An Event-Centric Prediction System for COVID-19

Xiaoyi Fu*, Xu Jiang*, Yunfei Qi*,
Meng Xu*, Yuhang Song*, Jie Zhang*, and Xindong Wu*†
*Mininglamp Academy of Sciences, Mininglamp Technology, Haidian District, Beijing, China
†Research Institute of Big Knowledge, Hefei University of Technology, China
Emails: fuxiaoyi@mininglamp.com, jiangxu@mininglamp.com, qiyunfei@mininglamp.com,
xumeng@mininglamp.com, songyuhang@mininglamp.com, zhangjie.c@mininglamp.com, wuxindong@mininglamp.com

***Abstract*—As COVID-19 evolved into a pandemic, a lot of effort has been made by scientific community to intervene in its spread. One of them was to predict the trend of the epidemic to provide a basis for the decision making of both the public and private sectors. In this paper, a system for predicting the spread of COVID-19 based on detecting and tracking events evolution in social media is proposed. The system includes a pipeline for building Event-Centric Knowledge Graphs from Twitter data streams about COVID-19, and uses the graph statistics to obtain a more accurate prediction based on the simulation of epidemic dynamic models. Experiments of 128 countries or regions conducted on the data set released by Johns Hopkins University on COVID-19 confirmed the effectiveness of the system. At the same time, the guidance our system provided to the plan of return-to-work for an enterprise has attracted the attention of and reported by top influential media.**

***Index Terms*—Event Detection, Event-Centric Knowledge Graphs, Epidemic Model, COVID-19, Time-Series Prediction**

I. Introduction

As COVID-19 evolved into a pandemic, the scientific community tried a variety of methodologies to predict the infectious trend of the disease to provide a basis for public sector planning decisions. In classical epidemic studies, usually, two types of models were used in the study of epidemic dynamics models: stochastic and deterministic models. Most popular epidemic dynamics models used are deterministic because they require less data, and are relatively easy to setup. The SEIR model, for example, includes four compartments represented by Susceptible, Exposed, Infectious and Recovered. The key difference of the SEIR model from its predecessor SIR model is: SEIR model considers the infected phase accounting for a latent period. However, classical epidemic models have several drawbacks in the big data era. The most salient shortcoming of classical epidemic dynamic models is the latency of response to the emergent event.

In Natural Language Processing forums, event detection has been an active research task and industrial systems has been deployed for social media stream analysis recently. The social media service provider Twitter provides a product feature named 'Trends For You' back-boned by a real-time event detection system [1]. Such service inspires our system which tracks top trends by detecting events in COVID-19 related social media stream and incorporates statistics of those trends into the epidemic prediction model.

The first three authors contribute equally and are alphabetically ordered.

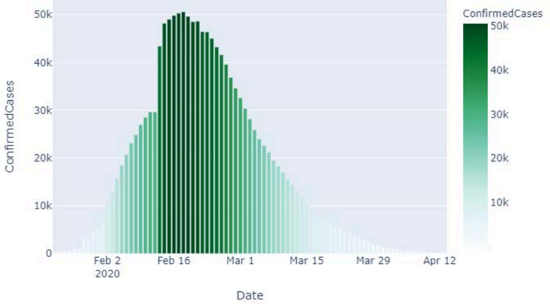

Fig. 1. Confirmed COVID-19 Cases Per Day in Hubei, China.

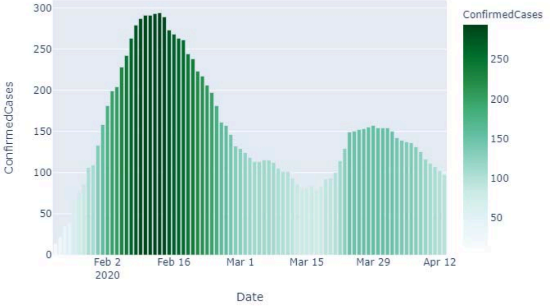

Fig. 2. Confirmed COVID-19 Cases Per Day in Beijing, China.

But how can we combine these two methodologies, namely, epidemic dynamics and event detection, to reach a more advanced prediction model to combat COVID-19? First, let's take a look at the statistics from Jan 22nd to Apr 13th country by country and state by state. As shown in Table I, the confirmed cases and fatalities vary significantly between 128 countries so as 33 states in China. We believe that the differences in international and provincial statistics are due to the difference of emergent date of patient zero and sizes of susceptible. Comes to the time-series, as shown in Fig. 1, the number of confirmed cases of China's Hubei apparently lags in trend comparing to Beijing's curve as shown in Fig. 2. Comparing the trend of fatalities according to the change of

DOI 10.1109/ICBK50248.2020.00037

© IEEE 2020. This article is free to access and download, along with rights for full text and data mining, re-use and analysis.

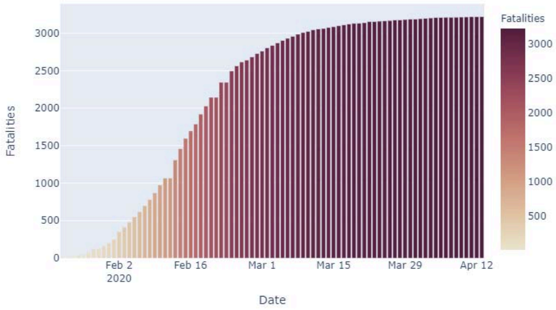

Fig. 3. COVID-19 Fatalities Per Day in Hubei, China.

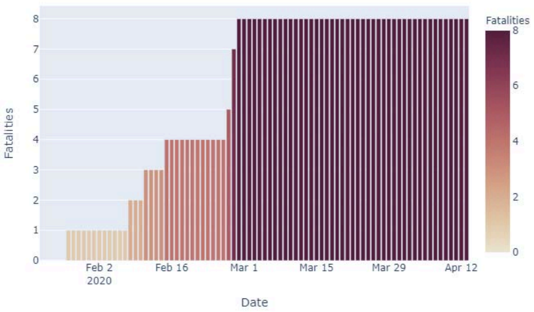

Fig. 4. COVID-19 Fatalities Per Day in Beijing, China.

TABLE I
STATISTICS FROM JAN 22ND TO APR 13TH

By Country (Global)	*Confirmed Cases*	*Fatalities*
Count	128	128
Mean	6225	640
Std	175556	2776
Max	103616	20465
By State (China Only)	***Confirmed Cases***	***Fatalities***
Count	33	33
Mean	1839	101
Std	8764	560
Max	50633	3221

date in Fig. 3 and Fig. 4, we can observe a smoother curve of Hubei province comparing to the city of Beijing's. To our assumption, the differences in the distribution in time series within the same country or region are caused by emergent events.

Based on the observations above, in our work two mechanisms are designed to address different problems respectively. On the one hand, our system minimizes the mean error of the epidemic model by fitting and simulation using different parameter settings for each country or region. Considering there are over 100 countries and regions globally, it is impractical to set all parameters by hand, so we designed a salable mechanism to search the best fits. On the other hand, by representing events from Twitter streams as knowledge graphs, our system incorporates a set of graph statistics as features into regression models to correct the prediction error of the epidemic models. Our system is experimented on a global data set and its effectiveness is verified in real-world practice.

II. Related Work

A. Epidemic Model

In research [2], a special cellular automata model is used, which can well reproduce the time evolution of diseases given by SIR model. In research [3], the virus infection coefficient R0 is proposed and applied in SIR model to simulate the spread of epidemic. On January 24, five authors including Jonathan published paper [4] to predict the number of people infected with covid-19 virus in Wuhan in the future. In this paper, SEIR model is constructed to estimate the infection coefficient of the virus according to the real data published by the government. It predicted that the number of people infected in Wuhan will be 9217-14245 by January 21. On January 31, the Lancet published the research [5] of Hong Kong scientists. The authors used the classic SEIR model in the dynamics of infectious diseases to construct differential equation. Then they predicted the turning point of epidemic in May according to the parameter regeneration number R0, the average incubation time De and the average treatment time Di. In research [6], the author introduces four kinds of people based on SEIR model: isolation of susceptible, isolation of latent, isolation of infected and inpatients. Combined with the traffic flow data from Wuhan to Beijing, it predicted that after the closure of Wuhan, the number of cases in Beijing can be reduced by 91.14% in next week. On February 28, academician Zhong Nanshan and others published a research [7] in the medical journal JTD. Their team used the LSTM model and used the case statistics data of SARS from April to June 2003 as the training set, with the epidemiological parameters of covid-19 to predict that the epidemic reached peak at the end of February and was basically controlled at the end of April. In the research [8] published by Zhu huaiqiu, a professor of the school of technology of Peking University, VHP (virus host prediction) was developed based on the deep learning algorithm. They found that the infectivity mode of mink virus is closer to the new coronary 2019 ncov virus. In the PKU team's simulation forecast for epidemic situation [9], the author made two improvements to SEIR model: (1) Further dividing the population into isolated and uninsulated patients; (2) Fitting the basic regeneration coefficient (R0) of virus with index function.Then they fitted out that the epidemic situation in Hubei Province reached the inflection point in the middle of February. In [10], the authors introduced isolation susceptible (SQ), isolation latent (EQ) and isolation infected (IQ) population on the basis of SEIR model.They reconstruct the dynamic equation, and predict that the number of people infected with covid-19 in Hubei Province reached the peak on February 19. In [11], the author combined with the flow of people in the city based on the classic SIR model. The research analyzed the effect of reducing public transport travel and blocking the densely populated places on the spread of the epidemic.

B. Event-Centric Knowledge Graphs

Currently, there are many definitions of event detection problems. Orr et al. [12] regard event detection as the recognition of trigger words to determine the type of event. However, concerning social data streams, it is difficult to predict trigger words, given the unstructured and noisy nature of the documents. In Twitter data, McMinn et al.[13] performed event detection and aggregated tweet streams into appropriate event-based clustering. Guille and Favre [14] also clustered related words from the tweet stream. Some important event detection techniques were introduced by [15, 16]. These techniques can be roughly divided into feature-pivot (FP) or document-pivot (DP) methods. The former corresponds to grouping entities within documents according to their distribution, while the latter requires clustering on documents according to their semantic distance. A popular category of FP technology is topic detection, which attempts to identify events by modeling documents as "a mixture of topics, where topics are probability distributions over words" [17].

However, many topic detection methods cannot fully capture the "burst" or speed of words over time, which is essential for distinguishing events from non-events [18, 19]. Bursty terms on Twitter are defined as terms that appear at an unusually high tweet rate. Many studies have attempted to use bursty term tracking to discover events. TwitterMonitor [20] recognizes bursty words and then used the greedy algorithm to merge them into groups according to the co-occurrence in the tweet so as to perform event detection. Each group represents an event. EDCoW [21] tracks all co-occurrences over a time window and used wavelet decomposition to identify bursty words. Most of the above methods did not consider the evolution of events. Some recent studies [22, 23, 24] proposed the use of incremental clustering [25] to solve the problem of event evolution. As new data arrives at the stream, the model will be updated incrementally. Due to the large size of the update, this method may not be suitable for Twitter Firehose. Fedoryszak M et al [1] solve this problem by linking event clusters and achieves event detection with evolution tracking in real-time through modeling events as cluster chains and addressing scaling concerns with new design choices.

The evolution of events are more practically appealing but even more challenging. For example, the previous work of the narrative event chain did not specifically focus on the learning narrative. All in all, the topic signature is a set of terms indicating the subject [26]. These terms can capture certain narrative relationships, but the model requires training data for topic classification. Bean and Riloff [27] proposed the use case framework network as a contextual role knowledge for anaphora resolution. Brody [28] proposed a method similar to the case framework. This method discovers high-level correlations between verbs by grouping verbs that share the same vocabulary items in the subject/object position. Chamber et al [29] described a three-step process to learning narrative event chains and introduce two evaluations: the narrative cloze to evaluate event relatedness, and an order coherence task to evaluate narrative order.

The latest work on unsupervised inference about the sequence of prototype events in texts began with Chambers and Jurafsky [29]. Based on this, Chambers and Jurafsky [30] make the inductive representation closer to the concept of the semantic framework and infer the event schema. Chambers [31] and Cheung, Poon, and Vanderwende [32] also focus on schema induction. Various developments of C&J08 have been proposed. Jans et al. [33] compared the methods of collecting and using model statistics to measure the correlation between events. Compared with C&J08's PPMI statistics, they can get better results through the bigram conditional probability model. Balasubramanian et al. [34] used open-domain relations extracted by the information extraction system Ollie instead of verbs to capture more information about the event. They also focus on event schema extraction. And another line of work approaches event knowledge acquisition using event schema descriptions (ESDs), natural language descriptions of typical sequences of events, written by hand (Regneri, Koller, and Pinkal [35]; Regneri et al. [36]; Modi and Titov [37]).

Predicting the relationships between events described in the text is essential for many applications, such as dialogue generation. Script event prediction is one of the most challenging task in this area. This task was first proposed by Chambers and Jurafsky [31], who defined it as providing the context of an existing event and needed to select the most reasonable follow-up event from the candidate list. Previous studies established prediction models based on event pairs [31, 38] or event chains [39]. Despite the successful use of event pairs and event chains, the rich connections between events have not been fully explored. Event evolutionary graph is another emergent research topic as a representation tool of relationship between event. Structurally, EEG is a directed cyclic graph, whose nodes are events and edges stand for the relations between events, e.g. temporal and causal relations. Duvenaud et al. [40] introduced a convolutional neural network that can be run directly on the graph, which can be used for end-to-end learning prediction tasks. Kipf and Welling [41] proposed a scalable semi-supervised learning graph method based on effective variants of convolutional neural networks. Li et al. Li Z et al [42] proposed a scaled graph neural network (SGNN), which is feasible to large-scale graphs and borrow the idea of divide and conquer in the training process that instead of computing the representations on the whole graph, SGNN processes only the concerned nodes each time.

Based upon the above advancement in Nature Language Processing research, an approach to create Event-Centric Knowledge Graphs (ECKGs) using state-of-the-art NLP tools was presented in [43]. We follow the same notion of ECKGs in this paper, because at the core of our system, the vision is to acquire and represent the fragmented knowledge about emergent event and use it to enhance epidemic prediciton by tracking the evolution of such knowledge.

III. Epidemic Dynamics

A. Model

Similar to SARS, COVID-19 has an incubation period. Once a patient recovers, he or she will not be re-infected or infect others in a short time, so SEIR model is adopted. In this model, there are 4 notions of population: $S(t)$ represents susceptible population, $E(t)$ represents exposed population, $I(t)$ represents infected population, and $R(t)$ represents recovered population. In the SEIR model, there are be four differential equations respectively, calculating the change in proportion of each population at time t:

$$\frac{dS}{dt} = -\beta \cdot S(t) \cdot \frac{I(t)}{N} \tag{1}$$

$$\frac{dE}{dt} = \beta \cdot S(t) \cdot \frac{I(t)}{N} - \sigma \cdot E(t) \tag{2}$$

$$\frac{dI}{dt} = \sigma \cdot E(t) - \gamma \cdot I(t) \tag{3}$$

$$\frac{dR}{dt} = \gamma \cdot I(t) \tag{4}$$

In the above formula, N represents the total number of people affected by the epidemic:

$$N = S + E + I + R$$

B. Data sets

The data sets released by Johns Hopkins CSSE[1] include globally reported epidemic numbers started from Jan 22nd. The data sets are divided according to different countries and regions. The data set roughly consists of three parts: daily number of confirmed cases in each country or region, daily number of fatalities in each country or region, and daily number of recovered cases in each country or region.

Treatement of number of confirmed. Notably, number of confirmed cases in the data refers to the cumulative number of confirmed cases per day, so it is not consistent with the hypothesis of SEIR model. Confirmed in the infectious disease refers to the number of cases so far, so it should be processed in the data. From the situation of the epidemic, once the confirmed patient dies, his or her body will be immediately disposed of, that is, the dead patient will not continue to be infectious to normal people. Therefore, the confirmed case in the data is treated as follows: Current confirmed retention = cumulative number - recovered number - number of deaths.

Data division. Since the spread of COVID19 virus is affected by many factors such as city size, population density and total population, different models should be established for different countries and regions, so the data are divided according to each country and region for simulation.

TABLE II
Parameter Search Space of SEIR

Parameter Name	**Search Space**		
	Lower Bound	***Upper Bound***	***Step Size***
S_0	10 * $max(I_t)$	500 * $max(I_t)$	50 * $max(I_t)$
β	0.1	10	0.1
σ	0.1	1	0.01
γ	1/14		
Period	Jan 22nd to Apr 13th		

C. Experiments

The number of susceptible people varies by regions. For example, China has adopted strong travel restrictions and residents must wear masks when going out, which has greatly reduced the total number of susceptible people S. Therefore, a heuristic is adopted to estimate S_0. The maximum number of infected people is multiplyed by a amplification ratio which is searched between 10 and 500 with the step length of 50 where S_0 represents the total number of people S when $t = 0$. The value of I_0 is the number of people infected when the epidemic is first detected in each region. R_0 takes the value of 0.

Initialization. As the parameter β is affected by various factors (measures taken, population density, epidemic prevention equipment, urban development), the values of β in different regions fluctuate greatly. Empirically, the values of β is known to vary from 0.1 to 10, so the search range of β is set as $(0.1, 10)$ and the step size is set as 0.1. σ is a patient's conversion probability of status from suseptible to exposed, and the search space of this value is also set as $(0.1, 1)$, and the step size is set as 0.01. According to the current published epidemic data and medical research, the recovery period of the infected person is 14 days, so it is assumed that $\gamma = 1/14$.

Simulation. For simulation, the period of 82 days are sampled 8200 times. So the step size of is 1/100. The calculated loss is calculated on the 82 days instead of the 8200 data points in total. And the interpolated point on each 82 days should be used to calculate the loss. For example, the predicted results of the next day should be calculated with the mean square error of points with an index number of 200.

Parameter search. The parameter search process involves three parameters, S_0, β and σ respectively, and the corresponding parameters are optimized and matched by grid search. In this process, to scale up to over 100 countries or regions, multi-thread parallelization is adopted to speed up the search. Computation task of each country is pushed into a message queue and each consumer thread runs the pipeline of search, simulate and error calculation for one country.

Prediction. The loss is calculated in the parameter search process, then is sorted in ascending order, and the set of parameters with minimum loss is selected as the final parameters for this country or region. After the parameters of each region are obtained, the epidemic trend in that region can be predicted by running the simulation on future dates. The predicted of number of confirmed cases, recovered cases and fatalities are obtained as features for downstream regression model.

[1] https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series

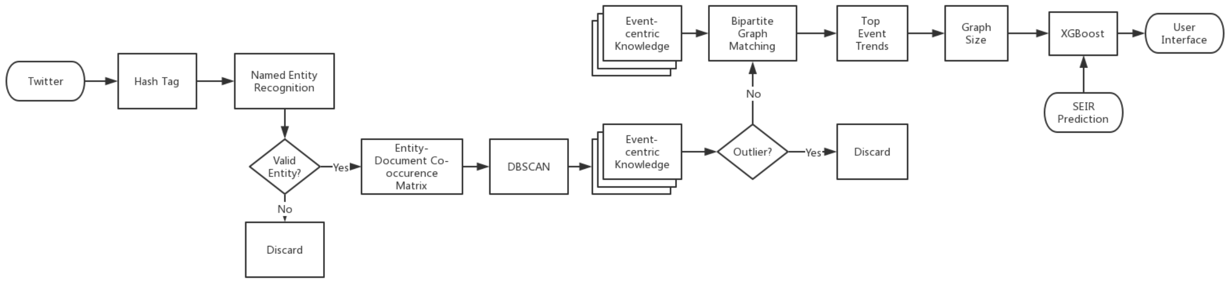

Fig. 5. The System of Epidemic Prediction with Event-Centric Knowledge Graphs.

IV. Prediction with Event-Centric Knowledge Graphs

Along with the SEIR models fit on global data to predict the evolution of the pandemic, as shown in Fig. 5, an Event-Centric system is introduced to calibrate the prediction error for spreading of COVID-19. First of all, Twitter posts are pulled from the official API to build pieces of Event-Centric Knowledge Graphs over the week before the prediction date. Then the sizes of graphs for the top trends are fed into an XGBoost model as features along with factors such as country, region, date, SEIR prediction of confirmed number, SEIR prediction of suspected number, etc. Finally, the established epidemic prediction model is verified by prediction error of number of confirmed cases and number of fatalities.

A. Event-Centric Knowledge Graphs

Event Detection. At the beginning of event detection pipeline, Twitter posts are first filtered by topic. In our experiments, 1500 of posts per day are randomly pulled from Twitter by searching COVID-19 related keywords such as 'COVID-19', 'Coronavirus', etc. Following the system definition introduced in [1], we view event as fragmented knowledge consisting of entities and vectors of their co-occurrence in Twitter posts. In another word, an event is a piece of knowledge graph with entities frequently occur in the same set of Twitter posts as nodes. In our definition, an entity should meet either one of the following two criterion: 1) it is a hash tag; 2) it has the same boundaries as an output of named entity by Stanford Corenlp [44] under one of the following four entity types: ORG, PERSON, LOCATION, MISC. We do the entity recognition step for a batch of posts and for each batch an entity-document co-occurrence matrix is build for clustering using DBSCAN [45]. In our experiments, a batch size of 100 posts are used and the average number of identified entities is 230-260, the average number of clusters is about 60 and the number of outliers is roughly between 5-10.

Event Evolution. Each entity cluster a.k.a Event-Centric Knowledge Graph is then linked to clusters formed by the previous batch to form an event trend by an algorithm based on bipartite graph matching proposed in [1]. With Twitter stream comes into the pipeline in the form of batches, the size of each graph grow with time. In our experiments with data from Apr 1st to Apr 13rd, a total number of 77 graphs are formed. The average number of entities in each graph per day deviates by two standard deviations and there is an average of 4 entities per graph per day.

B. Prediction Model

XGBoost [46] is an iterative regression tree algorithm and software package, which is an improved variant of Gradient Boost Decision Tree algorithm. XGBoost supports multi-threaded concurrent tasks because the feature columns are sorted and stored in memory as blocks that can be reused in iterations. XGBoost allows specification of the default direction of the branch for missing or specified values, which improves the efficiency of the algorithm. XGBoost also contains a large number of CART regression trees internally and an internal regularization over-fitting technique ensures the model's robustness. We choose the XGBoost package because of its robustness and efficiency.

In our model, SEIR prediction is included to capture a common understanding to the transmission mechanisms of viruses, including COVID-19. Moreover, the predicted indexes can be extracted from them as the features of our model. The purpose of event trends detection is to make up for the SEIR model when it fails to response to emergent events.

As shown in the Table III & IV, there are 6 families of features in total and they can be divided into three sets: features from the original data, features from SEIR model prediction, and features from top event trends. Features from the original data include State (City), Country (Region), Date. For date, the month, day are extracted and converted into numeric codes. Features from SEIR model prediction include: prediction of susceptible, prediction of confirmed infected, prediction of recovered, and prediction of fatalities. Features from top event trends are cluster sizes of top 10 event trends on the day to be predicted.

C. Parameter Tuning

Cross validation is used to solve the problem of high deviation or high variance in model training. Its working

TABLE III
SETTINGS FOR PREDICTION MODEL

Name	**Hyper-parameters of XGBoost**		
	Learning Rate	***Max Depth***	***N Estimators***
	0.1	6	200
$Feature_1$	Prediction Date		
$Feature_2$	Prediction of susceptible		
$Feature_3$	Prediction of confirmed infected		
$Feature_4$	Prediction of recovered		
$Feature_5$	Prediction of fatalities		
$Feature_6$	Graph Size of Top $1-10$ Trend		

principle is to take part of the original data as the training set, and the rest as the validation set, then use the validation set to test the model obtained from the training set to evaluate the performance of the model. In order to reduce the variability of the validation results, the original data set is divided for multiple times to obtain complementary subsets and conduct multiple cross-validation. Worth mentioning, the cross validation not only take the training error into consideration but also the generalization error.

In our experiment, k cross-validation method is adopted to divide the original data sets into 5 groups. Each data subset is verified once, and the rest 4 subset data is used as the training set. The mean prediction accuracy of the final verification set of 5 models is used to evaluate the performance indicators of the model so as to effectively avoid over-fitting and underfitting. The optimal hyper-parameters are shown in Table III.

D. Experiment Setting

In our experiment, the XGBoost algorithm is used to train the model in the COVID-19 epidemic data set, mainly to fit the residual of the prediction results and we use Mean Squared Error as the training objective as well as evaluation metric. Because there are two columns to be predicted: the confirmed cases and the number of fatalities, for scoring criteria we have to evaluate the two targets respectively, and then calculate the average of two targets' evaluation metrics. Based on the above experimental Settings, the specific steps of our experiments are as follows and their results are shown in Table IV.

- **Data cleaning**: completion of missing values.
- **Feature extraction**: optimally selected for confirmed cases prediction and fatalities prediction respectively, see 'E. Result Analysis'.
- **Data division**: both the training data and the test data are divided by country and province.
- **Model training**: search for the optimal model hyper-parameters by training on each training segment then using the trained model to predict the corresponding test segment, then combine the final results and output.
- **Evaluation**: MSE and RMSLE are used as the metric to score the test set.

The results shown in Table IV are obtained using XGBoost algorithm as the regression model and 5 fold cross-validation to tune the hyper-parameters. All metrics are computed from the best model on the test set.

TABLE IV
PREDICTION ERROR WITH DIFFERENT FEATURE SETS

Feature Sets	**MSE**		
	Fatalities	***Confirmed***	***Mean***
Single Model	1498356	43859940	22679148
Original + SEIR	5862	220317	113089
Original + SEIR + Event	5862	210240	108051

E. Result Analysis

Three experiments are conducted and their results are shown in Table IV.

- V0: Single model for all countries or regions with original data and SEIR prediction as features.
- V1: Separate training models by country and province and make predictions with original data , and SEIR model prediction as features.
- V2: Separate training models by country and province and make predictions with original data, event trends, and SEIR model prediction as features.

By comparing the above experimental results, we have observations as follows:

- Compared with V0, V1 is divided into separate training models according to country and province for prediction. The model accuracy is significantly improved, and both MSE and RMSLE are smaller.
- Compared with V1, V2 introduces the features from event trend detection and the error of prediction of fatalities changed little, meanwhile after the introduction of features related to Twitter posts, the prediction of confirmed cases is faced with a much smaller MSE value, and RMSLE slightly larger, indicating that the model error of the number of confirmed patients is reduced, but the number of confirmed patients is slightly higher.

To sum up, in prediction of fatalities, the original data and SEIR model prediction are good enough to build a decent prediction model. For prediction of confirmed cases, our experiments show that the event trends drive an significant improvement on prediction accuracy and thus worthy of further investigation.

V. CASE STUDY: ESTIMATE THE OPTIMAL DAY OF RETURN TO WORK

From December 9th, 2019, when COVID-19 virus was first detected, to the official announcement of the new pneumonia on December 31st, 2019, and as of 16:21 on February 4th, 2020, there were 20,503 confirmed cases and 426 deaths in China. The virus was spreading at five times the rate of SARS in 2003, and some have even compared its severity to that of the 1918 Spanish flu.

As of Feburary 4th, a few epidemic experts claimed the turning point would come on the 15th day of the first lunar month (February 8th). Xien Gui, a well-known epidemic expert, predicted that the COVID-19 outbreak before the fifteenth day of the month may appear inflection point. Professor Wenhong

Zhang, head of the Shanghai medical treatment expert group, asserted that the main battle to control a COVID-19 should end within February and come to an end in March. Based on latest statistics published by the national centers for disease control and prevention (CDC) as of Feburary 4th, our system predicted based on SEIR model that the spread rate might peak and then begin to decline on February 11th through the time-varying curve of the number of exposed patient. The conclusion was used to guide the planning of return-to-work for an enterprise with more than 3000 employees and reported by The People's Daily[2].

A. The Lurker's Curve

Basic regeneration number (R_0) refers to the number of second-generation cases that can be infected by a virus carrier after entering a susceptible population. According to the estimation of Kermack and McKendric [2], when the basic regeneration number R_0 continues to be less than 1, the newly added infection number will eventually converge to 0 over time. In other words, the virus will die on its own at this time and no longer pose a threat of infectious diseases.

Basic formula of basic reproduction number [3]:

$$R_0 = r \cdot c \cdot d$$

where r is the virus transmission (the probability that the virus carrier will transmit the virus to the contact person), c is the average contact rate between the susceptible population and the virus carrier, and d is the expected duration of the infection. Based on the policy in China, we have reason to believe that all discovered infected victims have been quarantined, and the quarantined virus carriers are not within the possible contact area, which means all we need to pay attention to are undiscovered viruses carriers a.k.a the un-quarantined exposed. Assuming that the virus exposed is quarantined the first time it is discovered, we used the curve of un-quarantined exposed rate over time (the lurker's curve) as an estimation of change in the number of exposed over time.

If we do not take into account: 1) environmental changes of employees' exposure to the virus and 2) protective methods used during the simulation time, then we can assume that virus transmission (r) does not change with time and the expected duration of infection (d) is a fixed value. Therefore, the change in the basic regeneration number (R_0) of undiscovered virus carriers is consistent with the average contact rate (c) between susceptible people and virus carriers. It is assumed that the average exposure rate (c) between the susceptible population and the virus carrier is consistent with the un-quarantined exposed ratio of the unexposed virus carriers (Un-quarantined Exposed Rate), combining assumption made above, we can conclude that the basic regeneration number (R_0) curve of undiscovered virus carrier is consistent with the lurker's curve.

B. The Passenger's Curve

If the contact between susceptible people and virus carriers in public transportation environment can be viewed as a Bernoulli test in which M passengers encounter each other with a fixed probability q and the un-quaranted exposed exposure Rate (u) 's curve equals the lurker's curve divided by M. Then the average exposure Rate (c) between susceptible people and virus carriers can be considered as the expectation of binomial distribution which approximately equals to $q \cdot u \cdot M^2$. Since q does not change with time, the risk of employees' virus infection in public transportation environment can be depicted by the product of the lurker's curve and the square of the passenger flow curve of provinces returning to Beijing. We name it as the passenger's curve.

To estimate the lurker's curve, we first collected the number of infected, deaths, and recovered from January 1st, 2020 to February 3rd, 2020. Then our prediction system were applied. In order to estimate the number of passengers returning to Beijing from various provinces this year, we first calculated the passenger volume curve of returning to Beijing during the Spring Festival travel rush by the total passenger volume of each date and the proportion of passenger volume of railway during each Spring Festival travel rush period of previous years. Then through the Spring Festival transport total passenger volume report, we obtained the passenger flow of each date of this year to predict the passenger flow of each province returning to Beijing after February 4th.

Finally, by scrutinizing the trend of the passenger's curve, we reached a conclusion that the optimal day of return-to-work was before the 15th day of the first lunar month (February 8th).

VI. Conclusion

The contributions of this work are three folds. First, a pipeline for detecting and representing COVID-19 related event trends are designed and implemented. Second, graph statistics from its results are applied to the prediction of number of confirmed cases and fatalities by combining epidemic dynamic model with top event trends using XGBoost regression model. Lastly, the prediction system we proposed is proved to be effective in real-world practice by planning optimal day for return to work. Controlled experiments showed preliminary results about the effectiveness of incorporating events into epidemic prediction models. For future work, more types of relationships between entities have to be constructed to enrich the feature set obtained from the Event-Centric Knowledge Graphs.

References

[1] Fedoryszak, Mateusz, et al. "Real-Time Event Detection on Social Data Streams." Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 2774–2782.

[2] Kermack, William Ogilvy, and A. G. McKendrick. "A Contribution to the Mathematical Theory of Epidemics." Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 115, no. 772, 1927, pp. 700–721.

[3] James Holland Jones, "Notes on R0", Stanford University, April 13, 2019.

[4] Jonathan M. Read, Jessica Re Bridgen, Derek At Cummings, Antonia Ho, "Novel coronavirus 2019-nCoV: early estimation of epidemiological parameters and epidemic predictions,"MedRxiv, January 2020.

[2] https://wap.peopleapp.com/article/5220507/5122857

[5] Wu, Joseph T., et al. “Nowcasting and Forecasting the Potential Domestic and International Spread of the 2019-NCoV Outbreak Originating in Wuhan, China: A Modelling Study.” The Lancet, vol. 395, no. 10225, 2020, pp. 689–697.

[6] Tang, Biao, et al. “Estimation of the Transmission Risk of the 2019-NCoV and Its Implication for Public Health Interventions.” Journal of Clinical Medicine, vol. 9, no. 2, 2020, p. 462.

[7] Yang, Zifeng, et al. “Modified SEIR and AI Prediction of the Epidemics Trend of COVID-19 in China under Public Health Interventions.” Journal of Thoracic Disease, vol. 12, no. 3, 2020, pp. 165–174.

[8] Qian Guo, Mo Li, Chunhui Wang, Peihong Wang, “Host and infectivity prediction of Wuhan 2019 novel coronavirus using deep learning algorithm”,BioRxiv, January 2020.

[9] Chen, Baoquan, et al. “Data Visualization Analysis and Simulation Prediction for COVID-19.” ArXiv Preprint ArXiv:2002.07096, 2020.

[10] Cao Shengli, et al. ”Modified SEIR infectious disease dynamic model applied to the prediction and assessment of the 2019 Coronavirus Disease (COVID-19) epidemic situation in Hubei Province.” Journal of Zhejiang University (Medical Edition), vol. 49, no. 1, 2020, pp. 0–0.

[11] Maher Elbayoumi, “Modelling the coronavirus epidemic in a city”, unpublished, January 2020.

[12] Orr, Walker, et al. “Event Detection with Neural Networks: A Rigorous Empirical Evaluation.” EMNLP 2018: 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 999–1004.

[13] Andrew J. McMinn and Joemon M. Jose. “Real-Time Entity-Based Event Detection for Twitter.” CLEF’15 Proceedings of the 6th International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction, vol. 9283, 2015, pp. 65–77.

[14] Adrien Guille and Cécile Favre. “Event Detection, Tracking, and Visualization in Twitter: A Mention-Anomaly-Based Approach.” Social Network Analysis and Mining, vol. 5, no. 1, 2015, p. 18.

[15] Farzindar Atefeh and Wael Khreich. “A Survey of Techniques for Event Detection in Twitter.” Computational Intelligence, vol. 31, no. 1, 2015, pp. 132–164.

[16] Mahmud Hasan, Mehmet A Orgun, and Rolf Schwitter. “A Survey on Real-Time Event Detection from the Twitter Data Stream.” Journal of Information Science, vol. 44, no. 4, 2018, pp. 443–463.

[17] Giovanni Stilo and Paola Velardi. “Efficient Temporal Mining of Micro-Blog Texts and Its Application to Event Discovery.” Data Mining and Knowledge Discovery, vol. 30, no. 2, 2016, pp. 372–402.

[18] Jianxin Li, Zhenying Tai, Richong Zhang, Weiren Yu, and Lu Liu. “Online Bursty Event Detection from Microblog.” Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing On , 2014, pp. 865–870.

[19] Giovanni Stilo and Paola Velardi. “Efficient Temporal Mining of Micro-Blog Texts and Its Application to Event Discovery.” Data Mining and Knowledge Discovery, vol. 30, no. 2, 2016, pp. 372–402.

[20] Michael Mathioudakis and Nick Koudas. “TwitterMonitor: Trend Detection over the Twitter Stream.” Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, 2010, pp. 1155–1158.

[21] Jianshu Weng and Bu-Sung Lee. “Event Detection in Twitter.” Fifth International AAAI Conference on Weblogs and Social Media, 2011.

[22] Hila Becker, Mor Naaman, and Luis Gravano. “Beyond Trending Topics: Real-World Event Identification on Twitter.” Fifth International AAAI Conference on Weblogs and Social Media, 2011.

[23] Miles Osborne, Sean Moran, Richard McCreadie, Alexander Von Lunen, Martin D Sykora, Elizabeth Cano, Neil Ireson, Craig Macdonald, Iadh Ounis, Yulan He, et al. “Real-Time Detection, Tracking, and Monitoring of Automatically Discovered Events in Social Media.” Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2014, pp. 37–42.

[24] Saša Petrović, Miles Osborne, and Victor Lavrenko. “Streaming First Story Detection with Application to Twitter.” Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2010, pp. 181–189.

[25] Mahmud Hasan, Mehmet A Orgun, and Rolf Schwitter. “TwitterNews: Real Time Event Detection from the Twitter Data Stream.” PeerJ, vol. 4, 2016.

[26] Chin-Yew Lin and Eduard Hovy. “The Automated Acquisition of Topic Signatures for Text Summarization.” COLING ’00 Proceedings of the 18th Conference on Computational Linguistics - Volume 1, vol. 1, 2000, pp. 495–501.

[27] David Bean and Ellen Riloff. “Unsupervised Learning of Contextual Role Knowledge for Coreference Resolution.” HLT-NAACL, 2004, pp. 297–304.

[28] Samuel Brody. “Clustering Clauses for High-Level Relation Detection: An Information-Theoretic Approach.” Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, 2007, pp. 448–455.

[29] Nathanael Chambers and Dan Jurafsky. “Unsupervised Learning of Narrative Event Chains.” Proceedings of ACL-08: HLT, 2008, pp. 789–797.

[30] Nathanael Chambers and Dan Jurafsky. “Unsupervised Learning of Narrative Schemas and Their Participants.” Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2009, pp. 602–610.

[31] Nathanael Chambers. “Event Schema Induction with a Probabilistic Entity-Driven Model.” Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1797–1807.

[32] Jackie Chi Kit Cheung, Hoifung Poon, and Lucy Vanderwende. “Probabilistic Frame Induction.” Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2013, pp. 837–846.

[33] Bram Jans, Steven Bethard, Ivan Vulić, and Marie-Francine Moens. “Skip N-Grams and Ranking Functions for Predicting Script Events.” Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, 2012, pp. 336–344.

[34] Niranjan Balasubramanian, Stephen Soderland, and Oren Etzioni. “Generating Coherent Event Schemas at Scale.” Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1721–1731.

[35] Michaela Regneri, Alexander Koller, and Manfred Pinkal. “Learning Script Knowledge with Web Experiments.” Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 2010, pp. 979–988.

[36] Michaela Regneri, Alexander Koller, and Manfred Pinkal. “Learning Script Participants from Unlabeled Data.” Proceedings of the International Conference Recent Advances in Natural Language Processing 2011, 2011, pp. 463–470.

[37] Ashutosh Modi and Ivan Titov. “Learning Semantic Script Knowledge with Event Embeddings.” ICLR (Workshop), 2014.

[38] Mark Granroth-Wilding and Stephen Clark. “What Happens next? Event Prediction Using a Compositional Neural Network Model.” AAAI’16 Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016, pp. 2727–2733.

[39] Zhongqing Wang, Yue Zhang, and Chingyun Chang. “Integrating Order Information and Event Relation for Script Event Prediction.” Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 57–67.

[40] David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P. Adams. “Convolutional Networks on Graphs for Learning Molecular Fingerprints.” NIPS’15 Proceedings of the 28th International Conference on Neural Information Processing Systems, Volume 2, 2015, pp. 2224–2232.

[41] Thomas N. Kipf, and Max Welling. “Semi-Supervised Classification with Graph Convolutional Networks.” ICLR 2017: International Conference on Learning Representations 2017, 2017.

[42] Zhongyang Li, Xiao Ding, and Ting Liu. “Constructing Narrative Event Evolutionary Graph for Script Event Prediction.” IJCAI 2018: 27th International Joint Conference on Artificial Intelligence, 2018, pp. 4201–4207.

[43] Marco Rospocher, et al. ”Building event-centric knowledge graphs from news.” Journal of Web Semantics, Volumes 37–38, 2016, pp. 132-151.

[44] Manning Christopher, et al. “The Stanford CoreNLP Natural Language Processing Toolkit.” Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2014, pp. 55–60.

[45] Ester Martin, et al. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.” Proc. 1996 Int. Conf. Knowledg Discovery and Data Mining (KDD ’96), 1996, pp. 226–231.

[46] Chen Tianqi, and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System.” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.