

Explainable Prediction of the Severity of COVID-19 Outbreak for US Counties

Shailik Sarkar*, Abdulaziz Alhamadani*, and Chang-Tien Lu*

* Department of Computer Science, Virginia Tech, Falls Church, VA 22043 USA

{shailik, hamdani, ctlu}@vt.edu

Abstract—Ever since the COVID-19 outbreak, various works have focused on using multitude of different static and dynamic features to aid the prediction of disease forecasting models. However, in the absence of historical pandemic data these models will not be able to give any meaningful insight about the areas which are most likely to be affected based on preexisting conditions. Furthermore, the black box nature of neural networks often becomes an impediment for the concerned authorities to derive any meaning from. In this paper, we propose a novel explainable Graph Neural Network (GNN) framework called Graph-COVID-19-Explainer (GC-Explainer) that gives explainable prediction for the severity of the spread during initial outbreak. We utilize a comprehensive set of static population characteristics to use as node features of Graph where each node corresponds to a geographical region. Unlike post-hoc methods of GNN explanations, we propose a framework for learning important features during the training of the model. We further apply our model on real-world early pandemic data to show the validity of our approach. Through GC-Explainer, we show that static features along with spatial dependency among regions can be used to explain the varied degree of severity in outbreak during the early part of the pandemic and provide a framework to identify the at-risk areas for any infectious disease outbreak, especially when historical data is not available.

Index Terms—Graph, COVID-19, explainable, COVID, GCN, GNN, forecasting, prediction, deep learning, graph neural network

I. INTRODUCTION

Between 1980 and 2013, there were over 12,000 outbreaks of human infectious diseases in 219 countries, including Zika, Ebola, Cholera, MERS, and H1N1, which reached pandemic thresholds [17], [23]. Studies on epidemic trend forecasting and epidemic outbreak early detection have become increasingly important due to the significant losses generated by epidemics. In the past 30 years, there has been a steady increase in the frequency and diversity of outbreaks. The outbreak of COVID-19 virus has also resulted in substantial human and economic loss. The existing works in COVID-19 modeling or analysis focus largely on utilizing the historical pandemic data to make more accurate prediction for cumulative number

of cases or deaths. Most of the work in COVID forecasting focuses singularly on achieving accurate forecasting for a period of time in future, either by predicting the number incidence or death. However, very few works utilize population level static characteristics to predict the severity of the spread at different areas and it is difficult to extract sufficient insight into the affect of these static features during the early outbreak ([4], [13], [15]). In this paper we focus on modeling this irregular spreading pattern of the infection across the United States during the early outbreak of the pandemic.

In the beginning of the outbreak, a disparity in the spread of the disease were observed across different regions. As a result, a cursory observation of the situation could definitely draw conclusions regarding the demography based vulnerability to the virus and certain conclusions regarding racial susceptibility were made both by mainstream media and academia. However, further inspection found that the contributing factors to such observation were often the underlying population characteristics including but not limited to socioeconomic status. Furthermore, the affect that neighboring regions might have on each other was also not explored. While there are works exploring the effect of socioeconomic features, they mostly focus on statistical analysis instead of exploring the viability of predictive model based solely on these features. Alhamadani et al. [3] on the other hand, did use an exhaustive collection of socioeconomic features to predict the rate of infection at zip-code level however while incorporating spatial dependency, the proposed GCN model does not provide node level feature explanation. Similarly, other works like Arik et al. [4], that have tackled the problem of incorporating explanation into the predictive model, provide explanation on dynamic features like, mask usage or mobility which are all affected by the lockdown measures or a population's behavior post outbreak. However, they do not focus on spatial dependency among regions or the static features, and whether or not they can be used to predict the severity of the outbreak when such dynamic historical pandemic data, or mobility data are not available.

To this effect, we have the following research questions: **Q1.** In the absence of any historical statistics about pandemic **how do we identify the regions most at risk?** **Q2) Does modeling the spatial dependency of the neighborhood improve the predictive performance?** **Q3) How do we provide explanations about the static population characteristics affecting the spread that can be useful for the concerned authorities?**

Through this paper, we show how both population characteristic features and spatial correlation among regions can be used for predicting the most severely affected regions. We formulate the problem as binary classification on graph, details of which are provided in section III. To this effect, we propose an explainable graph convolution network that gives interpretable prediction of the most severely effected areas. Most of the works in explainable COVID-19 forecasting have previously only focused on post-hoc explanation about the feature importance based on the drop in accuracy at the iterative exclusion of select features. Even in works related to explainable GNNs, the methodology largely consists of generating explanation for an already trained model. In contrast to these works, our proposed model incorporates a feature selection pipeline as the first step to be used for node-level feature aggregation through adjacency matrix based convolution at the graph convolution stage. While the connections for the learned adjacency matrix provide explanation for most important spatial correlation for each region, the feature selection through the incorporation of group lasso penalty gives more weight to the most important node features while penalizing unimportant features to almost zero. This is used as explanation about the prediction of the model. However, this is a particular problem setting that hasn't been tackled by papers related to "COVID-19 modeling" or "community health outcome prediction" which creates a lack of benchmark models. Hence, we conduct extensive experiments with different types of machine learning models including traditional GNN like GCN which substantiates the quality of the proposed framework. The contribution of our work can be summarized as follows:

- Unlike post-hoc methods for GNN explanation **we propose a novel framework Graph-Covid-Explainer that simultaneously gives prediction for high risk areas as well as insights about most for important features during the training of the model.**
- We introduce a **novel problem setting that tackles the paucity of historic data to identify high risk areas during the initial outbreak that can help authorities in better preparing for future crisis.**
- We apply our model Graph-COVID-Explainer(GC-Explainer) on real-world COVID-19 data to show that **static features about mobility, socioeconomic status and spatial dependency among regions can be used to make explainable prediction** about the varied degree of severity during early part of the outbreak without using historical pandemic data as features.
- We **build a county-level dataset containing the most important static features and the corresponding labels** indicating the rate of infection during the early part of the outbreak along with the adjacency matrix explaining the spatial dependency between different regions.

II. RELATED WORK

A. Covid Forecasting and Prediction:

Some of the previous works in this respect are in compartmental model-based simulation [7], [8]. Cooper et al. studied

the effectiveness of the modeling approach on pandemic due to the spread of the novel COVID-19 disease [6]. While these compartmental models have their advantages in giving insights by dividing the population into susceptible, exposed, infected, or recovered states, they are highly dependent on the surveillance data provided by the Centers for Disease Control and Prevention (CDC) [2]. Deep learning and other time series methods have gained popularity [10] as more epidemic-related data become available, these statistical and deep learning models become viable options for the forecasting task. Most of the current studies which model the spread of disease in deep learning-based approaches are works like [1], [12] and [5]. However, these methods often fail to provide explanations about the prediction which is particularly useful at the early stages of the outbreak when there is a dearth of pandemic data and an explanation behind prediction is highly desirable for the policy makers to employ the correct measures. However, there are few works that tries to explore the underlying disease dynamics by examining the effect that certain variables have. For example, Rodriguez et al. [13] proposes purely data driven approach while also providing a post-hoc explanation module. While the explanation of the model is based on the evaluation of the drop in accuracy for each set of features' exclusion, this does not provide a feature selection pipeline through which we can only interact with most pertinent features. This is a post-hoc explanation module to explain the predictions that the model has already made. In contrast. Arik et al. [4] which is a compartmental model, incorporates static and dynamic features as covariates to account for the transition from one compartment to the other. While, this is a truly interpretable model. it only provides insights about dynamic features and their usefulness. Other works like Roy et al. [14] relied solely on mobility data for COVID forecasting. Although the paper provided compelling evidence related to the importance of mobility data. it once again does't provide any interpretation about the static socioeconomic factors. Alhamadani et al. provided significant insight about the validity of socioeconomic features in determining the worst affected areas. [3] However, they do not propose a model that incorporates feature selection into the predictive model. One of the experiments from this work also suggests that spatial dependency among nodes can improve the prediction.

B. Graph Neural Networks and Explainability:

Hence, we look towards the vast research area of Graph Neural Network that includes but are not limited to works like GCN [9], GAT [18] or GIN [21]. Despite using spatial correlation and node level feature aggregation, all these models do not provide satisfactory explanation about the data and how it effects the prediction. Therefore, we have to focus on going through some of the more popular works in the area of explainable graph neural network. Ying et al. proposed GNNExplainer a method to identify significant graph paths and underline corresponding node feature information that is distributed through the pathway edges by implementing a recursive neighborhood-aggregation technique [22].Luo et al [11] explained predictions

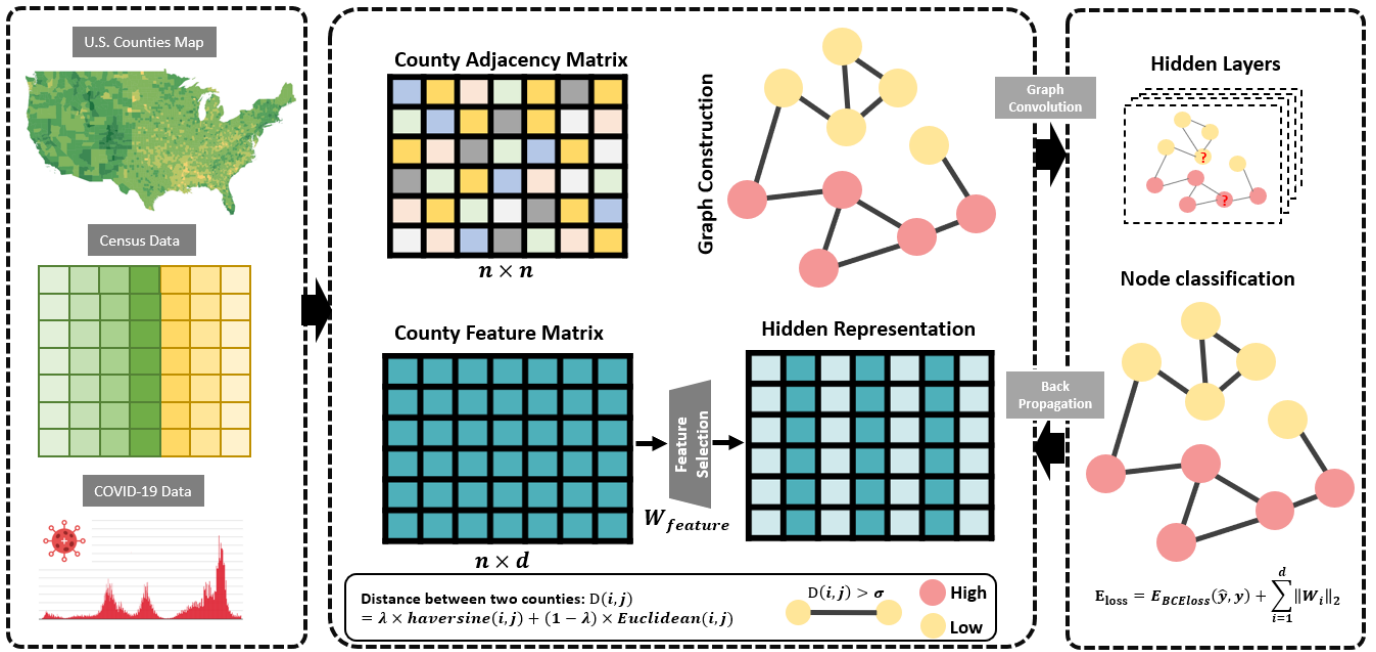


Fig. 1: Workflow diagram for the proposed GC-Explainer framework

of GNNs by proposing a generative probabilistic method. The utilization of generative methods allowed PGExplainer to obtain the explanations from the explored inherent structures of the generative models. Furthermore, GraphMask [16] is another post-hoc method similar to PGExplainer where it learns a classifier for every edge in every layer in a trained GNN model and predicts if an edge can be dropped. However, the classifier is differentiable and utilizes stochastic gates and L0 regularization. GNNExplainer, for example, does provide node level and feature level interpretation of the outcome. Similarly PG-explainer provides similar explanation about prediction, despite not giving any interpretation about the node features. Both these models fall under perturbation-based model that provides post-hoc explanation that requires a GNN model to be trained first. That is why we are exploring the possibility of building a Graph Convolutional network that will learn the most important features during the training process.

III. PROBLEM STATEMENT:

In this work we are trying to identify areas with higher severity of COVID-19 spread during the early outbreak, based solely on static population features that can influence the intensity of the spread. To treat this as a binary classification we have to define "high risk" areas. To standardize the severity of the spread we consider both the rate of increase of cases as well as the total number of cases standardized by the population. For total number of cases we use the New York Times dataset that reports both official and probable number. Also, we consider the same time frame for every county. For example, if a county has its first case in March, then for that county we start collecting the data from March while for another county where the first incidence happened around

February the data collection starts from February. Thus, for every county we used the collected data for the first forty five days of the outbreak to calculate the rate of the spread in standardized population (per hundred thousand) to label each county as either high or low-risk area. Based on the standardized infection rate for each county, we find out that the mean value for this metrics is 278 however only 729 of 3000 counties have values above that which indicates these counties are the more disproportionately affected areas. Hence, we label those counties as high risk areas and others as comparatively low. Our goal is to not only predict the high risk areas based solely on population-level static characteristics but also to provide explanation as to the contributing features and spatial correlation. For this purpose we formulate the problem as node classification over spatial graph network of counties.

The mathematical formulation of the problem can be stated as follows: For a given graph G and a set of n nodes $C = \{c_1, c_2, \dots, c_n\}$ with corresponding labels $Y = \{Y_1, Y_2, \dots, Y_n\}$ (where Y_i is 0 for low-impact area and 1 for high-impact area), each having static features $x = (x_1, x_2, x_3, \dots, x_f)$, we want to find a function f where $f_{\theta}(X_s, A) = Y$ where X_s is a subset of X , and consisting only of contributing factors, A is the adjacency matrix of the graph structure, and Θ is the learnable parameter of the model. We are aiming to learn both Θ and X_s for the set of nodes C . Hence, essentially it is a node classification problem with added focus on explainability through feature selection pipeline.

IV. REVIEWING GRAPH CONVOLUTIONAL NETWORK:

Graph Convolution Network(GCN)s are a specific kind of Graph Neural Network(GNN) that uses spatial level feature

aggregation as convolution operation on a given node vector. It was originally proposed by [9] for semi-supervised classification setting. The main idea can be summarized as follows: Given a graph G with node vector $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2 \cdots \mathbf{x}_n) \in \mathbb{R}^{n \times d}$ and an adjacency matrix $A \in \mathbb{R}^{n \times n}$, the output of the convolutional layer can be expressed as:

$$\mathbf{H} = \hat{\mathbf{A}}\mathbf{X}\Theta \quad (1)$$

Here $\hat{\mathbf{A}}$ denotes a normalized Adjacency matrix that is calculated as follows: $\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} + \mathbf{I}$ where \mathbf{D} and \mathbf{I} are diagonal node degree and identity matrix respectively. Depending on the number of hidden layer, a layer i in a GCN can be expressed as:

$$\mathbf{H}^i = \hat{\mathbf{A}}\mathbf{H}^{i-1}\Theta \quad (2)$$

while the final layer can be something to the effect of the effect:

$$\mathbf{H}^{final} = \sigma(\mathbf{H}^l\Theta) \quad (3)$$

Here, σ is an activation function that can vary from anything from a Relu or LeakyRelu to a Sigmoid function. Based on the loss function employed by the model and choice of optimizer, the parameters Θ will be updated through back-propagation.

V. METHODOLOGY: GC-EXPLAINER

In this section we will describe the details about our proposed framework Graph-COVID-19-Explainer(GC-Explainer). This section is divided into three section that describes the feature selection module, graph convolutional network and loss function with group lasso penalty.

A. Feature Selector Layer

This is a single hidden layer that takes as input the original feature vector $\mathbf{X} \in \mathbb{R}^{n \times d}$ where n is the number of instances in the input and d denotes the number of initial input features. The hidden layer has p number of nodes which are all fully connected. This means the weight vector for this layer is of the dimension $(d \times p)$ and the output of the layer can be written as follows:

$$\mathbf{H}_{feature} = \mathbf{X}\mathbf{W} + bias \quad (4)$$

The output of this layer is thus a tensor of size $n \times p$. The main objective of having this layer as a buffer before feeding it into the convolutional layer is to penalize the less important features. To further elaborate the, figure 2 shows the idea as feature sparsification which is essentially applied to penalize all the weights connected to less important features.

B. Group Lasso Penalty for feature sparsification

To achieve explainability we aim for sparsification of weight vector by only highlighting the most important feature through learning the sparsified weight matrix. We do this by means of group lasso or $L_{(1,2)}$ regularization term.

$$penalty = \sum_{i=1}^d \|W_i\|_2 \quad (5)$$

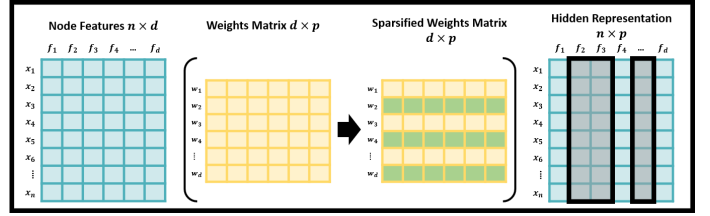


Fig. 2: Feature Slection through Weight Sparsification

Here, W_i corresponds to each vector of the previously used tensor in equation 4. The column vectors denote the weight associated with the corresponding hidden layer node. We hypothesize that this will mitigate the effect of including noisy unnecessary features on the original loss function as it is more prone to achieving a local minima during convergence. Therefore, this module works rather as a feature learner with more importance given to most important features which works as an inherent explanation for our model as depicted in figure 2.

To alleviate the problem of non differentiable penalty term, we further add a smoothing function akin to [20] and other works [19].

C. Graph Convolutional Layers

The inputs for this stage includes the output from the hidden layer described in equation 4 as $H_{feature}$ of the size $n \times p$ where n denotes the number of instances and d denotes the dimension of the extracted feature vector. Hence, for our proposed GC-Explainer the instead of using node vector we are going to use transformed $H_{feature}$ vector. We formulate the graph as $\mathcal{G} = (\mathcal{H}, \mathcal{E}, \mathcal{A})$ with n nodes. The nodes \mathcal{H} represent the feature transformed instances of n counties.

Adjacency Matrix construction for spatial explanation: Here the edges are expressed by \mathcal{E} indicating spatial dependency. Adjacency matrix is used to represent this spatial dependency formed by interconnected counties. Hence, the adjacency matrix is of the form: $\mathcal{A} = [A_{ij}] \in \{0, 1\}^{n \times n}$. Works in the past have hypothesized that geographically, the closer a region is to another, the higher the spatial dependency will be between the pair. However, we hypothesize that within a certain neighborhood of a region, a county further away can effect the spread of the virus more than their immediate adjacent county because of factors varying from job related inter-county commute, to demographic and other population characteristics. Hence we compute pairwise distance of a node within it's k -nearest neighbor based on havershine distance. Then we compute connection distance through the following equation:

$$Distance(\mathbf{X}_{i,j}) = \lambda Havershine + (1 - \lambda) Euclidian \quad (6)$$

Here, λ is a control parameter that controls the weight given to geographic havershine distance or euclidian distance between the features. Then, we connect two vertices only if the distance is lesser than a threshold. In our experiment for

3120 vertices we experiment with the threshold to control sparsity by starting with 5561 edges with k set to 5. In our final implementation, we finalize on the adjacency matrix with 3328 edges. $H_{feature} \in \mathcal{R}^{n \times p}$ represents the output of the feature selection task described in subsection V-A.

Node Feature Propagation: After designing the county graph structure, the GCN model is used for the node-level information to propagate according to the neighborhood relationships. One layer of the propagation rule is defined as the following convolution:

$$g_\theta * H \approx (\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}) H_{feature} \Theta \quad (7)$$

This forward pass is parameterized with Θ , with $\tilde{A} = A + I$, where I is the identity matrix and \tilde{D} is the diagonal node degree matrix of \tilde{A} . By stacking three layers of the graph convolutions, our final model is defined as follows:

$$Y = \sigma(g_{\theta_3} * g_{\theta_2} * (g_{\theta_1} * H_{feature})) \quad (8)$$

Here, σ is the activation function.

D. Computing Overall Loss Function:

Computing the overall loss function in each epoch includes computing both the binary cross entropy loss, as well as the group lasso penalty function. Backpropagation is used for updating the weights at each layer based on the computed gradients. The overall loss function for the model will look something like this:

$$\mathbf{E}_{\text{loss}} = (1 - \beta) \times \mathbf{E}_{BCELoss}(\hat{Y}, Y) + \beta \times \sum_{i=1}^d \|W_i\|_2 \quad (9)$$

Our objective is to minimize the loss while also updating the parameters of the model until convergence.

E. Training Algorithm:

In this section we explain the overall training process for the model through Algo 1. .

Algorithm 1 Model

Input: $\mathcal{X} \in \mathcal{R}^{N \times D}$, $A \in \mathcal{R}^{N \times N}$, $\mathcal{Y} \in \mathcal{R}^{N \times 1}$;

Output: $\hat{Y} \in \mathcal{R}^{N \times 1}$

Parameters: $\mathcal{W}_{feature}, \Theta$;

Initialize parameters;

for each epoch **do**

$H_{feature} = X.W_{feature} + bias$;

$\tilde{A} = \tilde{D}^{-\frac{1}{2}}(A + I)\tilde{D}^{-\frac{1}{2}}$;

$g_\theta * H \approx (\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}) H_{feature} \Theta$;

$\hat{Y} = \sigma(g_{\theta_3} * g_{\theta_2} * (g_{\theta_1} * H_{feature}))$;

$Loss = \mathbf{E}_{BCELoss}(\hat{Y}, Y) + \sum_{i=1}^d \|W_i\|_2$

Loss.backward()

Update $\mathcal{W}_{feature}, \Theta$;

end for

VI. EXPERIMENTS

In this section we talk about data collection, baseline comparisons and experimental settings and discuss the results.

A. Data Collection

Our predictive analysis is based on two separate kind of data.

COVID-19 cases Dataset: For each of the counties, the number of cumulative and probable cases has been collected, for each day starting from the outbreak to next forty five days individually, from the New York Times COVID-dataset. We use the population of each county to obtain a weighted rate confirmed COVID cases based on which we label counties as high severity or low.

Census and Mobility Data Collection: For each counties we also collect census and mobility features. American Census Surveys (ACS)¹ are a rich source for socioeconomic features, for mobility however, we collect data from SafeGraph for the days preceeding the first incident. SafeGraph (<https://www.safegraph.com>) 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. The dataset is used to calculate the static average of "mean distance traveled from home" in each county prior to the outbreak.

B. Experiment Settings

As explained before, we are tasked with the problem of predicting high regions based solely on static population level characteristics. Our procedure to label high risk areas is described in Section III. There is a significant lack of work that focuses on predicting such high risk areas in the absence of any historical data based purely on static population characteristics. Due to the unique nature of the task, we are faced with lack of benchmark in the literature of covid modeling to compare our framework to. However, it is our understanding that given significant number of works discovering the effect of socioeconomic and population factors on the spread of the virus, it is important to explore how we can use these static preexisting features to predict the at-risk areas. To ensure that our proposed methodology is effective, we have to compare this with different machine learning and deep learning models that are known to perform well for classification task using structured data such as ours (as opposed to image or text). The baselines are as follows:

- Logistic: We implement a logistic regression model as they generally gives good performance for classification tasks.
- SVM: SVM is a powerful model for the classification task with data of our size.
- XGBoost: Extreme Gradient Boosting is a variation of Gradient Boosting Decision Trees (GBDT), which we use to see if our feature selector module can improve the performance of decision tree based methods.
- MLP: To study the efficiency of neural networks we use a standard Feed Forward Neural Network that can potentially capture nonlinear interaction among the features.

¹<https://data.census.gov/cedsci/table>

TABLE I: Experimental Results

Model	Accuracy	F1	Precision	Recall
Logistic	0.6311	0.6159	0.5933	0.6402
SVM	0.6255	0.6091	0.5965	0.6223
XGBoost	0.6188	0.6200	0.6087	0.6214
MLP	0.6380	0.6318	0.6271	0.6366
GCN	0.7033	0.7053	0.7005	0.7102
GC-Explainer	0.7256	0.7252	0.7192	0.7315

- GCN: To examine if incorporating spatial dependency affects the performance in a positive or negative way and to do comparison with our own model, we employ a standard GCN model with two convolutional layer stacked on top of each other.

These models work as base model comparison described in Table I and are discussed in detail later. To account for the disproportionate distribution of labels in the problem, we split the data through each label in train and test set (80:20 split) and then combine it to train the model on training dataset before validating on test set to check for the performance. The data is standardized by following the min max standardization. We use pytorch library of python to implement all the models. Hyperparameters are set by standard grid-search method. For GC-Explainer, the learning rate is set for 0.0001 and we train the model for 10000 epochs. We further discuss the insights explanation provided by our proposed model about the selected features from the exhaustive list of socioeconomic features. It is important to note that in this section while working with GCN models we use the optimal adjacency matrix that we discovered in our ablation study, which is described in a later subsection. As we are predicting the label of each county in terms its outbreak severity, we use accuracy, F1, precision and recall score to make comparison.

C. Experimental Results

In Table I we can see the performance of each of the baseline models compared with the proposed GC-Explainer. If we look at the performance of MLP model compared to the Logistic regression model, no significant improvement in performance is observed. This may indicate the general assumption about neural network model outperforming traditional machine learning approach may not apply to a setting such as this where the training data is of limited size. However, our assumption about spatial correlation playing a part in successfully identifying high risk areas is vindicated by a significant change in the accuracy score which is over 70 percent. Furthermore, our proposed model with feature selection pipeline is the best performing model even though the improvement is not astonishingly significant. More importantly, this means that GC-Explainer provides explainable prediction while not having to compromise on model performance. This is specifically encouraging given its application in community health outcome prediction could be further utilized. In addition, this also showcases the feasibility of self explainable

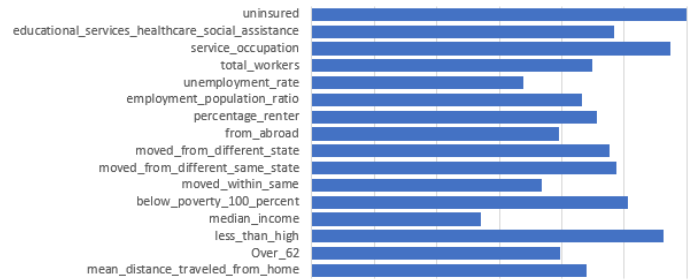


Fig. 3: Feature Importance

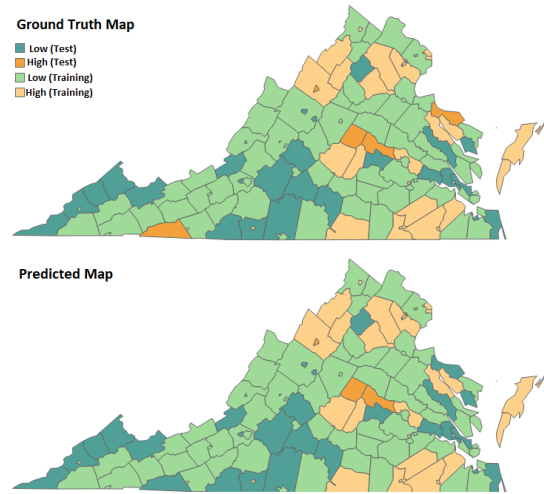


Fig. 4: Actual (top) and predicted (bottom) maps of VA counties indicating the severity of infection. The darker shades indicate testing data while the orange colors indicate high risk areas.

Graph Neural Network for node classification problem where may not always need to rely on post-hoc explanation of trained models.

D. Model Level Explanation

Now, let us take a look at the explanation provided for the trained model using the weight associated with $W_{feature}$ as explained in subsection V-A. What are the most important features? What are the set of features retained by the model where the weights associated to them were not reduced to approximately zero? Interestingly, we are left with the following features that was not affected my the sparsification of the weight matrix which is also displayed Fig 3:

Uninsured Population: This feature indicates the percentage of population medically uninsured. While having no insurance often means that people are less likely to do tests, in case of COVID, the availability of free testings and lack of medical cost may not deter people from getting tested. To the contrary, lack of medical insurance may be indicative of the socioeconomic status of the individual or the nature of their job which may play significant role in deciding vulnerability of a population.

Occupation in healthcare and service industry: This is an expected but significant observation. There has been observations of significant correlation between susceptibility to covid and any job requiring physical proximity which makes the community of healthcare workers and service employees more at risk during the early outbreak.

Unemployment rate of the population: Intuitively it makes a lot of sense given socioeconomic status associated with such population often leads to the inability to take all the preventive measures to avoid getting infected.

Percentages of population over the age of 62: It has already been known that older population are more vulnerable towards this disease than the comparatively younger one which makes sense for the early onset of COVID as observed in our findings.

Percentage of population without high school diploma: Education often plays an important role in deciding the nature of the job or overall financial strength indicating higher socioeconomic status. This could explain how this could be negatively correlated to the spread of the virus. **Population below poverty level:** This may once again indicate the socioeconomic status of a region and the nature of the jobs or lack there of, resulting in an inability to work from home or maintain social distancing.

Percentage of the population renting : one of the interesting observation is how this feature is deemed to be more important than other socioeconomic indicators like mean income or median household income. One possible explanation is that the living situation can often contribute to a population’s ability to adhere to COVID guidelines as a lot of families and professionals in urban neighborhood have to live in crammed up densely populated apartments to afford high living cost.

mean distance traveled from home: Another important finding is how the mobility characteristics of certain regions can affect the spread of the virus as in our experiment this feature is given more weight than many other important features like ”population over 62” or ”total employed worker”. This could be a unique observation to a disease like COVID which is airborne and relies on avoiding human contact to stop the spread. Hence, the preexisting mobility characteristics during the early onset of the disease plays an important role in its eventual severity.

All these observations are particularly useful to understand the importance of giving interpretation about the prediction that can be understandable to authority for them to implement preemptive measure at the earliest.

E. Ablation Study

In this section, we experiment with different parts of the architecture of the proposed model. In one of the experimental settings the classification result is observed while slowly increasing the percentage of node being trained. The result of this is depicted in Table II GC(α) is used for GC-Explainer model with α percent of the nodes as training set. For example, GC(80%) is used for the model with 80% of the nodes in training , 10% in validation (which remains fixed), and the

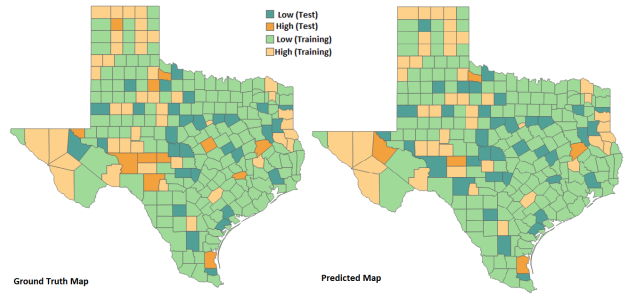


Fig. 5: Actual (left) and predicted (right) maps of Texas counties indicating the severity of infection. The darker shades indicate testing data while the orange colors indicate high risk areas

rest 10% in testing set. As already discussed in experimental results GC(80%) is the model GC-Explainer showed in Table I. As we already know it can achieve more than 70% accuracy.

TABLE II: Experiment: percentage of nodes

Model	Acc
GC(1%)	0.2455
GC(10%)	0.4013
GC(50%)	0.6729
GC(80%)	0.7256

TABLE III: Experiment: adjacency matrix

No of Edges	Acc
6365	0.6521
5527	0.6407
4219	0.7019
3229	0.7256

Furthermore, we conduct experiment with different threshold on the inter node distance as formulated in equation 6, each resulting in different number of edges. In Table III, we depict the effect that number of edges has on the performance of the model. From the observation it is clear that reducing the number of edges in a densely connected graph of this nature can result in better performance. One possible explanation in this particular case could be that propagation of node features from nodes that are too far could negatively effect the performance by introducing noise in the prediction.

F. Case Studies:

Here we will talk about two case studies we did on counties in the state of Virginia and Texas. The counties in the lighter shades of green and orange were the low and high risk areas in training dataset while the darker shade indicates the counties in test dataset as explained in the figure.

Virginia:(Fig 4) We can see from the map that our model is resoundingly successful at predicting high risk counties like Floovana and Goochland. We can also see that the neighborhood of these two counties also includes high risk counties like Buckingham and Cumberland which supports our hypothesis to incorporate spatial correlation through neighborhood aggregation. However, in one case for the county of Westmoreland on the eastern border of the state, a misclassification takes place. This can be due to the county being an outlier among its neighbor in terms of the assigned class.

Texas:(Fig 5) For Texas we can see similar results in terms of accuracy of the classification. Especially encouraging observation was the successful classification of counties like Kenedy and Houston who despite being an outlier in their immediate neighborhood the model was able to predict correctly. This suggests that our framework is not overtly dependent on the immediate neighbors of the county but also considers the static characteristics of the county and how these features also inform decision making.

VII. CONCLUSION:

In this paper we propose a self explainable novel GCN framework GC-Explainer for predicting high risk areas of COVID infection based solely on static population level features in the absence of historical pandemic data which is useful especially during the early part of the pandemic. We classify counties of United States as either high or low based on the predicted severity. Our model shows how incorporating spatial dependency through node level feature propagation largely benefits the performance of the model. Furthermore, through our proposed feature selection process we manage to penalize weights connected to negatively affecting features, only selecting the most contributing one which makes the prediction interpretable without the need of an expensive post-hoc method. The insights we get from the explanation of GC-Explainer can potentially help us in identifying high risk areas in advance and will be helpful towards avoiding or minimizing any potential damage. The problem that we posed, can also be tackled in future for different diseases that can hopefully spawn more easily understandable prediction of community health outcome.

VIII. ACKNOWLEDGEMENT

This research is supported in part by National Science Foundation grants CNS-2141095.

REFERENCES

- [1] Bijaya Adhikari, Xinfeng Xu, Naren Ramakrishnan, and B Aditya Prakash. EpiDeep: Exploiting embeddings for epidemic forecasting. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 577–586, 2019.
- [2] Marco Ajelli, Bruno Gonçalves, Duygu Balcan, Vittoria Colizza, Hao Hu, José J Ramasco, Stefano Merler, and Alessandro Vespignani. Comparing large-scale computational approaches to epidemic modeling: agent-based versus structured metapopulation models. *BMC infectious diseases*, 10(1):1–13, 2010.
- [3] Abdulaziz Alhamadani, Shailik Sarkar, Lei Zhang, Lulwah Alkulaib, and Chang-Tien Lu. Forecasting high-risk areas of covid-19 infection through socioeconomic and static spatial analysis. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 4313–4322. IEEE, 2021.
- [4] Sercan Arik, Chun-Liang Li, Jinsung Yoon, Rajarishi Sinha, Arkady Epshteyn, Long Le, Vikas Menon, Shashank Singh, Leyou Zhang, Martin Nikolchev, et al. Interpretable sequence learning for covid-19 forecasting. *Advances in Neural Information Processing Systems*, 33:18807–18818, 2020.
- [5] Domenico Benvenuto, Marta Giovanetti, Lazzaro Vassallo, Silvia Angeletti, and Massimo Ciccozzi. Application of the arima model on the covid-2019 epidemic dataset. *Data in brief*, 29:105340, 2020.
- [6] Ian Cooper, Argha Mondal, and Chris G Antonopoulos. A sir model assumption for the spread of covid-19 in different communities. *Chaos, Solitons & Fractals*, 139:110057, 2020.

- [7] Joshua M Epstein, Jon Parker, Derek Cummings, and Ross A Hammond. Coupled contagion dynamics of fear and disease: mathematical and computational explorations. *PLoS One*, 3(12):e3955, 2008.
- [8] Nicholas C Grassly and Christophe Fraser. Mathematical models of infectious disease transmission. *Nature Reviews Microbiology*, 6(6):477–487, 2008.
- [9] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.
- [10] 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)*, pages 1–7, 2020.
- [11] Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network. *Advances in neural information processing systems*, 33:19620–19631, 2020.
- [12] Alexander Rodriguez, Nikhil Muralidhar, Bijaya Adhikari, Anika Tabassum, Naren Ramakrishnan, and B Aditya Prakash. Steering a historical disease forecasting model under a pandemic: Case of flu and covid-19. *arXiv preprint arXiv:2009.11407*, 2020.
- [13] Alexander Rodriguez, Anika Tabassum, Jiaming Cui, Jiajia Xie, Javen Ho, Pulak Agarwal, Bijaya Adhikari, and B Aditya Prakash. Deepcovid: An operational deep learning-driven framework for explainable real-time covid-19 forecasting. *medRxiv*, pages 2020–09, 2021.
- [14] Padmaksha Roy, Shailik Sarkar, Subhodip Biswas, Fanglan Chen, Zhiqian Chen, Naren Ramakrishnan, and Chang-Tien Lu. Deep diffusion-based forecasting of covid-19 by incorporating network-level mobility information. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 168–175, 2021.
- [15] Santanu Roy, Gouri Sankar Bhunia, and Pravat Kumar Shit. Spatial prediction of covid-19 epidemic using arima techniques in india. *Modeling Earth Systems and Environment*, pages 1–7, 2020.
- [16] Michael Sejr Schlichtkrull, Nicola De Cao, and Ivan Titov. Interpreting graph neural networks for {nlp} with differentiable edge masking. In *International Conference on Learning Representations*, 2021.
- [17] Katherine F Smith, Michael Goldberg, Samantha Rosenthal, Lynn Carlson, Jane Chen, Cici Chen, and Sohini Ramachandran. Global rise in human infectious disease outbreaks. *Journal of the Royal Society Interface*, 11(101):20140950, 2014.
- [18] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.
- [19] Jian Wang, Chen Xu, Xifeng Yang, and Jacek M Zurada. A novel pruning algorithm for smoothing feedforward neural networks based on group lasso method. *IEEE transactions on neural networks and learning systems*, 29(5):2012–2024, 2017.
- [20] Jian Wang, Huaqing Zhang, Junze Wang, Yifei Pu, and Nikhil R Pal. Feature selection using a neural network with group lasso regularization and controlled redundancy. *IEEE Transactions on Neural Networks and Learning Systems*, 32(3):1110–1123, 2020.
- [21] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.
- [22] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32, 2019.
- [23] Alimuddin Zumla and David SC Hui. Emerging and reemerging infectious diseases: global overview. *Infectious Disease Clinics*, 33(4):xiii–xix, 2019.