Civil Unrest Event Forecasting Using Graphical and Sequential Neural Networks

Zheng Chen(✉) and Yifan Wang

School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China
zchen@uestc.edu.cn

Abstract. Having the ability to forecast civil unrest events, such as violent protests, is crucial because they can lead to severe violent conflict and social instabilities. Civil unrests are comprehensive consequences of multiple factors, which could be related to political, economic, cultural, and other types of historical events. Therefore, people naturally organize such historical data into time-series data and feed it into an RNN-like model to perform the forecasting. However, how to encode discrete historical information into a unified vector space is very important. Different events may have extensive and complex relationships in time, space, and participants. Traditional methods, such as collecting indicators of various fields as features, miss the vital correlation information between events. In this work, we propose a Graph Neural Network based model to learn the representation of correlated historical event information. By using the dates, events, participants, and locations as nodes, we construct an event graph so that the relationship between events can be expressed unambiguously. We organize date-node's representations into time-series data and use an LSTM to predict if there will be a violent protest or demonstration in the next few days. In the experiments, we use historical events from Hong Kong to evaluate our system's forecasting ability in 1-day, 2-day, and 3-day lead-time. Our system achieves recall rates of 0.85, 0.86, 0.88, and precision rates of 0.75, 0.77, 0.75, respectively. We also discussed the impact of longer prediction lead times, and external events in China Mainland, the United States, and the United Kingdom on the Hong Kong civil unrest event prediction.

Keywords: Event forecasting · Civil unrest event · Graph neural network

1 Introduction

Civil unrest is a kind of social problem that includes riots, violent demonstrations, marches, protests, barricades, and strikes [6]. Sometimes it can cause a significant amount of economic and political loss [3]. Thus, predicting the occurrence of violent protests is of interest to policymakers and citizens, as these may lead to civil unrest and regional instability, threatening to life, and property.

Supported by the Sichuan Science and Technology Plan Project 2020YFG0009.

I. Farkaš et al. (Eds.): ICANN 2021, LNCS 12893, pp. 192–203, 2021.
https://doi.org/10.1007/978-3-030-86365-4_16

Factors in the emergence of civil unrest include social interactions and injustices, changes in domestic and international policies cultural awareness, and economic factors, such as poverty, unemployment levels, and food prices [8]. All these factors can be regarded as historical information and found in historical events. In recent years, open-source data, such as social media content and event data, have been used with varying degrees of success to forecast civil unrest [17]. By using traditional machine learning models, Muthiah et al. [11] and Qiao et al. [12] predicted civil unrest events in many countries in Southeast Asia and Latin America. By using neural network models such as LSTM, Cortez et al. [2] have made remarkable achievements in predicting civil disturbance time. Some of these method's f1-scores are in the range of 0.68 to 0.95 [8]. One limitation of many such studies is that the methods only involve limited features extracted from data, thus cannot establish the extensive correlation between events. Besides, maintaining databases manually or obtaining information from social media is complicated and biased. For example, if we want to obtain the daily flow on the topic of protests on Twitter, we need to collect a large number of tweets, select keywords carefully, then count the number of tweets related to these keywords. In this process, the choice of keywords is so vital that it will directly affect the follow-up prediction result. Furthermore, civil unrest events are a complex process that cannot be fully characterized by collecting indicators in some fields isolated.

In this paper, we utilize graph structure to construct the correlation between historical events and then adopt a sequential model to predict future events. We call our **Gra**phical and **S**equential **Net**work the **GasNet**. To the best of our knowledge, this is the first attempt at civil unrest forecasting that combines representation learning of events graph and time-series forecasting using an RNN-like model. First, we construct our event graph with not only event nodes but also date nodes, event actor nodes, and attribute nodes. Then we use graph neural networks to learn the semantic representation of all those nodes. After that, we organize the data nodes, which serve as the readout nodes, in chronological order to use the convolutional neural networks to extract the features in the temporal dimension. Finally, we input the extracted temporal features into a recurrent neural network to predict future events. To predict the civil unrest events in Hong Kong, we exploit the above model to build a verification experiment. Experimental results show that by making an event prediction three days in advance, our method can achieve a precision rate of 0.75 and a recall rate of 0.88. We also study the impact of lead time and external events through experiments.

2 Related Work

2.1 Event Database

Early event databases were built manually. However, even the biggest human team is incapable of fully reading and analyzing billions of words and images posted every day over the whole world. Fortunately, with the development of event extraction technologies, large-scale automated structured event databases, such as ICEWS [5], NewsReader [16], and EMBERS [11], have gradually been established. As the largest open-access event database, GDELT(Global Database of Events, Language, and Tone) uses complex algorithms combined with deep learning tailored for news articles to create a real-time structured record of global events [9]. Each event in GDELT will be parsed into about 60 fields. In this paper, we mainly use *SQLDATE*, *EventBaseCode*, *NumMentions*, *ActorCountryCode*, *ActorTypeCode*, and *ActionGeo_CountryCode*, which represent the date, event type, number of mentions of the event, participant's country, identity, and event's location, respectively.

2.2 Civil Unrest Forecasting

Events database such as GDELT and social media data such as Twitter data are widely used in event prediction. Qiao et al. [12] use GDELT to build a Hidden Markov Models (HMMs) based framework to discover the development mechanism of potential events and predict indicators associated with country instability. Islam et al. [6] filter tweet stream and classifies tweets using linear Support Vector Machine (SVM) classifier. After distributing weights to tweets, use them to predict civil unrest in a location. In order to predict civil unrest events more accurate, heterogeneous data sources are used. Korkmaz et al. [8] use the input data containing predictors extracted from social media sites (Twitter and blogs), news, and requests for Tor to predict the probability of occurrence of civil unrest events. Logistic regression models with Lasso are used to select a sparse feature set from datasets.

In the work described above, various kinds of traditional machine learning methods are used. In recent years, many deep learning methods have also been utilized in civil unrest forecasting [18]. Chen et al. [1] count the number of various types of events in GDELT and use a LSTM-like model to predict future events that will occur between a specific pair of countries. The experiment only provides a coarse-grained prediction that can indicate the future trend of the relationship between the two countries. Meng et al. [10] use historical data, social media data, and economic indicators as structured data; the embedding of related tweets as unstructured data. They use a model that combines convolutional layers and LSTM layers to learn patterns from various data sources and predict civil unrest events.

These systems only collect indicators of possible related fields in isolation. However, we argue that events are universally correlated. This idea motivates us to study various potential civil unrest drivers by using a graph neural network that learns to represent the input event data via a graph structure.

Besides, it is also essential to be able to predict the occurrence of civil unrest earlier. Muthiah et al. [11] develop EMBERS that can capture significant societal unrest with an average lead time of 4.08 days. In this paper, we discussed the influence of the lead time on the prediction results. We also discussed the influence of external events related to China mainland and other areas on the events in Hong Kong.

2.3 Graph Convolutional Networks

Graph neural networks can be understood as special cases of a simple differentiable message-passing framework [13]:

$$\boldsymbol{h}_i^{(l+1)} = \sigma\Big(\sum_{m \in M_i} g_m(\boldsymbol{h}_i^{(l)}, \boldsymbol{h}_j^{(l)})\Big) \tag{1}$$

where $\boldsymbol{h}_i^{(l)} \in R^d(l)$ is the hidden state of node v_i in the l-th layer of the neural network, with $d^{(l)}$ being the dimensionality of this layer's representations. Incoming messages of the form g_m are accumulated and passed through an element-wise activation function $\sigma()$. M_i represents the set of incoming message for node v_i and is usually means the set of incoming edges of v_i. g_m is message passing function. In Kipf and Welling's work [7], g_m is simply a linear transformation: $g_m(\boldsymbol{h}_i, \boldsymbol{h}_j) = \boldsymbol{W}\boldsymbol{h}_j$ with a weight matrix $\boldsymbol{W}$.

Duvenaud et al. [4] introduce a convolutional neural network that operates directly on graphs, which have satisfactory performance on fingerprint learning and other tasks. Kipf and Welling [7] present a semi-supervised learning method on graph structured data that learns hidden layer representations that encode both local graph structure and features of nodes. Their work achieve significant progress on citation networks and a knowledge graph dataset. Relational Graph Convolutional Networks are developed specifically to deal with the multi-relational data [13]. The record in the event database can be naturally represented as a graph structure with multi relations. Therefore, we employ this model to operate on event graph.

3 Methodology

The goal of this work is to predict whether civil unrest events will occur in a particular region within the next n days, according to the input historical events.

We consider this task as a classification problem. Our model consists of two parts: (i)representation learning and (ii)prediction. We construct historical event data as event graph G, then learn node representations in the event graph through Graph Convolutional Networks. We take the representation of the date node in the event graph as the feature of this date. Through the Graph Convolutional Network, this date node aggregates information about events connected to that node. Assuming that $\boldsymbol{x}_t \in R^n$ represents t day's features, then we organize these nodes' representations into a sequence data in chronological order. $[\boldsymbol{x}_{t-s}, \boldsymbol{x}_{t-s+1}, ..., \boldsymbol{x}_t]$ is the input of the prediction model, and $y_{t+\Delta t}$ is the output. $y_{t+\Delta t} = 0$ or 1, which means whether there would be protests or demonstrations on day $t + \Delta t$. The task of the model is to learn the representations of the date node $\boldsymbol{x}_t$, and then use the learned date representation to predict whether there would be protests or demonstrations event in $t + \Delta t$ day.

The overall structure of our model is shown in Fig. 1. The entire forecasting process can be divided into 5 steps.

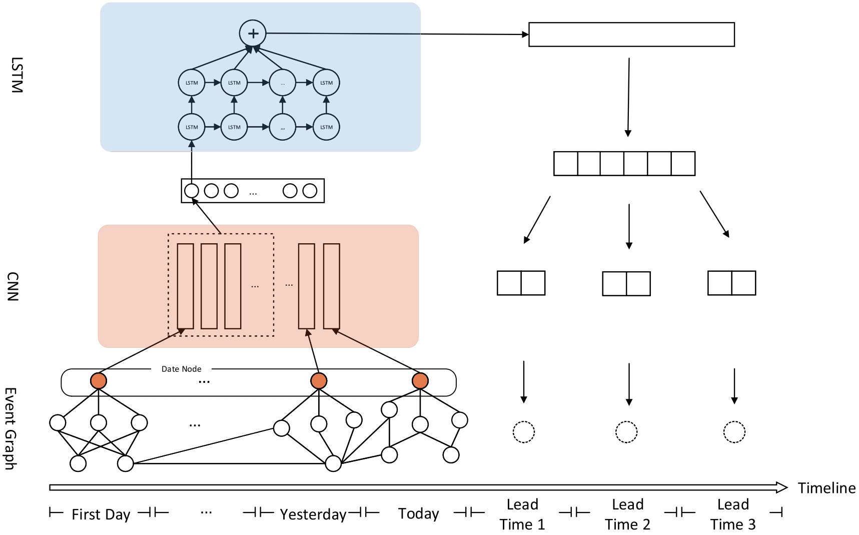


Fig. 1. The overall structure of our Graphical and Sequential Network(GasNet) for civil unrest event forecasting.

Step 1: we collect historical events related to the target area in accordance with specific rules. Since we use GDELT as our dataset, the specific rule is whether the '*ActionGeo_CountryCode*' of the event is target area, which means whether it happened in the area we are concerned about.
Step 2: we structure the historical event data into a graph structure and then train it with the graph neural network.
Step 3: we organize the representation of the date nodes, which serve as the graph's readout, into time-series. Then perform one-dimensional convolution in the time dimension to better extract its temporal features.
Step 4: the output of the convolutional layer is then inputted into a 2-layer LSTM network for sequential learning.
Step 5: finally, we concatenate the output vectors of the LSTM and then produce the prediction results with a linear layer and a softmax layer.

3.1 Construction of the Event Graph

We construct structured event data in GDELT as an event graph, which is the input of the Graph Neural Network. The construction method of the event graph is as follows.

There are three main types of nodes in our event graph: event nodes, attribute nodes, and date nodes. An individual event node represents each event. Event has some attributes. Hence, each event node connects several attribute nodes. In the GDELT event dataset, each event has two participants. Participant has their name, identity, country, and other information. We ignore the participant's

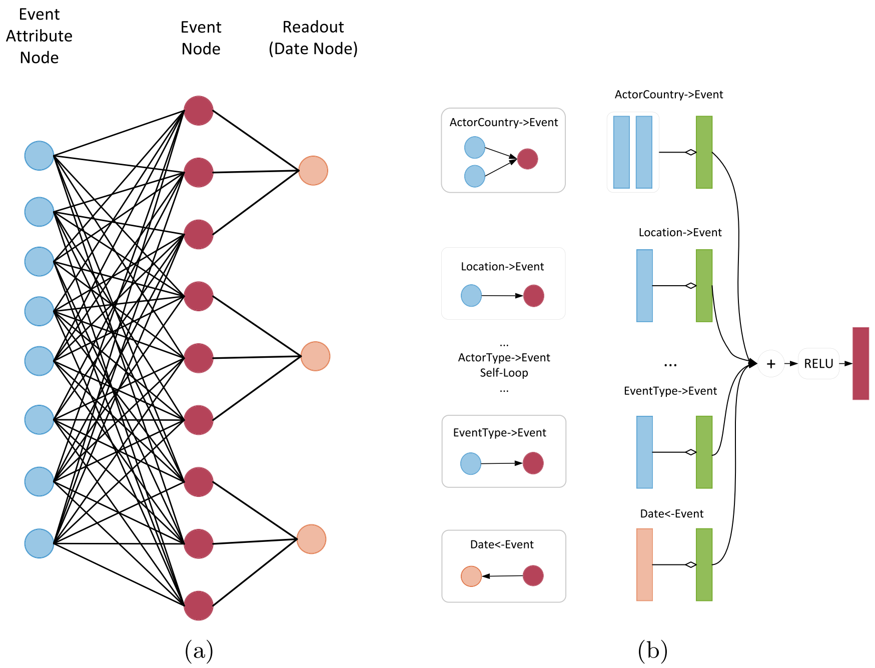


Fig. 2. (a) Construction of the event graph. (b) Update of event node representation. (Color figure online)

specific name, only consider his abstract attributes, especially the 'ActorCountry' and 'ActorType'. In addition to the participants, GDELT also records the event's type and location as 'EventType' and 'Location' attributes. In our event graph, each different value of these attributes also represented by an individual node. Event nodes connect to event attribute nodes by different types of relations. Since there are hundreds of events every day, but types and locations are limited, the number of event nodes is much higher than the number of event attribute nodes. Each event node also connects to the corresponding date node according to the date it occurred. In this way, events indirectly connect by sharing the same date nodes, event type nodes, location nodes, and other attribute nodes. An example of our event graph is shown in Fig. 2a.

3.2 Representation Learning

We employ a Graph Neural Network to learning the representation of the previously constructed event graph. Since there are various directed relations in the graph, we follow the method of the Relational Graph Convolutional Networks(R-GCNs) to perform different linear transformation according to each type of edge.

A graph G is a triplet $G = (V, \xi, R)$ with nodes $v_i \in V$ and directed edges $(v_i, r, v_j) \in \xi$, where $r \in R$ is a relation type. To calculate the forward-pass update of an entity or node denoted by v_i, an equation is defined as follow [13]:

$$\boldsymbol{h}_i^{(l+1)} = \sigma(\sum_{r\in R}\sum_{j\in N_i^r}\frac{1}{c_{i,r}}\boldsymbol{W}_r^{(l)}\boldsymbol{h}_j^{(l)} + \boldsymbol{W}_0^{(l)}\boldsymbol{h}_i^{(l)}) \tag{2}$$

where N_i^r denotes the set of neighbor indices of node i under relation $r \in R$. $c_{i,r}$ is a problem-specific normalization constant that can either be learned or chosen in advance(such as $c_{i,r} = N_i^r$). $\boldsymbol{W}_r^{(l)}$ represents the message passing function of relation r, $\boldsymbol{W}_0^{(l)}$ means self-connection weight.

After neighboring nodes are transformed by the message passing function, which is different depending on the relation type and direction of an edge, this formula accumulates them through a normalized sum. To ensure that the representation of a node at layer $l + 1$ can also be informed by the corresponding representation at layer l, a single self-connection of a special relation type to each node in the data is added.

The computation graph for a single node update in the model is depicted in Fig. 2b. Take an event node's update for example, two 'ActorCountry' nodes (blue) connected to this event node are gathered and then transformed for this particular 'ActorCountry → Event' relation. The resulting representation (green) is accumulated in a normalized sum and passed through an ReLU activation function. The message passing process of other neighbor nodes is similar. This per-node update can be computed in parallel with shared parameters across the whole graph.

3.3 Prediction

Civil unrest forecasting is, in essence, a classification problem over time series data. Suppose we employ the date nodes' representation as the event graph's readout. In that case, the intuitive and straightforward idea is to classify whether a specific event will occur in the next few days according to today's date node's representation. However, we argue that this approach, which performs the forecasting relying entirely on current information, is challenging to train. LSTM could capture long-term dependence on sequence data. Therefore we consider LSTM as a better predictive model in our case. Moreover, to better extract the temporal features, we add a convolutional layer before the LSTM. By applying multiple convolution kernels, the convolutional layer could produce a feature vector for the historical event information over a short period, a week, for example. These feature vectors are then fed into the LSTM to capture the long-term temporal patterns. At last, a fully connected layer takes the output of LSTM. It predicts the probabilities of the future civil unrest event that will occur at a particular date, which is our model's final output.

We organize the representation of date nodes learned by graph neural networks into a time series form: $\boldsymbol{X} = [\boldsymbol{x}_{t-s}, \boldsymbol{x}_{t-s+1}, ..., \boldsymbol{x}_t]$, then a one-dimensional convolutional neural network with a kernel size of 7 is used to extract its temporal feature as follows:

$$\boldsymbol{h}_1 = \boldsymbol{W} \otimes \boldsymbol{X} + b_1 \tag{3}$$

The output is then fed into LSTM.

$$\boldsymbol{h}_2 = LSTM(\boldsymbol{h}_1) \tag{4}$$

Final probabilities of civil unrest event in the subsequent Δt days is calculated as:

$$y_{t+\Delta t} = softmax(\boldsymbol{W}\boldsymbol{h}_2 + b_2) \tag{5}$$

In training process, our model is trained to minimize the cross-entropy loss function.

4 Experiments

As a result of some political, economic, and legal challenges, demonstrations have been held in Hong Kong over the past two years, which have significantly impacted social stability and people's lives. Hence, we chose the civil unrest events occurring in Hong Kong as our prediction target.

We use event data about Hong Kong in the GDELT as our training and testing dataset as described in Sect. 3. There are 20 primary categories and more than 100 subcategories of events in the GDELT database. The primary category 14 (protest), for example, is divided into six subcategories, 141–146. Subcategory 141 stands for Demonstrate or Rally, which is the social unrest event that we are targeting. Small-scale demonstrations take place almost every day in Hong Kong. However, some of these events have only a few participants and minor impacts on society. For this reason, we divide social unrest events into major and minor according to their influence and only target the major ones. GDELT provides "NumMentions" as the total number of event mentions across all source media in the first 15 min of the initial report. We use it to assess the importance of an event: the more mentions of an event, the more likely it is to be significant. In experiments, we regard events with "NumMentions" more than or equal to 10 as major events and events with "NumMentions" less than 10 as minor events. The training set includes historical events from 2015-02-18 to 2020, the train set size is about 1700. The last 300 days of year 2020 is used as the test set.

For comparison, we implement an LSTM model as the baseline model. Without the event graph, we could only count each subcategory's events on each date and use them as input to predict the probability of future events. For the ablation study, we also implement two other models, which are the CNN-LSTM model and the GCN-LSTM model. The CNN-LSTM model is similar to the implementation in [10]. It uses the same input of the base LSTM model but employs a CNN to extract the local temporal features before feeding them into the LSTM to perform the prediction. Comparing the prediction results of the CNN-LSTM model with those of our GasNet will illustrate the event graph's effectiveness. On the other hand, to show the effectiveness of the convolutional layer in the

GasNet, an GCN-LSTM model has also been implemented. It bypasses the convolutional layer and feeds the GCN's readout directly into the LSTM.

All these models are implemented based on pytorch and DGL library. In graph neural networks, we use 200 dimensional vector to represent node. The hidden size of LSTM layer is 256, and the input sequence length is 20. After that, we use a convolving kernel of size 7 to extract features. As for the imbalance of the training data, we use cross-entropy loss with weight to counter.

5 Results and Analysis

5.1 Results

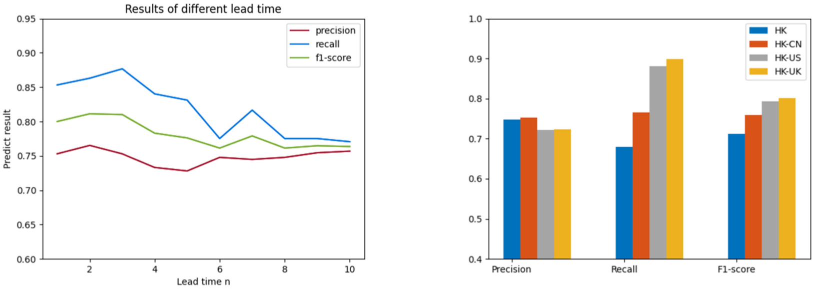
Table 1 shows the performance for various models. The lead time is defined as the number of days between the date that prediction is generated and the date that predicts. The baseline model could achieve about 0.72 F1-score. The CNN-LSTM model uses an additional convolution layer to extract temporal features, which significantly improves the prediction performance by about five percent. To our surprise, the GCN-LSTM model performs even worse than the baseline model. It achieves the best precision as well as the worst recall. We argue that the GCN does extract some strong indicators from the correlation of the historical events. However, the GCN's outputs, which are highly abstractive graph representations, are possibly too complicated for LSTM to perform the time series prediction directly. Hence, we add a convolution layer between the GCN's outputs and the LSTM's inputs. By performing the convolution operation on multiple inputted date representations, the convolution layer could extract the inputs' temporal features, which could also be regarded as a simplification of the input information. Finally, our GasNet achieved about 0.81 F1-score, nine percent higher than the baseline model.

Table 1. The civil unrest event forecasting performances of different models.

Models	Lead time = 1			Lead time = 2			Lead time = 3		
	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
Basic LSTM	0.75	0.70	0.73	0.75	0.70	0.73	0.75	0.68	0.71
CNN-LSTM	0.76	0.80	0.78	0.78	0.75	0.77	0.75	0.77	0.76
GCN-LSTM	**0.81**	0.62	0.70	**0.83**	0.63	0.72	**0.82**	0.63	0.71
GasNet(ours)	0.75	**0.85**	**0.80**	0.77	**0.86**	**0.81**	0.75	**0.88**	**0.81**

5.2 The Influence of Lead Time

Predicting serious civil unrest events and providing early warning will help the public sector take timely measures to maintain social order. It will also help



(a)

(b)

Fig. 3. (a) The influence of different lead time on prediction results. (b) The influence of external events on prediction of events in Hong Kong.

residents and passengers in the affected area to avoid risks and protect their lives and property. Therefore, we want to perform the prediction as early as possible. However, as a commonsense, the earlier the forecast does, the lower the accuracy will be. Hence, we extend the forecast lead time to ten days to illustrate how the forecast accuracy changes accordingly.

As can be seen from the Fig. 3a, generally speaking, with the extension of the lead time, the prediction results become worse, which is consistent with our expectation. However, when the lead time is 7, there is a significant peak in recall. We think this is due to the apparent periodicity of people's behavior, especially the collective activities that need to be planned, such as rallies and protests. Seven days, which equals one week, is the smallest periodic in people's daily life that our model could capture. Hence, our model could produce better prediction results seven days in advance, rather than six or eight days.

5.3 The Influence of External Events

Hong Kong, which was once a British colony from 1842 to 1997, and a special administrative region of China since 1997, has become one of the world's most significant financial centers and commercial ports. Therefore, we want to explore whether including related external events will improve Hong Kong's civil unrest events prediction performance. We choose three countries and regions closely related to Hong Kong: Mainland China, the United Kingdom, and the United States. We collect events related to Hong Kong in these areas in GDELT according to whether the two participants involved in the event belong to Hong Kong and the target area respectively. The specific number of events is shown in Table 2. At last, we conduct experiments based on these events. The prediction results are shown in Fig. 3b.

The results show that the external events can significantly increase the recall of civil unrest events prediction in Hong Kong. However, it may also slightly decrease its precision at the same time. The reason for this results can be explained: More events bring more clues, which leads to a higher recall rate.

Table 2. Number of Hong Kong related events in different countries and regions.

Location	Events related to Hong Kong
Hong Kong	681329
United Kingdom	65685
United States	99660
Mainland China	292146

Meanwhile, more events bring more complexity, which leads to a lower precision rate. However, the UK has the fewest related events with Hong Kong, but the increase of the recall rate is the highest. The reasons for this result may be complicated, but we can reasonably assume that after the end of the colonial period, UK and HK still maintain a very close relationship. Further discussion of the political and economic reasons behind this finding is far beyond the scope of this paper. Although the prediction performance is greatly improved, it needs enormous extra work to obtain and process external events. Furthermore, the limitation of computing capability also makes it impracticable to add more external events without restriction.

6 Conclusions

In this paper, we propose a model that combines graphical and sequential neural networks for civil unrest forecasting. We utilize a graph network to construct the correlation between historical events, a convolutional layer to extract temporal features, and an LSTM layer to predict future events. Our experiments show that the recall of Hong Kong civil unrest event forecasting is higher than 0.85, which illustrated that our model could produce reliable prediction results. We also explored the impact of the lead time and the external events on the prediction results. Future work should explore forecasting other types of events as well as targeting other countries or districts. To further improve our model's performance, we should also consider integrating the latest research results, such as the Transformer [14] and the Graph Attention Network [15].

References

1. Chen, P., Jatowt, A., Yoshikawa, M.: Towards future event prediction using graph-LSTM
2. Cortez, B., Carrera, B., Kim, Y.J., Jung, J.Y.: An architecture for emergency event prediction using LSTM recurrent neural networks. Expert Syst. Appl. **97**, 315–324 (2018)
3. Dermisi, S., et al.: Social media, civil unrest and fallout for cities and hotels. Technical report. European Real Estate Society (ERES) (2017)
4. Duvenaud, D., et al.: Convolutional networks on graphs for learning molecular fingerprints. In: Advances in Neural Information Processing System 2015, pp. 2224–2232 (2015)

5. Gerner, D.J., Schrodt, P.A., Francisco, R.A., Weddle, J.L.: Machine coding of event data using regional and international sources. Int. Stud. Quart. **38**(1), 91–119 (1994)
6. Islam, M.K., Ahmed, M.M., Zamli, K.Z., Mehbub, S.: An online framework for civil unrest prediction using tweet stream based on tweet weight and event diffusion. J. Inf. Commun. Technol. **19**(1), 65–101 (2019)
7. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 24–26 April 2017, Conference Track Proceedings. OpenReview.net (2017). https://openreview.net/forum?id=SJU4ayYgl
8. Korkmaz, G., Cadena, J., Kuhlman, C.J., Marathe, A., Vullikanti, A., Ramakrishnan, N.: Combining heterogeneous data sources for civil unrest forecasting. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, pp. 258–265 (2015)
9. Leetaru, K., Schrodt, P.A.: GDELT: global data on events, location, and tone, 1979–2012. In: ISA Annual Convention, vol. 2, pp. 1–49. Citeseer (2013)
10. Meng, L., Srihari, R.K.: Increasing lead time and granularity of civil unrest prediction through time series data (2019)
11. Muthiah, S., et al.: Planned protest modeling in news and social media. In: Twenty-Seventh IAAI Conference. Citeseer (2015)
12. Qiao, F., Li, P., Zhang, X., Ding, Z., Cheng, J., Wang, H.: Predicting social unrest events with hidden Markov models using GDELT. In: Discrete Dynamics in Nature and Society 2017 (2017)
13. Schlichtkrull, M., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: Gangemi, A., et al. (eds.) ESWC 2018. LNCS, vol. 10843, pp. 593–607. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_38
14. Vaswani, A., et al.: Attention is all you need, pp. 6000–6010 (2017)
15. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: International Conference on Learning Representations (2018)
16. Vossen, P., et al.: Newsreader: using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. Knowl.-Based Syst. **110**, 60–85 (2016)
17. Zhao, L.: Event prediction in the big data era: a systematic survey. ACM Comput. Surv. 54(5) (2021). https://doi.org/10.1145/3450287
18. Zhu, C., Chen, M., Fan, C., Cheng, G., Zhan, Y.: Learning from history: modeling temporal knowledge graphs with sequential copy-generation networks. In: AAAI (2021)