ABSTRACT

COVID-19 has caused lasting damage to almost every domain in public health, society, and economy. To monitor the pandemic trend, existing studies rely on the aggregation of traditional statistical models and epidemic spread theory. In other words, historical statistics of COVID-19, as well as the population mobility data, become the essential knowledge for monitoring the pandemic trend. However, these solutions can barely provide precise prediction and satisfactory explanations on the long-term disease surveillance while the ubiquitous social media resources can be the key enabler for solving this problem. For example, serious discussions may occur on social media before and after some breaking events take place. These events, such as marathon and parade, may impact the spread of the virus. To take advantage of the social media data, we propose a novel framework, Social Media enHAnced pandemic suRveillance Technique (SMART), which is composed of two modules: (i) information extraction module to construct heterogeneous knowledge graphs based on the extracted events and relationships among them; (ii) time series prediction module to provide both short-term and long-term forecasts of the confirmed cases and fatality at the state-level in the United States and to discover risk factors for COVID-19 interventions. Extensive experiments show that our method largely outperforms the state-of-the-art baselines by 7.3% and 7.4% in confirmed case/fatality prediction, respectively.

CCS CONCEPTS

• Information systems → Data mining.

KEYWORDS
time series prediction, information extraction, social media mining

1 INTRODUCTION

Over 200 countries and territories have been deeply impacted by the outbreak of the coronavirus disease 2019 (COVID-19). As of 2021 May, a total of 164 million cases and 3.4 million deaths were reported all over the world\(^1\). It is critical to forecast the short-term and long-term trends of the epidemic, to help governments and health organizations determine the prevention strategies and help researchers understand the transmission characteristics of the virus.

Modeling the COVID-19 pandemic is challenging. Previous studies present three types of disease transmission approaches to explain and model the pandemic, which are exponential growth models [52], self-exiting branching process [42], and compartment models (e.g., Susceptible-infected-resistant (SIR) [39], Susceptible-Exposed-Infected-Removed (SEIR) [4] and Herd Immunity [23]). However, exponential growth models can only address the initial outbreak while self-exiting-branching process and compartment models favor the development and peak stages [8]. Besides, the pandemic trend varies dramatically across different locations and times in response to real-time breaking events. To tackle these challenges, some data-driven approaches [3, 19] that ensembles

\(^1\)https://covid19.who.int/
statistical and machine learning models emerge for monitoring the confirmed cases, fatality, and hospitalizations. [25, 60] leverage graph neural networks to incorporate the population mobility data, i.e., how many people traveled from one place to another, to encode the underlying diffusion patterns into the learning process. However, these models take into consideration only a small number of homogeneous features. They are incapable of capturing potential risk factors and identifying various intervention mechanisms of this new pandemic as well.

As the quarantine life takes over the world and people turn to online platforms for communication and information, social media become more influential than ever [27, 58]. The vast collections of social media streams can capture local activities (e.g., public gatherings and vaccination progress) that may affect the transmission of the virus in real-time. Over 170 million tweets are posted every day in the United States related to observations, behaviors, and thoughts of individual users [17]. The social media users can be naturally treated as robust "social sensors" [34] to unveil the surveillance evidence over time and space. For example, in Figure 1, the severe discussions related to the coming social events such as "Marathon" and "Parade" may indicate a potential risk of virus spread while some hot hashtags like "#StayHome" or "#GetVaccine" may represent the safety awareness of individuals in the prevailing areas. Over the past decades, researchers have successfully applied social media data to monitor the earthquakes [68] or air quality [34]. Inspired by these works, we aim to incorporate social media content to forecast the pandemic.

To this end, we want to answer the following interesting research questions:

- Can social media contents further enhance the short-term and long-term COVID-19 forecasts?
- How to identify potential risk factors from the social media data as these factors may vary over time and space?

Motivated by them, we collaborate with Twitter and use their COVID-19 stream API service to crawl large-scale tweets related to COVID-19 based on Twitter’s internal COVID-19 annotations. We propose a novel framework, Social Media enHanced pandemic suveillance Technique (SMART), which is composed of two modules, information extraction module and time series prediction module. Specifically, in the information extraction process, we recognize named entities and identify relationships among them from the large-scale tweet corpora. Based on the entities and relationships, we build a spatial-temporal heterogeneous knowledge graph. We then propose a Dynamic Graph Neural Network (DGNN) with a Bidirectional Recurrent Neural Network (Bi-RNN) to forecast pandemic trends and suggest risk factors for each location.

Our main contributions are summarized as follows:

- To the best of our knowledge, we are the first to simultaneously detect social events for pandemic surveillance and suggest the risk factors.
- We propose a novel framework, SMART, for domain-specific information extraction from social media data and time series prediction on dynamic spatial-temporal graphs. Extensive experiments show the effectiveness of our approach. We achieve 7.3% and 7.4% improvements from the state-of-the-art methods for confirmed case/fatality predictions.

- We will open-source our implementations to facilitate the research community.

2 RELATED WORK

2.1 Pandemic Forecast

Epidemic Prediction Models. There are three types of epidemic prediction models in literature, including exponential growth models [52], self-exiting branching process [42], and compartment models [4, 6, 7, 9, 23, 28, 29, 39, 43, 57, 65, 71]. The dynamics of infectious diseases are expressed by the compartment models for predicting the epidemic trends using ordinary differential equations [65]. SIR [39], as the most prevailing compartment model, segments the population into three parts: Susceptible, Infectious, and Recovered and express the population flow among them with evolving equations. Later, many cumulative studies based on SIR emerge, including SEIR [4], SEIS [77], MSEIR [29], SuEIR [83], and MSIR [57]. In specific, SEIR includes the Exposed compartment and SEIS, MSEIR, SuEIR extend SEIR by taking into account either immunity or untested/unreported compartments. However, as concluded in [8], the exponential growth models can only address the initial outbreak while self-exiting-branching process and compartment models favor the development and peak stages. None of these models are expected to be precise and robust in the long-term pandemic prediction.

Statistical and Machine Learning Models. Researchers also apply statistical time series prediction models such as ARIMA and PROPHET for COVID-19 pandemic prediction [44, 53]. ARIMA [10] is an Autoregressive Integrated Moving Average model, relying on a basic assumption that the future time series are linear aggregations of the past ones. PROPHET [74] is an additive model that emphasizes seasonal effects so that the model works better on time series with periodic patterns. Chimmula and Zhang [14], Rodriguez et al. [64], Saba and Elsheikh [66] aggregate neural networks to an Autoregressive model, to enhance inter-region connections or temporal dependencies. However, these models conduct pandemic forecasts highly depending on the trend and seasonality instincts behind the historical COVID-19 statistics, incapable of incorporating heterogeneous features. Gao et al. [25], Jin et al. [37], Panagopoulos et al. [60] apply graph neural networks to take advantage of the mobility data across different regions but still cannot detect hidden risk factors for the pandemic modeling. Therefore, in this paper, we propose a social media enhanced pandemic forecast framework to incorporate the extracted entities and relationships for confirmed case/fatality prediction with strong interpretability.

2.2 Prediction with Social Media Data

Plenty of studies have utilized the social media data for various prediction tasks including air pollution monitoring [31, 34, 35, 38, 56], earthquake forecast [68, 79], stock market prediction [36, 59], and disease detection [13, 55, 67]. However, limited work incorporates the social media data to calibrate the COVID-19 pandemic surveillance. Qin et al. [63] employ the search index of Baidu search engine to serve as a pandemic early predictor. Bae et al. [5] leverage the social effect of media information to strengthen the compartment model for pandemic prediction. However, this study solely takes into consideration the social effects of the media to users’ normal
life while our method curate every tweet and detect significant social events to enhance the pandemic prediction.

3 SOCIAL MEDIA ENHANCED PANDEMIC SURVEILLANCE

Given a large-scale collection of social media data together with the historical confirmed cases/fatalities and the population mobility statistics, we aim to forecast the pandemic trend and recognize potential risk factors. The framework of our SMART model consists of two components: (i) information extraction module including a named entity recognizer and a relation identifier (as shown in Figure 2); (ii) spatial-temporal dynamic graph encoder for pandemic trend forecast (as shown in Figure 3).

3.1 Constructing Dynamic Knowledge Graphs from Social Media Data

We propose a bottom-up solution to extract entities and relations to construct the heterogeneous dynamic knowledge graphs.

Named Entity Recognition (NER). NER is a natural language processing (NLP) task which labels the tokens in a sequence with their corresponding labels. In this work, we adopt the NER setting to extract entities of interest from the social media data by labeling the words or phrases in the tweet sentences. As examples in Figure 2, we want to recognize nurse as OCCUPATION, stay home as INDIVIDUAL_BEHAVIOR, race as EVENT, and so on.

Traditional NER approaches heavily rely on expensive and time-consuming feature engineering including parsing the Part-of-Speech tags of each word and the syntactic dependency structures of the sentences. Some recent studies incorporate neural networks with statistical models, such as conditional random fields, to improve the model performance. With deep language models like BERT and RoBERTa, the NER performance can be further improved. Without the loss of generality, we leverage BERT model to provide contextualized embeddings and learn a supervised named entity recognizer. To overcome the problem with the nonexistence of annotated tweets as training data, we collect the benchmark corpora and their annotations for multiple NER tasks, including I2B2-2010, CORD-NER and MACROBAT-2018. Based on those external datasets, we jointly learn a recognition model to extract entities on the COVID-19 related tweets data. On average, we extracted 10,040 unique entities of 45 entity types from 270k tweets corpus every day.

Relation Extraction. Given the extracted entities, the next step is to identify the relationships among the entities. Note that we only extract intra-tweet relations. In other words, we do not predict the relation between entities in different tweets. Existing solutions formulate the problem as a sequence classification task, given a textual sequence and the positions of two named entities. Specifically, a multi-class classification is conducted to assign a label from a desired set for the relationship. However, this formulation highly depends on the quality and quantity of the annotated datasets to achieve satisfactory performance. It is obviously incapable of identifying emerging new relation types.

To overcome the above challenge, we convert the multi-class prediction task to a binary classification problem of only identifying the existence of a potential relationship between any entity pair in each tweet instance. We aggregate datasets from multiple tasks including Wiki80, I2B2-2012, and MACROBAT-2018 to create the positive training data (labeled as ‘True’). In order to achieve balanced training, validation and test datasets, we apply negative sampling to create the same number of instances with the label ‘False’. Note that we assume no relation between any two entities if the entities were not annotated. Similarly, we acquire the sequence representations from the fine-tuned BERT language model and feed them into a binary classification layer for label prediction. During the inference stage, we enumerate all possible pairs of entities in each tweet and assign binary labels for them.

Domain-specific Pre-trained Language Model. To tackle domain-specific tasks, such as Clinical information extraction and Bioinformatics knowledge acquisition, recent studies pre-train new language models with large-scale corpora collected from those domains to learn customized token and sequence representations. Motivated by these approaches, we leverage all COVID-19 relevant text corpora together with the social media data to pre-train a CoronaBERT language model with 12 layers of Transformers and over 110 million parameters, in order to equip our models with powerful input embeddings. We ceaselessly fine-tune the parameters in CoronaBERT as more COVID-19 stream corpora become available and release the models on a quarterly basis.

Heterogeneous Knowledge Graph Aggregation. After named entity recognition and relation extraction, we apply the DBSCAN clustering model to merge semantically similar entities for reducing the noises in the entity sets. This step is essential for cleaning the entities extracted from tweets. For example, “Marathon” and “Marathon;)” are supposed to be merged and “COVID-19” is indeed the same as “COVID2019”. In specific, we cluster the entities based on the similarity among their entity embeddings acquired by CoronaBERT. We assign the node in each cluster with the highest occurrence in tweets as the cluster head. Other nodes in the same cluster will be replaced by the cluster head.

Based on the clustering results, we aggregate the denoised knowledge pieces into a heterogeneous knowledge graph. Two types of nodes exist in the graph, including location nodes and entity nodes. Here we set the location nodes as the 50 states in the United States while our methods can be easily extended to the county-level locations or applied to other countries and regions. Next, we build three types of edges as follows:

- **Entity-Entity** edges: we add an edge between any two entities if there is a ‘True’ relationship identified.
- **Location-Entity** edges: we look up the geo-location attribute of the tweet where each entity is extracted and add an edge between the entity node and the geo-location.
- **Location-Location** edges: we add an edge between a location pair under two circumstances, (i) two locations are adjacent to each other on the US map, (ii) we detected population transition from one location to another according to the mobility data. More details of the mobility data are provided in Section 4.1.

We build one knowledge graph for each day. Later, knowledge graphs within a certain time period will be further aggregated for time series prediction, as described in Section 3.2.
3.2 Time Series Prediction with Dynamic Graph Attention Network

**Dynamic graph aggregation.** We represent the heterogeneous knowledge graph of the $t$-th day as $G^{(t)} = (V^{(t)}, E^{(t)})$ where $n = |V^{(t)}|$ denotes the number of nodes, $V^{(1)} = V_L^{(1)} \cup V_E^{(1)}$, where $V_L^{(1)}$ is the location node set and $V_E^{(1)}$ is the entity node set. Given a sequence of knowledge graphs $\{G^{(1)}, G^{(2)}, \ldots, G^{(T)}\}$ of length $T$, we aim to predict the COVID-19 courses including confirmed cases and fatality cases on the day $T + l$. We regard it as a short-term prediction when $l < 14$ or a long-term prediction when $l \geq 14$. We formulate the time series prediction problem as a regression task.

We continue to aggregate the length-$T$ graph sequence into one spatial-temporal graph $G^{S} = (V^{S}, E^{S})$ as shown in Figure 3. First, we keep all the location nodes from different times in the period, i.e. $V^{S}_L = V_L^{(1)} \cup V_L^{(2)} \cup \ldots \cup V_L^{(T)}$. On the other hand, we merge entity nodes of different times, i.e. $V^{S}_E = V_E^{(1)} \cup V_E^{(2)} \cup \ldots \cup V_E^{(T)}$, where $\cup$ denotes a time-unaware set union. For example, the entity node $e_1$ is recognized in the location $s_1$ on both time 1 and time 2, but we only keep one $e_1$ in $V^{S}_E$ by connecting $e_1$ to $s_1^{(1)}$ and $s_1^{(2)}$. In this way, we introduce the inter-time propagation edges to expand the node neighbors along the temporal dimension so that we can easily model the structural temporal dependencies among the nodes.

**Node Features.** Our pre-trained CoronaBERT is applied to generate the initial semantic features $x_i^{SE}$ of dimension $d_e$ for node $i$ of any type. We also incorporate the historical COVID-19 statistics $x_i^{ST}$ of $d_t$ days ahead of the current time as an extra feature set for location nodes, resulting in a node feature embedding $x_i = x_i^{SE} || x_i^{ST}$ of dimension $d_e + d_t$, where $||$ denotes a vector concatenation. Note that we keep the embedding dimensions of location nodes and entity nodes the same, in order to smooth the graph propagation computation. Hence, we append a zero vector of length $d_t$ at the end of each entity vector.

**Dynamic Graph Neural Network.** We propose a multi-head DGNN architecture to perform the graph propagation. We first conduct a linear transformation on the input node embeddings:

$$z_i = W_p x_i,$$

where $W_p$ is a learnable weight matrix; $p = \{1, \ldots, H\}$; $H$ is the number of heads. Then, we compute a pair-wise un-normalized attention score of an edge between any two neighbors (two nodes $i$ and $j$) in the graph:

$$e_{ij,p} = \text{LeakyReLU}(W_p^T (z_i || z_j)).$$

where $W_p$ is a learnable weight vector and LeakyReLU [80] is applied as a non-linear transformation. We use the attention score to indicate the importance of a neighbor node in the message passing process, especially when we interpret the risk entities to each location. A Softmax is applied to normalize the attention weights to a probability distribution so that we can easily interpret and compare...
the importance of all incoming edges,
\[ a_{ij,p} = \frac{\exp(e_{ij,p})}{\sum_{k \in N_E(i) \cup N_a(i)} \exp(e_{ik,p})}, \]
where \( N_E(\cdot) \) and \( N_a(\cdot) \) denote the sets of neighboring location nodes and entity nodes. We finally aggregate the embeddings of neighboring nodes. The aggregation is scaled by the normalized attention scores. We compute the averaged embeddings over different heads,
\[ x_i' = \sigma \left( \frac{1}{H} \sum_{p=1}^{H} \sum_{j \in N(i)} a_{ij,p} x_{ij} \right). \]

**Attentive Bi-Recurrent Neural Network.** We intend to further encode the temporal dependencies between the location nodes over times and learn a hidden state of the overall graph using an Attentive Bi-RNN module. We collect embeddings from the same location of different times \( x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(T)} \) and recursively feed them into a Bi-RNN with Gated Recurrent Units (GRU) [15]. We choose GRU instead of Long Short Term Memory (LSTM) [30] unit due to its computational efficiency and capability of tackling shorter sequences like tweets [16]. The hidden representation of each location in time \( t \) is learned from two directions,
\[ h_i^{(t)} = \text{GRU}(h_i^{(t+1)}, x_i^{(t)}), h_i^{(t)} = \text{GRU}(h_i^{(t-1)}, x_i^{(t)}), \]
\[ h_i^{(t)} = \overrightarrow{h_i^{(t)}} \oplus \overleftarrow{h_i^{(t)}}, \]
We then aggregate the hidden states with another attention mechanism,
\[ \psi_i = \sum_{t=1}^{T} \frac{\exp(u^T h_i^{(t)})}{\sum_{k} \exp(u^T h_i^{(k)})}, \]
where \( u \) denotes a context vector and \( \beta_i^{(t)} \) are attention scores reflecting the contribution of the hidden representation in time \( t \).

**Learning Objective.** We feed the context-aware node representation \( \psi_i \) into two layers of Feed Forward Networks (FFN) and lastly generate a scalar \( \hat{y}_i^{(t+i)} \) representing the predicted COVID-19 confirmed case or fatality number in \( f \) days ahead of time \( t \). We compute the loss with the following Mean-Squared-Error (MSE) objective [70],
\[ \mathcal{L} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{t=1}^{n} (\hat{y}_i^{(t+i)} - y_i^{(t+i)})^2, \]
where \( m \) is the number of location nodes and \( n \) is the number of days that requires a prediction.

## 4 EXPERIMENTS

### 4.1 Datasets

**Twitter Stream Data.** We collaborate with Twitter and build a real-time tweet crawler to steadily acquire relevant social media tweets using their COVID-19 streaming API\(^2\) [54]. In detail, the streaming API returns real-time tweets related to COVID-19 based on Twitter’s internal COVID-19 tweet annotation system. The data collected for this paper start on May 15, 2020 and end on April 8, 2021. Figure 4 compares the distributions of the US population and the number of tweets over 20 states. We notice except that

\(^2\)https://developer.twitter.com/en/docs/labs/covid19-stream/api-reference/.

New York people are more passionate about posting COVID-19 related tweets while California people do the opposite, other states have relatively similar spatial distributions over the population and number of tweets.

**Mobility Data.** As Panagopoulos et al. [60] conclude a strong relationship between the population transition and regional COVID-19...
trends, we also collect the mobility data that describe the population transition in the United States from SafeGraph\(^4\) for pandemic forecast. As shown in figure 5, we illustrate a mobility data sample which includes the population transition among five states on 01-01-2021. The majority transitions are in-state transitions.

**COVID-19 Statistics.** We leverage the US state-level COVID-19 statistics gathered by the New York Times\(^4\) based on reports from state and local health agencies for building the ground truths of pandemic forecasts. We use the statistics of confirmed new cases and fatalities from May 5, 2020 to April 8, 2021. Note that the start date is the earliest date when we have Twitter Stream data available. The average new confirmed cases and fatalities over 50 states are 1788.3 and 28.7 per day while the standard deviations are 3374.8 and 63.5. California has the highest average number of new confirmed cases (10988.5) and fatalities (173.4). Vermont has the lowest numbers (60.0 new confirmed cases and 0.5 fatalities).

### 4.2 Experimental Setup and Evaluation Metrics

Following the experimental setup in \([60]\), we train a model with the data from time 1 to time \(t\) and use it to predict the numbers on time \(t+1\). We evaluate the model on short-term (\(l = \{1, 7\}\)) and long-term (\(l = \{14, 28\}\)) predictions. Note that we learn a different model to predict the cases for time \(t+l\) and \(t+l_j\), where \(i \neq j\). In the training process, we select 5 data points from the training set as the validation set to identify the best model.

We evaluate the performance of our method by computing the Mean-Absolute-Error (MAE) [69],

\[
errormae = \frac{1}{mn} \sum_{t=1}^{n} \sum_{i=1}^{m} |y_{i}(t+l) - \hat{y}_{i}(t+l)|,
\]

where \(m\) and \(n\) denote the numbers of test instances and location nodes. We also follow \([1, 41]\) to compute the symmetric Mean-Absolute-Percentage-Error (sMAPE) to show the average error rate over times and locations,

\[
errorsmape = \frac{1}{mn} \sum_{t=1}^{n} \sum_{i=1}^{m} \left| \frac{y_{i}(t+l) - \hat{y}_{i}(t+l)}{y_{i}(t+l)} + \frac{\hat{y}_{i}(t+l) - y_{i}(t+l)}{\hat{y}_{i}(t+l)} \right|.
\]

### 4.3 Baselines

We select three types of baselines and benchmark models to compare to our approach.

**Compartment models.** As there are a large number of compartment models proposed in recent days for COVID-19 forecast, we select three of them with the top performance and complete results in the desired time period from the COVID-19 Forecast Hub\(^6\): JHU_IDD-CovidSP \([48]\), UCLA-SuEIR \([83]\), and RobertWalraven-ESG \([76]\). In detail, JHU_IDD-CovidSP proposes a modified SEIR compartment model where the time in the Infected compartment follows an Erlang distribution to produce more realistic infectious periods. RobertWalraven-ESG is a mathematical model that approximates the SEIR method with a particular skewed Gaussian

| Hyperparameter                          | Value |
|-----------------------------------------|-------|
| Learning Rate                          | 0.001 |
| Batch Size                              | 4     |
| Dropout Rate                            | 0.5   |
| Bi-RNN Hidden State Size                | 64    |
| DGNN Hidden Unit Size                   | 64    |
| Graph Sequence Length T                 | 7     |
| Semantic Feature Dim. \(d_e\)           | 768   |
| Historical COVID-19 Statistics Feature Dim. \(d_t\) | 7     |

\(^1\)https://www.safegraph.com/
\(^2\)https://github.com/reichlab/covid19-forecast-hub
\(^3\)For example, if we predict the next-day (i.e., \(l = 1\)) case number for date 12-31-2020, we make use of all the data between 5-15-2020 and 12-31-2020 to build the training set.
\(^4\)The model descriptions and up-to-date predicted results can be found at https://github.com/reichlab/covid19-forecast-hub

\(^5\)https://www.nytimes.com/covid-19-data.
\(^6\)https://github.com/github.com/facebook/prophet.

Statistical time series prediction models. Two commonly used statistical models are compared to our approach: ARIMA and PROPHET. ARIMA [44] is an autoregressive moving average model, explaining a given time series based on its past values. PROPHET [53] is a time series prediction model\(^7\) where non-linear trends can be fit with seasonality, plus holiday effects.

**Neural network-based models.** A simple two-layer LSTM-based neural network (LSTM) is used for COVID-19 pandemic prediction [14], taking the sequence of case numbers from the previous week as the input. MPNN [60] is a message passing neural network, building graphs to aggregate the historical case numbers from the neighboring locations based on the mobility magnitude. MPNN+LSTM [60] takes advantage of both MPNN and LSTM by jointly learning the graph propagation and temporal dependencies over case numbers of different times.

### 4.4 Implementation Details

**Information Extraction.** We train the named entity recognition and relation extraction models both for a maximum of 10 epochs. The models are implemented in PyTorch and we use Adam optimizer [40] to optimize the model parameters. We randomly select 10% instances from the training set as the validation set to select the optimal models. To avoid the GPU out-of-memory problem, we filter out tweets with more than 40 words (around 0.17%). In this work, we focus on the information extraction from English tweets so we also remove the tweets if 90% of the contents are non-English.

**Time Series Prediction.** We train the model for a maximum of 300 epochs. Early stopping occurs after 100 epochs. Similarly, we utilize PyTorch to implement the model and leverage Adam [40] for parameter optimization. Batch normalization [33] and dropout [72] are applied to the outputs of DGNN and FFN layers to avoid overfitting. It takes around 8 hours to finish the complete training and evaluation cycle with one NVIDIA V100 GPU. We employ grid search to find the optimal hyperparameters of our model. Detailed hyperparameter values are listed in Table 1.

\(^7\)https://github.com/facebook/prophet.

Table 1: Grid-search is used to find the optimal hyperparameters of our model.
Table 2: Performance of the short-term (1 day & 7 days ahead) and long-term (14 days & 28 days ahead) new confirmed case number forecast. All the improvements of SMART over the baseline methods are statistically significant at a 99% confidence level in paired t-tests. SMART achieves 5.6%, 9.5%, 9.4%, and 5.6% lower MAE than the best baseline MPNN+LSTM when forecasting the new confirmed case numbers for 1, 7, 14, 28 days ahead.

Table 3: Performance of the short-term (1 day & 7 days ahead) and long-term (14 days & 28 days ahead) new fatality number forecast. All the improvements of SMART over the baseline methods are statistically significant at a 99% confidence level in paired t-tests. SMART achieves 3.2%, 7.3%, 9.1%, and 8.7% lower MAE than the best baseline MPNN+LSTM when forecasting the fatality for 1, 7, 14, 28 days ahead.
4.5 Results

Confirmed Case Forecast. Results of the confirmed case short-term and long-term forecasts are shown in Table 2. Compared to the best baseline method MPNN+LSTM, our model improves the average MAE and sMAPE by 7.3% and 2.3%, respectively. The results show SMART significantly outperforms the compartment models, such as JHU_10D-Covid15P and UCLA-SuEIR. We think the big gap between our method and the compartment models results from the serious over-fitting issue in the SEIR model and its extensions. The SEIR model tends to assume that the peak would come right after the current data and is especially weak at predicting the progression at the early pandemic stage [25]. We also notice that the two statistical time series prediction models perform differently, and ARIMA gets much lower errors than PROPHET especially in the long-term prediction. This could be because PROPHET is supposed to work best with time series that have strong seasonal effects which is obviously not the situation in the COVID historical statistics. It turns out that a simple linear aggregation over the past case numbers can achieve relatively good performance. Besides, MPNN gets higher errors compared to its temporal variant, MPNN+LSTM, denoting the effectiveness of learning the temporal dependencies together with the graph aggregation. However, solely using LSTM to conduct the pandemic forecast achieves quite inaccurate predictions. We think it is because sequence modeling approaches like LSTM are unstable to handle the sequential inputs with sharply changing patterns [60]. For instance, it may be hard for LSTM to recognize turning points, such as lockdowns and reopen. SMART initially outperforms other models by a small margin (1-day-ahead forecast) while the improvement increases as the model predicts on later days. Compared to MPNN+LSTM, SMART achieves the largest error reduction of 9.5% and 9.4% while forecasting the case numbers in the next 7th and 14th day. This could be because the ongoing events discussed on social media would not immediately affect the COVID-19 confirmed case numbers. More precisely, we need 1-2 weeks on average for the newly infected cases to be self-identified, tested and confirmed, based on our observations.

To observe the detailed forecast performance on every test instance, we plot the smoothed MAE curve for SMART and three neural network-based baselines (LSTM, MPNN, MPNN+LSTM). Note that every data point on the curves represents the MAE over all the test instances before that date. Each data point on the curves represents the MAE over all the test instances before that date. All the curves quickly converge as the models obtain enough training instances, denoting the stability of our method.

Fatality Forecast. We show the results of fatality forecasts in Table 3. SMART achieves 7.4% and 5.5% lower MAE and sMAPE, compared to the best baseline model MPNN+LSTM. Among the three compartment models, UCLA-SuEIR performs the best. We surmise that taking unreported/untested cases leads to better modeling on fatalities. We notice the MAE of LSTM model is lower than SMART by 2.9% while its sMAPE is higher than SMART by 26.1%. We believe the LSTM model has been over-fitted to some extremely large or small values so that a large MAE can be avoided but the sMAPE will explode. Again, we find that the improvements of SMART on the 7, 14-28-day-ahead forecast tasks (7.3%, 9.1%, and 8.7%) are much more significant than the 1-day-ahead forecast task (3.2%), demonstrating the long-term advantages of our method.

4.6 Ablation Study

We present the ablation study on the 7-day-ahead new confirmed case forecast task to demonstrate the effectiveness of each module in our framework. We observe similar results on other forecast tasks. Here we explain the different settings of our model variants as follows.

w/o RE module. Under this setting, we exclude the Entity-Entity edges in the heterogeneous knowledge graphs so that we can observe the improvement from our relation extraction module.

w/o NER module. We continue to exclude the Location-Entity edges to check the contribution of our named entity recognition module. Under this setting, all the edge propagation between location nodes and entity nodes are eliminated.

w/o Attentive Bi-RNN module. We remove the Attentive Bi-RNN module from our framework. We alternatively compute an element-wise averaged representation for each location node and feed it into the FNN layer for the pandemic forecast.

w/o DGNN module. To verify the contribution of our DGNN module, we replace the DGNN module with a BERT language model (BERT-BASE) to initialize the semantic representations for each node.

w/o CoronaBERT Language Model. We also observe the contribution from our pre-trained CoronaBERT language model by replacing it with a BERT language model to initialize the semantic representations for each node.

In summary, every component in our framework is proved effective. Removing Entity-Entity and Location-Entity edges leads to 4.3% and 8.9% error lifts, respectively. When we jump over the DGNN module, the error dramatically increases, proving the capability of the heterogeneous graph to encode a rich spatial-temporal
representation for each location node. The Attentive Bi-RNN module also makes a significant improvement of 10.4% on the forecast performance.

4.7 Risk Factor Discovery
To identify the potential location-wise risk factors of the COVID-19 pandemic, we make use of the normalized attention score $a_{i,j}$ (introduced in Section 3.2) which indicates the contribution of each entity node $i$ when node $i$’s message is passed to the location node $j$. For each location, we first rank all the dates based on the number of confirmed cases in decreasing order. We then pick the top 20% dates with the biggest numbers from all the dates to build a high set. Ultimately, we aim at discovering a group of significant entities from the tweets that are used to predict the confirmed cases on the dates from the high set. Specifically, during each inference process, we retrieve the attention scores of all the Location-Entity edges for each location node. The Attentive Bi-RNN module also makes a significant improvement of 10.4% on the forecast performance.

| Model                        | MAE    | Error Lift(%) |
|------------------------------|--------|---------------|
| SMART                        | 474.164| -             |
| w/o RE module                | 495.688| +4.3          |
| w/o NER module               | 518.389| +8.9          |
| w/o Attentive Bi-RNN module  | 528.025| +10.4         |
| w/o DGNN module              | 1112.334| +120.8       |
| w/CoronaBERT Language Model  | 500.878| +5.6          |

Table 4: Ablation study on the 7-day-ahead forecast task. Similar results can be achieved from other forecast tasks. We can observe significant improvements from all components in our framework.

| California | New York | Florida |
|------------|----------|---------|
| #1         | pharmacists | traveler | workers |
| #2         | #endthelockdown | doctors | #stopcovidcorruption |
| #3         | mexico city  | test results | crimes |
| #4         | covid-positive | bill gates | voting |
| #5         | msm          | public health | #stayconnected |

| Ohio       | Hawaii    | Vermont  |
|------------|-----------|----------|
| #1         | golf      | mental health | #endthelockdown |
| #2         | #hydroxychloroquine | immigrants | rape |
| #3         | #wg1wgaworldwide | surf | #wakeupamerica |
| #4         | crush     | 2ndwave   | burger |
| #5         | traveler  | patients  | sickness |

Table 5: Top-5 risk factors in six different states related to COVID-19 pandemic.

| HASHTAG          | SIGN_OR_SYMPTOM |
|------------------|-----------------|
| #1               | genocides      |
| #2               | loyalty        |
| #3               | discord        |
| #4               | voting         |
| #5               | racism         |

| SOCIAL_INDIVIDUAL_BEHAVIOR | ORGANIZATION |
|----------------------------|--------------|
| #6                         | youtube      |
| #7                         | @nytimes     |
| #8                         | nih          |
| #9                         | amazon       |
| #10                        | msm          |

Table 6: Top-5 risk factors under four different entity categories related to COVID-19 pandemic.

of the virus. When people pay more attention to the local doctor resource or public health condition in New York, the peak of the pandemic should not be far away. However, it may be hard to interpret some entities like msm without the contexts since msm can be the abbreviation of either mainstream media or master of science in management.

We also incorporate the named entity recognition results to show in Table 6 the top-5 risk factors under 4 different categories: HASHTAG, SIGN.OR_SYMPTOM, SOCIAL_INDIVIDUAL_BEHAVIOR and ORGANIZATION. We notice msm is categorized as an organization, so it is more likely to be interpreted as the mainstream media. It is obvious that the pandemic is getting more serious if we are facing the personal protective equipment shortage. The government and health institutes are better to be prepared if more and more people become sick and have the symptoms such as cough and sneezes. There are limitations if we only rely on the entities with high attention scores to interpret the risk factors. For example, we cannot simply conclude that the prevailing entity amazon results in an increasing trend of the pandemic. The relationship between amazon and increasing trend might not be causal but just co-occurrence.

5 CONCLUSION AND DISCUSSION
In this paper, we conduct the first trial to incorporate the entities and relationships extracted from social media data to simultaneously enhance the pandemic surveillance and detect the potential risk factors. We propose a dynamic graph neural network to learn the temporal dependency among nodes of different times and propagate the messages among the heterogeneous nodes. Extensive experiments show the effectiveness and robustness of our forecast model. We will open-source our framework and release the pre-trained CoronaBERT language model to facilitate future research in this direction.

Overall, we provide a generic solution for taking advantage of the informative entities and relationships in the social media data. It is straightforward to apply our approach to any future epidemiological surveillance. Our approach also has the potential to tackle other real-world problems, such as environment monitoring and crime detection. In the future, we will focus on detecting the risk factors in a more strict manner by identifying the relationship between the risk factors and the pandemic trends or predicted targets.
We would like to thank the anonymous reviewers for their helpful comments. This work was partially supported by the National Science Foundation [NSF-DGE-1829071, NSF-IIS-2031187] and the National Institutes of Health [NIH-R35-HL135772, NIH/NIBIR-R01-EB027650].

REFERENCES

[1] Hossein Abbassimehr and Reza Paki. 2021. Prediction of COVID-19 confirmed cases combining deep learning methods and Bayesian optimization. Chaos, Solitons & Fractals 142 (2021), 110551.

[2] Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. arXiv preprint arXiv:1904.03323 (2019).

[3] Nick Altieri, Rebecca L Barter, James Duncan, Raaz Devedzi, Karl Kumbur, Xiao Li, Robert Netzorg, Briton Park, Chandan Singh, Yan Shao Tan, et al. 2020. Curating a COVID-19 data repository and forecasting county-level death counts in the United States. arXiv preprint arXiv:2005.07882 (2020).

[4] Joan L Aron and Da B Schwartz. 1984. Seasonality and period-doubling bifurcations in an epidemic model. Journal of theoretical biology 110, 4 (1984), 665–679.

[5] Sujin Bae, Eunyoung Sung, and Ohbyung Kwon. 2021. Accounting for social media effects to improve the accuracy of infection models: combatting the COVID-19 pandemic and infodemic. European Journal of Information Systems (2021), 1–14.

[6] Norman TJ Bailey et al. 1975. The mathematical theory of infectious diseases and its applications. Charles Griffin & Company Ltd, 5a Crendon Street, High Wycombe, Bucks HP13 1LE.

[7] Ross Beckley, Cametria Weatherpoon, Michael Alexander, Marissa Chandler, Anthony Johnson, and Ghan S Bhutt. 2013. Modeling epidemics with differential equation. Tennessee State University Internal Report (2013).

[8] Andrea L Bertozzi, Elisa Franco, George Mohler, Martin B Short, and Daniel Sledge. 2020. The challenges of modeling and forecasting the spread of COVID-19. Proceedings of the National Academy of Sciences 117, 29 (2020), 16732–16738.

[9] Mu Haider Ali Bawas, Luis Tiago Parva, and MDR De Pantho. 2014. A SBR model for control of infectious diseases with constraints. Mathematical Biosciences & Engineering 11, 4 (2014), 761.

[10] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. 2015. Time series analysis: forecasting and control. John Wiley & Sons.

[11] Xavier Carreras, Lluís Màrquez, and Lluís Padró. 2002. Named entity extraction in clinical narratives. Charles Griffin & Company Ltd, 5a Crendon Street, High Wycombe, Bucks HP13 1LE.

[12] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for named entity recognition through classifier combination. In Proceedings of the seventh International Conference on Learning Representations. arXiv preprint arXiv:1505.01792 (2015).

[13] Herbert W Hethcote. 2000. The mathematics of infectious diseases. SIAM review 42, 4 (2000), 599–653.

[14] Jian Zhou, Zhiping Hu, and Zhaohui Lu. 2013. Evently: Modeling and analyzing Reshare Cascades with Hawkes Processes. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2627–2632.

[15] Wei Ji, Yandong Wang, Ming-Huang Tsou, and Xiaokang Fu. 2015. Using social media to detect outdoor air pollution and monitor air quality index (AQI): a geo-targeted spaetiotemporal analysis framework with Sina Weibo (Chinese Twitter). PloS one 10, 10 (2015), e141185.

[16] Yue Yang, Jinyu Cao, and Jiaying Li. 2019. A comprehensive study of the air pollution problem in Chinese cities. The Professional Geographer 67, 3 (2015), 351–363.

[17] William Ogilvy Kermack and Anderson G McKendrick. 1927. A contribution to the mathematical theory of epidemics. Proceedings of the royal society of London. Series A, Containing papers of a mathematical and physical character 115, 772 (1927), 700–721.

[18] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).

[19] Ismail Kirbas, Adnan Sizen, Aziz Dogdu Tuncer, and Fikret Sinasi Karakocoglu. 2020. Comparative analysis and forecasting of COVID-19 cases in various European countries with ARIMA, NARNN and LSTM approaches. Chaos, Solitons & Fractals 138 (2020), 110015.

[20] Kun Lan, Dan-tong Wang, Simon Fong, Lian-sheng Liu, Kelvin KL Wong, and Nilanjan Dey. 2018. A survey of data mining and deep learning in bioinformatics.
