i,t}$ is available only for the conditioning range. We assume that the model distribution $Q_\theta(Z_{i,t_0:T} | Z_{i,1:t_0-1}, W_G * X_{i,1:T})$ consists of a product of the likelihood factors:

$$Q_\theta(Z_{i,t_0:T} | Z_{i,1:t_0-1}, W_G * X_{i,1:T}) = \prod_{t=t_0}^T Q_\theta(Z_{i,t} | Z_{i,1:t-1}, W_G * X_{i,1:T}) = \prod_{t=t_0}^T \ell(Z_{i,t} | \theta(h_{i,t}, \Theta)) \quad (2)$$

which is parameterized by the output $h_{i,t}$ of an autoregressive recurrent network.

$$h_{i,t} = \mathbf{h}(h_{i,t-1}, Z_{i,t-1}, W_G * X_{i,t}, \Theta)$$

The network output $h_{i,t}$, $Z_{i,t-1}$ is used to compute the parameters $\theta_{i,t}$ of the likelihood $\ell(Z | \theta)$ and that is used for training the model parameters. The likelihood $\ell(Z_{i,t} | \theta(h_{i,t}))$

is a fixed distribution whose parameters are defined by the function $\theta(h_{i,t}, \Theta)$ of the network output $h_{i,t}$. Given the model parameters Θ , joint samples are obtained through ancestral sampling as

$$\tilde{Z}_{i,t_0:T} \sim Q_\theta(Z_{i,t_0:T} | Z_{i,1:t_0-1}, W_G * X_{i,1:T}). \quad (3)$$

For $t = t_0, t_0 + 1, \dots, T$, the sampling is performed as $\tilde{Z}_{i,t} \sim \ell(\cdot | \theta(h_{i,t}, \Theta))$. Samples obtained from the model can be used to calculate the quantities of interest from the posterior distribution for a given time range in the future.

In this context, the likelihood function $\ell(Z | \theta)$ must be carefully chosen to match the statistical properties of the data. The network predicts the parameters mean and variance of the probability distribution for the next time point. In our case, we have chosen the Gaussian likelihood for predicting the death count data which is parameterized using its mean and standard deviation as given below:

$$l_G(Z | \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(Z - \mu)^2}{2\sigma^2}\right)$$

$$\mu(h_{i,t}) = W_\mu^T h_{i,t} + b_\mu$$

$$\sigma(h_{i,t}) = \log(1 + \exp(W_\sigma^T h_{i,t} + b_\sigma))$$

The standard deviation is often followed by a softplus activation function in order to ensure the value is positive always.

D. Mixture Density Layer to model the multimodal data

If we assume the conditional distribution of the target data to be Gaussian, we can estimate the parameters using the maximum likelihood. But it gives the idea of replacing the conditional distribution of the target vector with a mixture model that is flexible to completely model a general distribution function. Thus, we model the probability density of the target data as a linear combination of the Kernel functions.

$$\mathcal{P}(Z | \mathcal{X}) = \sum_{i=1}^m \alpha_i(x) \cdot p_i(Z | \mathcal{X})$$

where, m is the number of components in the mixture and $\alpha_i(x)$ are the mixing coefficients. When the Kernel functions

are assumed to be Gaussian in nature, the equations are of the form:

$$p_i(\mathcal{Z}|\mathcal{X}) = \frac{1}{(2\pi)^{c/2}\sigma_i(x)^c} \exp\left(-\frac{\|\mathcal{Z} - \mu_i(x)\|^2}{2\sigma_i(x)^2}\right)$$

where μ_i represents the centre of the i^{th} kernel.

Given this information, a Gaussian Mixture Model (GMM) with this simplified kernel can approximate any given density function to arbitrary accuracy, providing the mixing coefficients and the Gaussian parameters are correctly chosen [26]. Although the output of the Auto-regressive LSTM model minimizes a softmax loss function, these losses are not suitable for modeling the variation in the data over a long period of time. Thus a single Gaussian distribution often fails to model the multimodal nature of the real world data. Therefore, our main hypothesis for modeling the output distribution of the parameters is performing the regression on a multimodal data. Our model combines a traditional AR-LSTM model with a mixture density model that gives a general framework for modeling conditional density functions. Therefore, the conditional distribution for K Gaussian mixtures can be written as:

$$\mathcal{P}(\mathcal{Z}_{i,t}|h_{i,t}) \sim p_1\mathcal{N}(\mu_1(x), \sigma_1^2(x)) + p_2\mathcal{N}(\mu_2(x), \sigma_2^2(x)) \\ \dots + p_k\mathcal{N}(\mu_k(x), \sigma_k^2(x)) + \dots + p_K\mathcal{N}(\mu_K(x), \sigma_K^2(x))$$

where, p_k , μ_k and σ_k are the probability, mean and standard deviation respectively for the k^{th} Gaussian component. The parameters of the GMM are computed from the $h_{i,t}$ output of the recurrent layer as follows:

$$p_k = \frac{\exp(\theta_k^{(p)}h)}{\sum_{l=1}^K \theta_l^{(p)}h} \\ \sigma_k = \exp(\theta_k^{(\sigma)}h) \\ \mu_k = \exp(\theta_k^{(\mu)}h)$$

where, $\theta_k^{(p)}$, $\theta_k^{(\sigma)}$ and $\theta_k^{(\mu)}$ represents the learned parameter of the output layer corresponding to the mixture probability, variance and mean respectively.

E. The ARM3Dnet algorithm and the loss function

Given the time series $\mathcal{Z}_{i,1:T}$, $i = 1, \dots, N$ and associated graph convoluted covariates $\mathcal{GC}_{i,1:T}$, where the time range is chosen in such a way that $\mathcal{Z}_{i,t}$ in the prediction range is known, the parameters Θ of the model can be learnt by maximising the log-likelihood function.

$$\mathcal{L} = \sum_{i=1}^N \sum_{t=t_0}^T \log\left(\sum_{k=0}^m \alpha_k(\theta) \cdot \ell(\mathcal{Z}_{i,t}|\theta(h_{i,t}))\right) \quad (4)$$

where, $h_{i,t}$ is a deterministic function of the input and can be optimized directly via the stochastic gradient descent algorithm during backpropagation by computing the gradients with respect to Θ . The pseudocode of ARM3Dnet is provided in Algorithm 1.

Algorithm 1: ARM3Dnet Model

Input: $\mathcal{X} \in \mathcal{R}^{N \times P}$, $A \in \mathcal{R}^{N \times N}$;
Parameters: $\mathcal{W}_{g_{c_1}}, \dots, \mathcal{W}_{g_{c_n}}, \mathcal{W}, b$;
Initialize: $h_0 = 0, C_0 = 0 \in \mathcal{R}^N$;
for $t = 1$ **to** T **do**
 $\tilde{A} = \mathcal{D}^{-1}A$;
 $\mathcal{GC}_t = (\mathcal{W}_{gc} \circ \tilde{A}) \mathcal{X}_t$;
 $h_{i,t} = \mathbf{h}(h_{i,t-1}, \mathcal{Z}_{i,t-1}, \mathcal{GC}_{i,t}, \Theta)$;
 $\tilde{\mathcal{Z}}_{i,t} \sim Q_{\Theta}(\mathcal{Z}_{i,t}|\mathcal{Z}_{i,t-1}, \mathcal{GC}_{i,t})$.
 calculate σ ;
end
return $\tilde{\mathcal{Z}}_{i,t_0:T}$

For each of the county level time series data of covid, we create multiple training instances by selecting windows with different starting point from the original time series. Here, during training the total length of the time series T and the conditioning and prediction ranges are kept fixed for all training examples. We ensure that the entire prediction range is always covered by the ground truth data. During training, the values of $\mathcal{Z}_{i,t}$ are known in the prediction range and can be used to compute the model parameters $h_{i,t}$. However, during prediction values of $\mathcal{Z}_{i,t}$ are unknown in the time period $t \geq t_0$ and therefore a single sample $\tilde{\mathcal{Z}}_{i,t} \sim \ell(\cdot|\theta(h_{i,t}))$ from the model distribution is used to compute $h_{i,t}$. In a multivariate normal distribution, the standard approach is to maximise the log-likelihood function.

IV. EXPERIMENTATION

In this section, we discuss the datasets, experimental settings, performance metrics, the baseline methods, and their parametric settings, and the comparative results. Our implementation of the proposed model will be available at <https://github.com/padmaksha18/ARM3Dnet>

A. Datasets

In this study, we focus on county-level COVID-19 data and point-of-interest (POI) based Mobility data developed by SafeGraph. Here, we describe our data sources, methods of collecting them as well as two of our curated datasets (ML-Dataset and Covariate Feature Dataset) designed for its use in mobility-based future work on forecasting.

a) **COVID-19 data:** We collected the data from the public repository maintained by JHU and NYT. We used cumulative cases and deaths for each date and each county, from May to December of 2020.

b) **SafeGraph POI based Mobility Data:** We used the data from SafeGraph data (<https://www.safegraph.com>) to build a human mobility network. SafeGraph uses device location to build a dataset that details the number of visits to a POI from all other Census Block Groups (CBGs) on every single day based on a number of devices.

Using the POI based mobility data, the final derived dataset had attributes like Date, Origin County FIPS, Destination

County FIPS, Mean Distance traveled by a device (corresponding to Origin County), Aggregated Visits (from origin to destination) and Device Count. Aggregated Visits is used as the metric of inter county mobility flow whereas Mean Distance is considered intra-county mobility feature.

c) **Curated Mobility Dataset:** We create two different datasets using the aforementioned mobility features for different sets of models.

ML-Dataset: This dataset is used as mobility features for DeepCOVID-19, one of our baselines. The features are a combination of intra-county and inter-county mobility. The intra-county mobility feature is "mean distance traveled by a device" from SafeGraph. We derive 'inflow' as inter-county mobility corresponding to each county. We define inflow at time t and county i , as $\sum_j X_{j,i} * C_j$ (where $X_{j,i}$ is the aggregated visit where origin county is j and destination county is i ; C_j is the number of cumulative cases for county j).

Mobility-convoluted covariate features: The covariates in our case are time-dependent. It is based on the assumption that both inter and intra-county people mobility plays a key role in disease propagation (number of cases, here) through the county mobility network where each node is a county. The covariate dynamic time series information, when graph convoluted with the county level adjacency matrix formed on the basis of the weights given to the number of people traveling between them, will provide additional information to the model during prediction. The covariates are standardized to follow zero mean and unit variance and thus we create for each day a covariate dataset which is used by ARM3Dnet and DC-RNN.

B. Experimental Settings:

We used COVID-19 cumulative case and death data from May to December and experiment with different length of training data to do prediction for the next 7 days. We also changed the sparsity of the adjacency matrix by incorporating more counties in the graph and observed how our proposed model gave a better performance with an increase in the number of nodes. For the mobility features, we use inter and cross county mobility from publicly available data from SafeGraph.

C. Baseline Models

We use the following baseline methods for comparison.

- **SARIMAX:** This is a statistical auto regressive model for time series prediction. We use Augmented Dickey-Fuller Test to check if the time series data is stationary and then use grid-search for the selection of parameters.
- **LSTM:** Traditional RNN model ideal for sequential time series data. We consider a time window of 15 days.
- **DeepCovid-19** [24]: This is an existing deep learning-based model for COVID forecasting, that was designed for county level forecasting of positive cases using mobility features. Similar to LSTM, we use our ML-Dataset as a mobility feature.

- **DCRNN** [29]: This is a state-of-the-art deep learning framework for spatiotemporal forecasting. Our mobility-based Covariate feature dataset is used as the input from the graph convolution element of the model.
- **DeepAR** [14]: This is a state-of-the-art time series prediction model that deals with multiple correlated time series. We use the same covariate features for DeepAR as described earlier.

We simulate these baselines using the hyperparameter settings recommended by the literature. For our model, each covariate feature matrix is $\in \mathcal{R}^{T \times 1}$ where each row corresponds to the number of days in the time series data and the columns represent dynamic time series features like graph convoluted mobility features. In some cases, we found that considering the disease time series (number of cases) in the graph convolution operation also helps in improving uncertainty of prediction which indirectly supports the contagious nature of the disease. Under Gaussian likelihood, we use the Adam optimizer with early stopping to train the model for 50 epochs.

D. Metric for Evaluation:

We compared the accuracy of our forecasting on both numbers of cases and deaths using Normalized Root Mean Squared Error (NRMSE) and Normal Deviation (ND). The scale of the county level time series varies widely and therefore the normalized RMSE was considered. The uncertainty growth over time from the data was measured and compared with the ND metric to gauge the confidence of the predictions over the entire time period. They are calculated as follows.

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{N(T-t_0)} \sum_{i,t} (\mathcal{Z}_{i,t} - \hat{\mathcal{Z}}_{i,t})^2}}{\frac{1}{N(T-t_0)} \sum_{i,t} |\mathcal{Z}_{i,t}|},$$

$$\text{ND} = \frac{\sum_{i,t} |\mathcal{Z}_{i,t} - \hat{\mathcal{Z}}_{i,t}|}{\sum_{i,t} |\mathcal{Z}_{i,t}|}.$$

E. Results & Discussions

The results of our baselines comparison, which are normalized over 300 time series data, clearly shows that our proposed model significantly outperforms traditional deep learning and statistical models and even challenges state of the art models like Deep AR on the basis of ND and NRMSE. The purely statistical models like SARIMAX and the vanilla LSTM model performs worse in comparison to the other deep learning models like DeepCovid-19 and DC-RNN, where the inter and intra county mobility information has been incorporated as a feature during modeling. This justifies the need of the granular level mobility information while formulating such a problem.

Our ARM3Dnet model outperforms both DeepAR and DeepCovid-19¹, proving the fact that, the dynamic graph convoluted covariate features has important correlation with the

¹DeepCovid-19 is used for predicting cases by using a combination of Deep learning and SEIR model. At any given time they calculate the infected and recovered number of population to get cumulative positive cases, and currently do not provide a method for predicting number of deaths

TABLE I: Predictive results of ARM3Dnet and other baseline methods in forecasting the number of cases and deaths due to COVID-19. The values are normalized over 300 time series.

| Models | Metrics | Cases Forecasting | | Deaths Forecasting | |
|----------------|---------|-------------------|--------------|--------------------|--------------|
| | | NRMSE | ND | NRMSE | ND |
| SARIMAX | | 0.210 | 0.033 | 0.129 | 0.070 |
| Ensembled LSTM | | 0.368 | 0.064 | 0.203 | 0.065 |
| DC-RNN | | 0.323 | 0.046 | 0.096 | 0.033 |
| DeepCovid19 | | 0.189 | 0.065 | N/A | N/A |
| DeepAR | | 0.170 | 0.035 | 0.054 | 0.024 |
| ARM3Dnet | | 0.091 | 0.024 | 0.051 | 0.018 |

output time series. Implementing an autoregressive recurrent neural network and using GMM also proves to improve the forecasting performance as we also manage to outperform DCRNN in terms of accuracy and uncertainty. Accumulating the mobility diffusion information across nodes and node level people mobility data has helped our model to beat a state of the art Deep AR model on both the metrics as shown in Figure 4 for uncertainty estimation. Since the real time covid data show multiple peaks over the period of a year and is truly multimodal in nature, optimizing the GMM with the correct number of components can accurately model this complex data distribution. Our GMM implementation uses 5 mixture components to accurately model the mixed distribution.

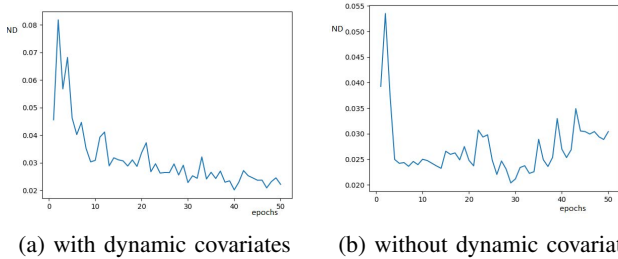


Fig. 4: Illustrating the effects of dynamic covariates in the ARM3Dnet performance by plotting ND(dynamically generated at each epoch) vs epochs. We observe that incorporating the dynamic covariates leads to smoother convergence of uncertainty.

We find that the model being aware of the previous information of all the related time series instead of a different model being fitted for each time series independently actually improves both the forecasting accuracy and uncertainty as it can learn the important correlation between all the time series. Thus, sharing information across time series can improve forecast accuracy. The improvement can be seen in the results of our model when compared with an ensemble LSTM model created to train over the different time series independently. The uncertainty growth curve Figure 5 shows that our model likelihood very closely adjusts itself with the ground truth data in the forecasting range, where we have taken 12 samples for every iteration during training when the mobility information is incorporated in the covariates.

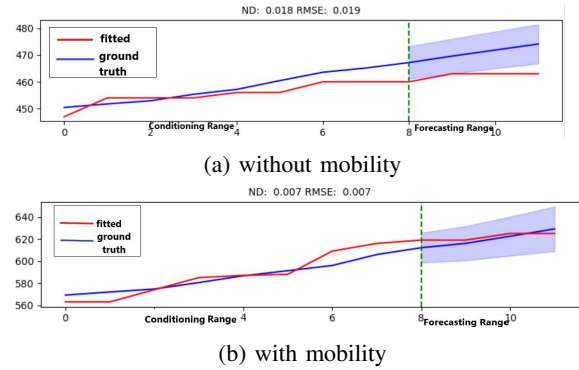


Fig. 5: Likelihood estimation for parameters with best ND values (a) with mobility, and (b) without mobility. During stages of training we notice that on adding the mobility information, the model likelihood follows the ground-truth curve more closely in the forecasting range.

At its core, our proposed model depends on utilizing the correlation information of the people diffusion between the neighbouring counties and the county level people mobility with the spread of the disease. Therefore we tried to obtain a best threshold for our connected graph network. We found that increasing the sparsity of the adjacency matrix by adding around 300 nodes and assigning more weight-age by connecting only those edges where the aggregated number of visits have crossed 200 on daily basis significantly increases our forecasting results. The results verify our initial assumption that people mobility and the spread of the disease will bear significant correlation during different phases of the pandemic as we can see in Figure 6.

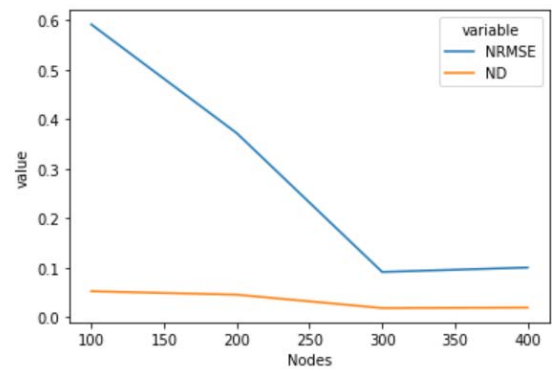


Fig. 6: We observe that on adding more number of nodes to the graph network improves performance of our model initially.

While training the entire data, we have observed sudden spikes in the loss function during some batches and this can be attributed to the fact that the NYT COVID data has the aggregated data missing for a consecutive number of days for some counties and are instead replaced by single day count of cases or deaths. We believe that addressing these data-related issues can further improve our model's performance.

V. CONCLUSIONS

In this paper, we tackle the problem of spatiotemporal forecasting of COVID-19 and similar infectious disease at a spatially granular level. We observed that incorporating human mobility network structure and mobility flow information as a covariate feature could empower modern deep learning-based techniques and improve both forecasting accuracy and uncertainty estimation over a long period of time. The proposed ARM3Dnet framework creates a global model based on multiple related time series data and models the complex spatiotemporal dependency among multiple nodes in the human mobility network. Structural information like population diffusion and within county people mobility generates calibrated forecasts with high accuracy when modeled over a vast geographical region involving multiple counties.

In future, we would like to address the skewed nature of real world data using more robust data distribution likelihoods. We also believe that there is further scope in experimenting with different forms of the GMMs like the Gaussian Processes to more accurately capture the multi-modal nature of real-world data and our model can be leveraged for different datasets in the context of spatiotemporal forecasting. Also, we believe that hyperparameter search and optimization [30], [31] of probabilistic disease forecasting models will help to improve the predictive performance, especially in case of the gaussian mixture to choose the optimum set of mixture components.

ACKNOWLEDGMENT

This research is supported by National Science Foundation grant DGE-1545362. **Disclaimer:** The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any institution, NSF, or the U.S. Government.

REFERENCES

- [1] K. Nikolopoulos, S. Punia, A. Schäfers, C. Tsinopoulos, and C. Vasilakis, "Forecasting and planning during a pandemic: Covid-19 growth rates, supply chain disruptions, and governmental decisions," *European journal of operational research*, vol. 290, no. 1, pp. 99–115, 2021.
- [2] I. Cooper, A. Mondal, and C. G. Antonopoulos, "A SIR model assumption for the spread of COVID-19 in different communities," *Chaos, Solitons & Fractals*, vol. 139, p. 110057, 2020.
- [3] S. Roy, G. S. Bhunia, and P. K. Shit, "Spatial prediction of COVID-19 epidemic using ARIMA techniques in India," *Modeling Earth Systems and Environment*, pp. 1–7, 2020.
- [4] P. Melin, J. C. Monica, D. Sanchez, and O. Castillo, "Multiple ensemble neural network models with fuzzy response aggregation for predicting COVID-19 time series: The case of Mexico," *Healthcare (Basel)*, vol. 8, no. 2, Jun. 2020.
- [5] M. Kim, J. Kang, D. Kim, H. Song, H. Min, Y. Nam, D. Park, and J.-G. Lee, "Hi-COVIDNet: Deep learning approach to predict inbound COVID-19 patients and case study in south korea," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3466–3473.
- [6] L. M. Rossen, A. M. Branum, F. B. Ahmad, P. Sutton, and R. N. Anderson, "Excess deaths associated with COVID-19, by age and race and ethnicity—United States, January 26–October 3, 2020," *Morbidity and Mortality Weekly Report*, vol. 69, no. 42, p. 1522, 2020.
- [7] S. Khalatbari-Soltani, R. C. Cumming, C. Delpierre, and M. Kelly-Irving, "Importance of collecting data on socioeconomic determinants from the early stage of the COVID-19 outbreak onwards," *J Epidemiol Community Health*, vol. 74, no. 8, pp. 620–623, 2020.
- [8] S. Das, "Prediction of COVID-19 disease progression in India: Under the effect of national lockdown," *arXiv preprint arXiv:2004.03147*, 2020.
- [9] L. Thunström, S. C. Newbold, D. Finnoff, M. Ashworth, and J. F. Shogren, "The benefits and costs of using social distancing to flatten the curve for COVID-19," *Journal of Benefit-Cost Analysis*, vol. 11, no. 2, pp. 179–195, 2020.
- [10] A. Kapoor, X. Ben, L. Liu, B. Perozzi, M. Barnes, M. Blais, and S. O'Banion, "Examining COVID-19 forecasting using spatio-temporal graph neural networks," *arXiv preprint arXiv:2007.03113*, 2020.
- [11] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2008.
- [12] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *CoRR*, vol. abs/1609.02907, 2016. [Online]. Available: <http://arxiv.org/abs/1609.02907>
- [13] K. Bandara, C. Bergmeir, and S. Smyl, "Forecasting across time series databases using long short-term memory networks on groups of similar series," *arXiv preprint arXiv:1710.03222*, vol. 8, pp. 805–815, 2017.
- [14] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "Deepar: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.
- [15] 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.
- [16] V. K. R. Chimmula and L. Zhang, "Time series forecasting of COVID-19 transmission in Canada using LSTM networks," *Chaos, Solitons & Fractals*, vol. 135, p. 109864, 2020.
- [17] J. Shaman, E. Goldstein, and M. Lipsitch, "Absolute humidity and pandemic versus epidemic influenza," *American journal of epidemiology*, vol. 173, no. 2, pp. 127–135, 2011.
- [18] Q. Zhang, N. Perra, D. Perrotta, M. Tizzoni, D. Paolotti, and A. Vespignani, "Forecasting seasonal influenza fusing digital indicators and a mechanistic disease model," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 311–319.
- [19] P. Chakraborty, P. Khadivi, B. Lewis, A. Mahendiran, J. Chen, P. Butler, E. O. Nsoesie, S. R. Mekaru, J. S. Brownstein, M. V. Marathe *et al.*, "Forecasting a moving target: Ensemble models for ILI case count predictions," in *Proceedings of the 2014 SIAM international conference on data mining*. SIAM, 2014, pp. 262–270.
- [20] L. C. Brooks, D. C. Farrow, S. Hyun, R. J. Tibshirani, and R. Rosenfeld, "Flexible modeling of epidemics with an empirical bayes framework," *PLoS Comput Biol*, vol. 11, no. 8, p. e1004382, 2015.
- [21] N. Kumar and S. Susan, "COVID-19 pandemic prediction using time series forecasting models," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2020, pp. 1–7.
- [22] M. Liu, R. Thomadsen, and S. Yao, "Forecasting the spread of COVID-19 under different reopening strategies," *medRxiv*, 2020.
- [23] R. Goel and R. Sharma, "Mobility based sir model for pandemics - with case study of covid-19," in *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2020, pp. 110–117.
- [24] M. A. Bhouri, F. S. Costabal, H. Wang, K. Linka, M. Peirlinck, E. Kuhl, and P. Perdikaris, "COVID-19 dynamics across the US: A deep learning study of human mobility and social behavior," *medRxiv*, 2020.
- [25] B. Adhikari, X. Xu, N. Ramakrishnan, and B. A. Prakash, "Epiddeep: Exploiting embeddings for epidemic forecasting," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 577–586.
- [26] C. M. Bishop, "Mixture density networks," 1994. [Online]. Available: <https://publications.aston.ac.uk/id/eprint/373/>
- [27] G. J. McLachlan and K. E. Basford, *Mixture models: Inference and applications to clustering*. M. Dekker New York, 1988, vol. 38.
- [28] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.
- [29] —, "Graph convolutional recurrent neural network: Data-driven traffic forecasting," *CoRR*, vol. abs/1707.01926, 2017.
- [30] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated machine learning*. Springer, Cham, 2019, pp. 3–33.
- [31] S. Biswas, A. D. Cobb, A. Sistrunk, N. Ramakrishnan, and B. Jalaian, "Better call Surrogates: A hybrid Evolutionary Algorithm for Hyperparameter optimization," *arXiv preprint arXiv:2012.06453*, 2020.