CS245 Project Proposal
COVID-19 Pandemic Prediction with DGNN

Team Ideas: Chenyang Wang, Danfeng Guo, Haochen Yin, Huiling Huang, Panqiu Tang, Yijing Zhou
University of California, Los Angeles
Los Angeles, California

1 INTRODUCTION

COVID-19 (COV19) is caused by a coronavirus called SARS-CoV-2. It is an ongoing pandemic starting in Dec, 2019. COV19 is highly infectious, and recent research suggests that it mainly spreads through the air. Up to now, it has caused over a million deaths. Let alone the fact that it has brought uncountable loss to economy. To prevent COV19 from further spreading, it is crucial to identify potential carriers and isolate them. However, as the epidemic spreads, the demand for COV19 tests outgrows the availability. Given the limited supply of test reagents, only a fraction of potential cases can be tested. Hence, it becomes essential to select the maximum probable epidemic carriers. Numerous research has been conducted on pandemic prediction models. Specifically, attempts have been made to evaluate the risk level of specific test groups. The purpose of our project is to build a pandemic prediction model based on Graphical Neural Network (GNN). The model takes static and dynamic features of locations and location interactions as inputs, and predicts the risk factor of each location as outputs. Our model is based on a GNN model[8]. The reason to use this model is that it incorporates the interaction among nodes by using a weighted matrix to represent the population flow among locations.

2 RELATED WORKS

In general, there is a typical model called "SIR". which is a compartmental model in epidemiology. The model consists of three compartments: S for the number of susceptible, I for the number of infectious, and R for the number of recovered or deceased (or immune) individuals, where we use the parameters β, σ, and γ to describe the different transition rates. The dynamics of an epidemic, for example, the flu, are often much faster than the dynamics of birth and death, therefore, birth and death are often omitted in simple compartmental models. However, how to select these transfer parameters is a critical problem in establishing the model. These parameters are not selected and set by us, but are automatically generated by the DGNN-based model, which will also be the main goal of the project.

Studies on applying GNNs on the applications such as Social Influence Prediction and Disease Spread Prediction have been already conducted. For example, Qiu et al.(2018) proposed a Deep Learning framework that utilize GNN techniques to learn node's latent feature representation for predicting social influence[9]. And Trivedi et al. (2019) proposed a learning framework specifically designed for dynamic graphs that captures the underlying dynamics of nodes interaction, predicts dynamic links and time related tasks. In addition, this framework can be generalized to unseen nodes[11]. These models and frameworks provide us some general ideas and techniques for node's representation learning as well as learning the dynamics of the graph. In terms of the specific application of Disease Spread Prediction, Deng et al. (2019) proposed a GNN framework for epidemic forecasting called ColaGNN, that first learns latent representation for each node representing each location using RNN, then use these learnt representations to derive an attention matrix that captures the influence and dynamics between locations. However, this model has only been tested on influenza-like illness (ILI) in US and Japan, and has not yet been evaluated for this COV19 pandemic[1]. Another model called STAN was proposed by Gao et al. (2020). This model, using multiple outputs from the neural network, can predicts the parameters of an epidemic model as well as infected and recovered cases[4]. Although these models have been evaluated and seems powerful, there is a drawback that many detailed data are required to train these models, such as infected, recovered and death cases, demographics data. In many cases, such as in poor and less developed countries, these data are not available instantly, thus the model we based on takes these factors into account and trying to bypass this problem[8].

3 METHOD

3.1 Graph Formulation

Each larger region (a state or country) will be represented as a series of graphs. Smaller regions such as counties are represented as nodes in these series of graphs. Each node is a location within the region with static and dynamic features represented in a feature matrix. Spatial information (mobility and proximity) is represented by edges between these nodes in a graph. Then to represent temporal information, different layers of graphs are joined by edges between nodes that represent the same location but different timestamps. For example, there will be a temporal edge between the node (Los Angeles at March 8) and the node (Los Angeles at March 9). The dynamic attributes in the graph is captured in a feature matrix along with static attributes, and the attributes in each step are updated.

3.2 Prediction Model

Given a series of Graphs at different timestamps, we will update the node representations in each layer based on its adjacency information and recent past temporal information. The model uses a family of GNNs known as the message passing neural networks (MPNN) [5]. They contain neighborhood aggregation layers that can update each layer of graphs based on the normalized weight adjacency matrices following Kipf and Welling [7]. When computing node features, we want to incorporate temporal information by summing nodes in the spatial range in a sliding window of specified time range. This model is applied to all layers in a graph that differ

by time, with unique adjacency matrices and node representation matrices for layers, and shared weights between layers. The model captures more global information as the number of neighborhood aggregation layers increases. However, there should be a balance between maintaining local and global information. The node representation matrices are concatenated to create skip connections between each layer and the output layer. This will allow the network to encode multi-scale structural information. Then the output of the network is followed by a ReLU.

To utilize temporal information, we could also consider combining Long-Short Term Memory network into MPNN. Sequence of graphs representing different time stamps are processed by MPNN, and the resulting sequences of node representation matrices are utilized by two layers of Long-Short Term Memory network (LSTM) [6]. This addition of features can then capture long term temporal information in hidden states.

To improve efficiency of learning in different regions, MPNN could also be extended with the Model-Agnostic Meta-Learning (MAML) algorithm [2]. MAML could capture patterns in a regional outbreak from the start and transfer it to other regions. Different waves of COVID-19 may share information, however this also means the model has fewer training samples and should start predicting from as early as the 15th day. The weight matrices and bias learned from all layers in the MPNN models can be used as datasets and extract feature parameters for initializing states in other regions. The MAML model randomly initializes parameters and uses gradient descent to locally minimize the loss based on different prediction tasks.

From the predicted number of cases, the output of the network, we can combine other information such as population and mobility to define a risk factor for the corresponding location.

4 EXPERIMENT

In our model, we will use datasets from CDC Pandemic Tracker and SafeGraph Mobility as our primary inputs.

The dataset from CDC Pandemic Tracker is an aggregation of daily numbers confirmed and probable case and deaths reported to CDC by different states or territories in U.S. over time[3]. Table 1 demonstrates the schema of this dataset.

The datasets from Safe Graph mobility is richer and more complex in content. We are mainly using 3 types of datasets from SafeGraph. First, the patterns data from SafeGraph details visitors and demographic aggregations available for around 4.2 MM points of interests (POIs). The schema of patterns data is too long to list here [10]. However, the patterns dataset provides some features that we believe will be play an important role in our feature set. In particular, features such as **visits_by_day**, **raw_visitor_counts**, and **popularity_by_day** should be able to provide us a precise and accurate way to quantify the spatial information such as mobility and proximity. Furthermore, we have also considered to use Panel Overview Data from SafeGraph. The Panel Overview data provides home location distributions by states or census block groups, which will be useful as increases in COV19 cases are often community outbreaks. Finally, we have added Normalization Stats dataset into our input selection. The Normalization Stats dataset primarily aggregates the **total_visits** in a particular region. This dataset should serve as our primary source of mobility and proximity approximations, while we can gradually select the features from pattern dataset to build a final feature set.

We have noticed that there is some granularity mismatch in the datasets. For instance, the patterns data from SafeGraph is at a finer granularity, as it uses POIs instead of state / territory groupings. Moreover, the data from the Panel Overview dataset sometimes uses census block groupings as well. Thus, we intend to run the model on a smaller dataset initially, limiting the number of states around one to three and fine-tune the hyper parameters first. In the final version of our project, we will expand the scale of our model and set all the states in the US as graph nodes. The prediction outcome will be the total number of cases within each state.

4.1 Schedule and Expected Result

Given the huge amount of data, pre-processing and feature selection are expected to take the most time in this project. We are planning to take at most two weeks on that. Then we will use the following one week for model training. And the remaining time for miscellaneous works. All works will be distributed amount members as evenly as possible.

REFERENCES

[1] Songgaojun Deng, Shusen Wang, Huzefa Rangwala, Lijing Wang, and Yue Ning. 2019. Graph message passing with cross-location attentions for long-term ILI prediction.
[2] P.; Finn, C.; Abbeel and S. Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks.
[3] Centers for Disease Control and Prevention. 2020. United States COVID-19 Cases and Deaths by State over Time.
[4] Junyi Gao, Rakshith Sharma, Cheng Qian, Lucas M. Glass, Jeffrey Spaeder, Justin Romberg, Jimeng Sun, and Cao Xiao. 2020. STAN: Spatio-Temporal Attention Network for Pandemic Prediction Using Real World Evidence.
[5] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. 2017. Neural message passing for Quantum chemistry.
[6] A.; Graves and N. Jaitly. 2014. Towards End-to-End Speech Recognition with Recurrent Neural Networks.
[7] Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks.
[8] George Panagopoulos, Giannis Nikolentzos, and Michalis Vazirgiannis. 2020. United We Stand: Transfer Graph Neural Networks for Pandemic Forecasting.
[9] Jiezhong Qiu, Jian Tang, Hao Ma, Yuxiao Dong, Kuansan Wang, and Jie Tang. 2018. DeepInf: Social Influence Prediction with Deep Learning.
[10] SafeGraph. 2020. Places Schema.
[11] Rakshit Trivedi, Mehrdad Farajtabar, Prasenjeet Biswal, and Hongyuan Zha. 2019. DyRep: Learning Representations over Dynamic Graphs.

Table 1: CDC dataset schema[3]

Column Name	Description	Type
submission_date	Date of counts	Date and Time
state	Jurisdiction	Plain Text
tot_cases	Total number of cases	Number
conf_cases	Total confirmed cases	Number
prob_cases	Total probable cases	Number
new_case	Number of new cases	Number
pnew_case	Number of new probable cases	Number
tot_death	Total number of deaths	Number
conf_death	Total number of confirmed deaths	Number
prob_death	Number of new deaths	Number
pnew_death	Number of new probable deaths	Number
created_at	Date and time record was created	Date and Time
consent_cases	f Agree, then confirmed and probable cases are included. If Not Agree, then only total cases are included.	Plain Text
consent_deaths	If Agree, then confirmed and probable deaths are included. If Not Agree, then only total deaths are included.	Plain Text