Tutorial: Advances in Human Event Modeling: From Graph Neural Networks to Language Models

Songgaojun Deng, Maarten de Rijke, and Yue Ning

Sunday, August 25 2024
10:00 AM – 1:00 PM (CEST)

KDD2024
BARCELONA, SPAIN

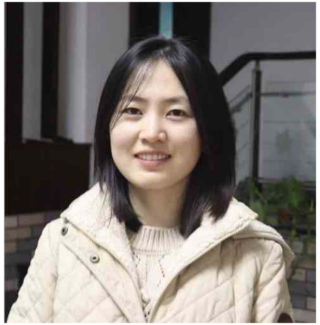
UNIVERSITY OF AMSTERDAM

STEVENS INSTITUTE OF TECHNOLOGY 1870

Acknowledgements

Tutorial based in part on materials in the Tutorial at AAAI 2021: *Explainable AI for Societal Event Predictions: Foundations, Methods, and Applications* and a number of published papers.

Organizers



Songgaojun Deng
Postdoc Researcher
University of Amsterdam
(Presenter)
ML and data mining in social science, health and e-commerce

Maarten de Rijke
Distinguished Professor
University of Amsterdam
(Contributor)
Information retrieval and machine learning

Yue Ning
Assistant Professor
Stevens Institute of Technology
(Presenter)
Socially Responsible AI

Objectives

- Gain a solid understanding of human event modeling and its significance.
- Learn about the latest advancements in graph neural networks (GNNs) and large language models (LLMs).
- Recognize open challenges and opportunities for further exploration.
- Engage in discussions to exchange ideas with peers.

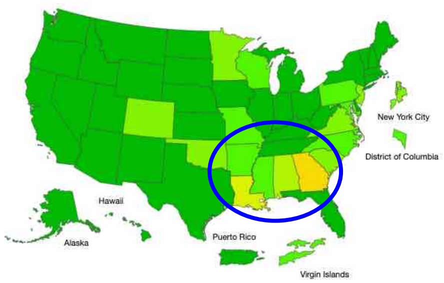
Roadmap

- Introduction and motivation
- Methodology
  - Part 1: Graph Neural Network (GNN)-based methods
  - Part 2: Large Language Model (LLM)-based methods
- Conclusion and future directions

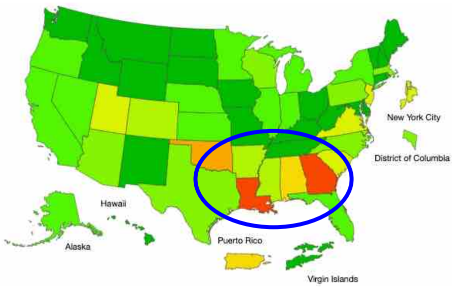
Introduction and motivation

Human events

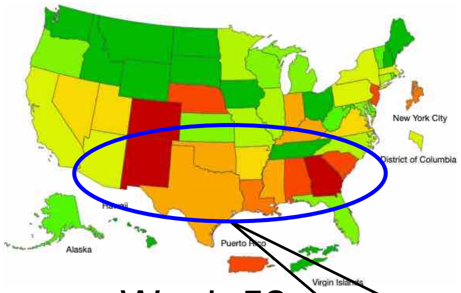
Epidemic outbreak during 2018-2019 in southern region



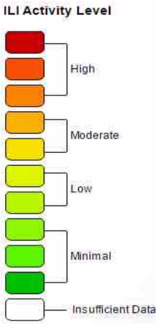
Week 44



Week 47



Week 52



influenza

Terrorism events

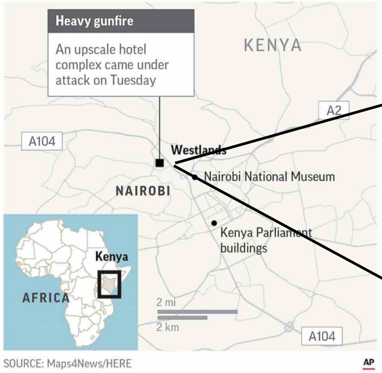
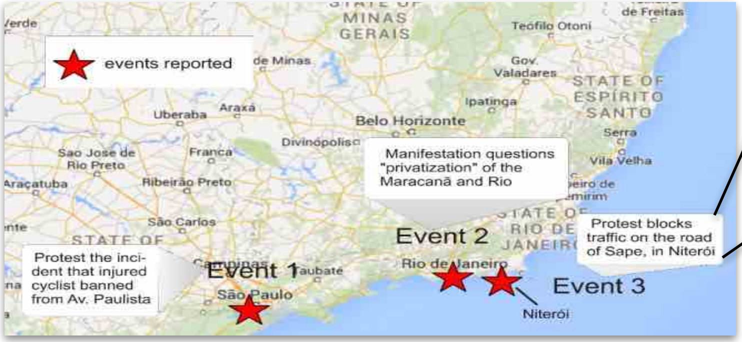


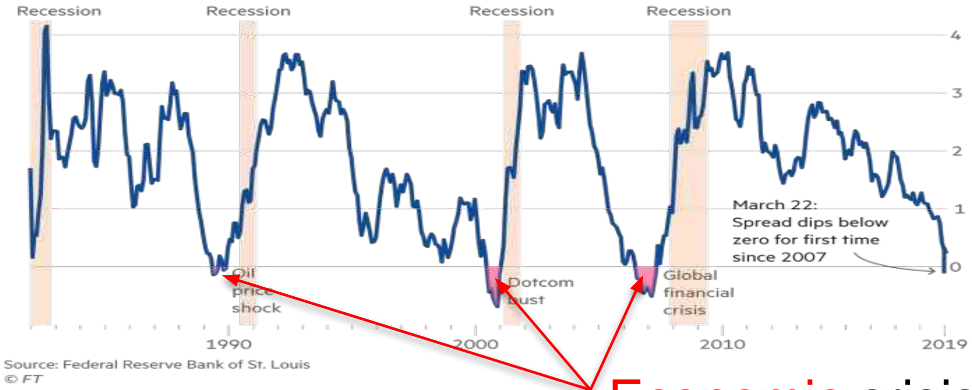
Figure credit: [ABC news]

Human events



Protests

Civil unrest events on Mar 17, 2013 in Brazil



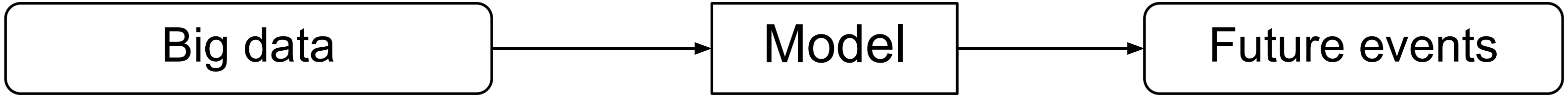
Economic crisis

Figure credit: [Medium]

Human events

Event forecasting

- The task is to predict the occurrence of events in the future using historical data.
- Underlying mechanism of societal events
  - Complex, dynamic, sparse
  - Hard to comprehensively model with limited data
  - Largely unknown

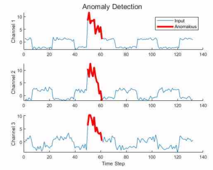


Build the forecaster driven by large historical data

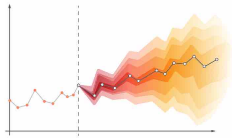
Comparison of event modeling tasks

- Extraction
  - Given a piece of text, extract key elements such as location and entities.
  - Supervised/Semi-supervised learning
  - **Techniques**: natural language processing

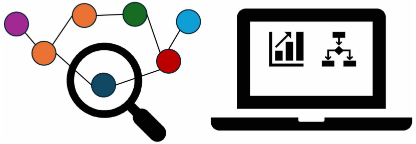
- Detection
  - Given historical or ongoing events, detect anomaly
  - Unsupervised learning
  - **Techniques**: anomaly detection, outlier detection, change detection, motif discovery



- **Forecasting** (or prediction/projection)
  - Given historical data, predict event occurrences <u>in the future</u>
  - Supervised/self-supervised/semi-supervised learning
  - **Techniques**: autoregressive, Markov chain, sequential models, neural networks



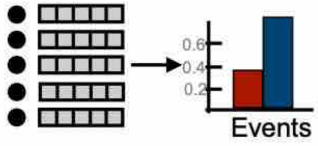
- **Interpretation**
  - Given a prediction, provide explanation or evidence
  - **Techniques**: gradient-based methods, multi-instance learning, attention, knowledge distillation, causal inference



Problem formulation

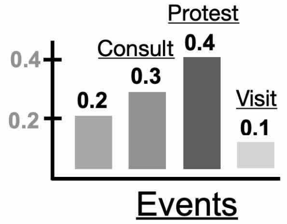
Binary event prediction

- Given static and dynamic input features, learn a classifier that maps the input to a **binary event variable** (e.g., protest or not) at a future time for a target location .



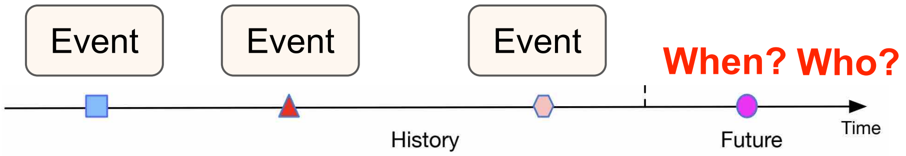
Multi/Concurrent event prediction

- Events can occur concurrently (e.g., Appeal for judicial cooperation, accuse of crime).
- Learn a classifier that maps the input to a **multi-hot vector denoting the occurrences of different event types.**



More

- **Time** prediction
- **Actor** prediction



Lead time

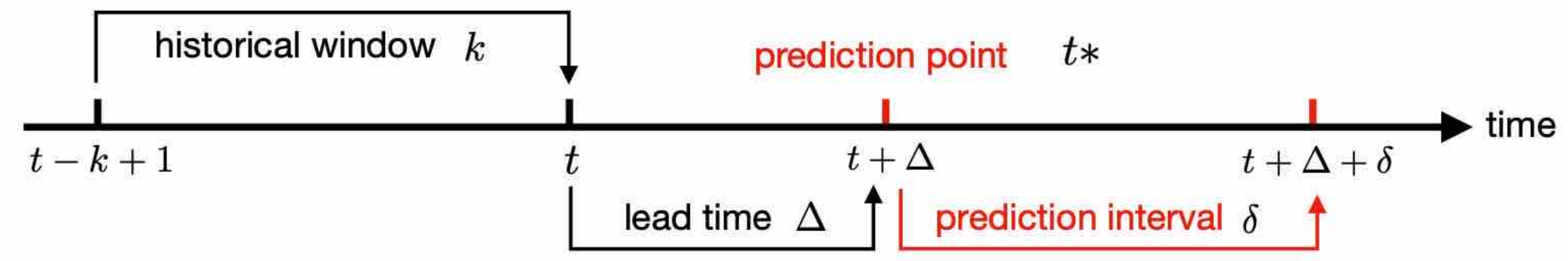
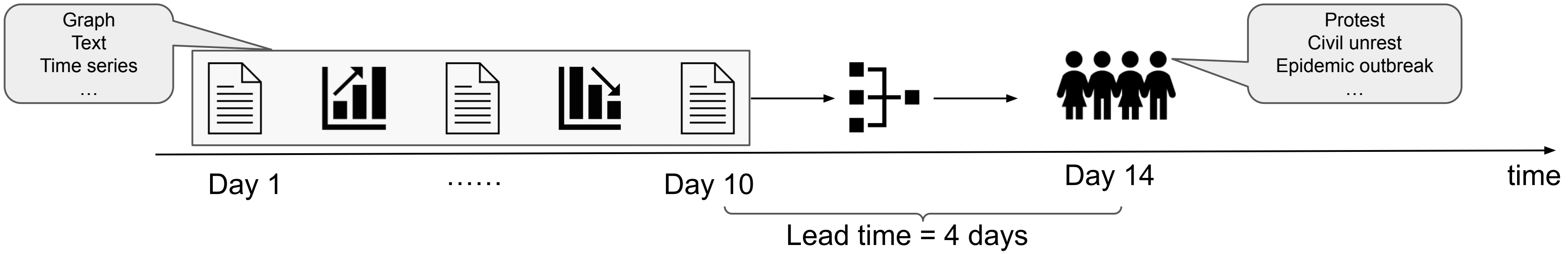
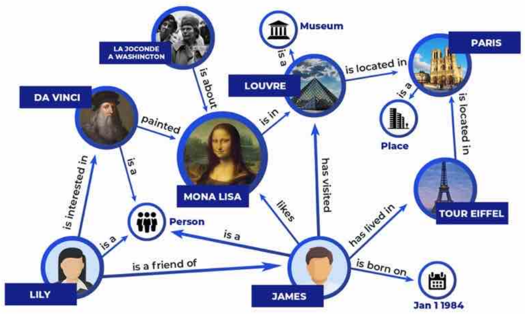


Figure: Example of lead time and prediction time



Human event data

Graph



Text


The New York Times
MEN WALK ON MOON
ASTRONAUTS LAND ON PLAIN; COLLECT ROCKS, PLANT FLAG
Voice From Moon: 'Eagle Has Landed'
A Powdery Surface Is Closely Explored

Time Series

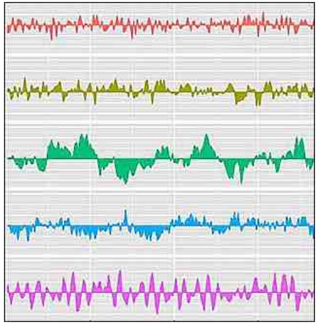
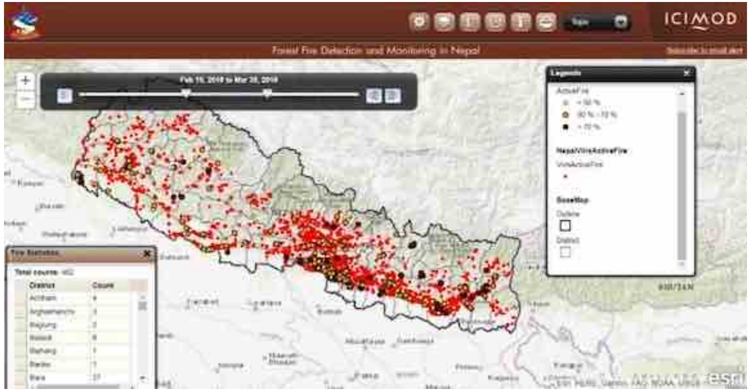
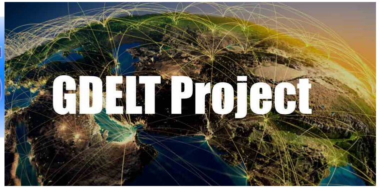


Figure credits:
[Let the Machines Learn](); [New York Times](); [MIT OpenCourseWare]()

Examples of event data



[Forest Fire Detection and monitoring in Nepal]

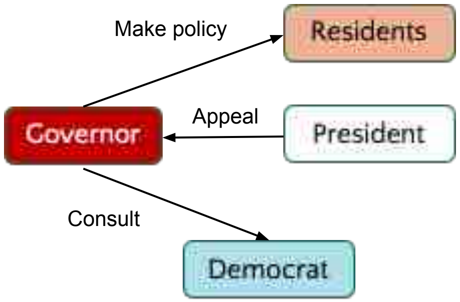


Global event encoding system

Characteristics of social indicators

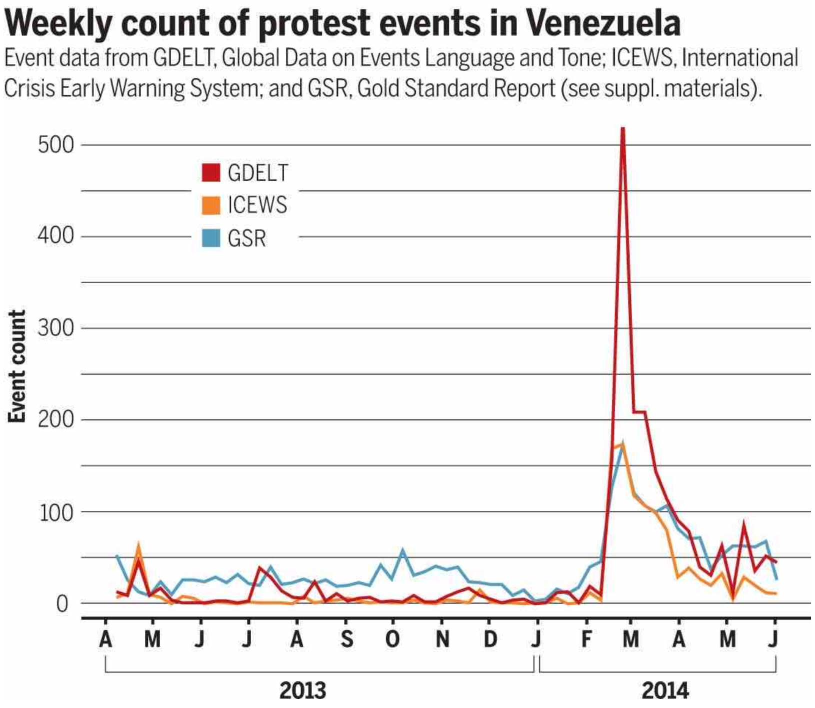
- Social Media
  - Ubiquitous
    - Every user/agent of social media/web/forum is a social sensor (citizen sensor)
    - They are everywhere observing the world all the time.
  - Timeliness
    - 6,000 tweets every second.
    - 500 million tweets per day.
    - Usually beats the earliest official reports.

- Event encoding systems
  - Annotations
    - Each event is labeled with subject, object, type,and time, formulating a graph
  - Comprehensive coverage
  - Less prompt



Challenges in human event prediction

Multi-source unstructured data

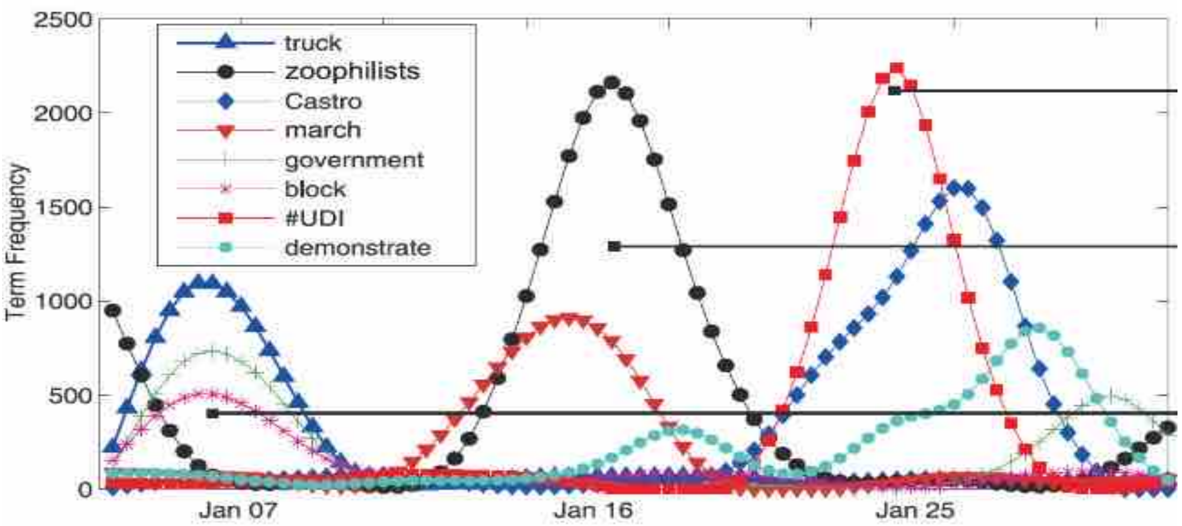


Source: Wei Wang, Ryan Kennedy, David Lazer, Naren Ramakrishnan. Growing pains for global monitoring of societal events. Science 30 Sep 2016. Vol. 353, Issue 6307, pp. 1502-1503

Challenges in human event prediction

Dynamic and rapid changes over time

New words, entities, relations and shift of topics

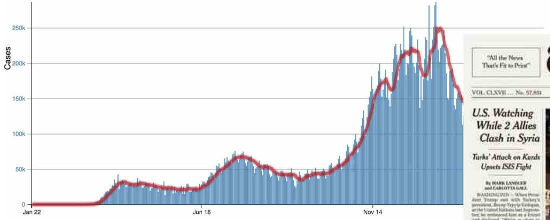


Source: Spatio-Temporal Event Forecasting and Precursor Identification. Tutorial in 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2019)

Challenges in human event prediction

Heterogeneous data

Need to align data samples with different time granularity




"All the News That's Fit to Print"

The New York Times

Late Edition

VOL. CLXVII ... No. 57,931

NEW YORK, TUESDAY, JANUARY 23, 2018

$3.00

U.S. Watching While 2 Allies Clash in Syria

Turks' Attack on Kurds Upsets ISIS Fight

SHUTDOWN ENDS, SETTING UP CLASH OVER 'DREAMERS'

NEWS ANALYSIS

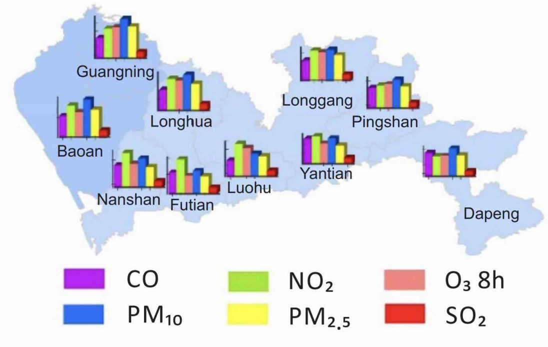
Hope for Bargain in a Swift Surrender

Immigration Issues Unresolved Under Senate Accord

Tax-Law Bonuses Veil a Bonanza for Companies

Other challenges

Dependencies among events, e.g., spatial dependencies



Lack of interpretability – benefit for decision making

Model —predict→ Future events

Why?

Explainable event prediction

- Social indicators can be general signals, features, and even distributions in open source data sets.
- Precursor discovery refers to identifying specific examples or instances in the historical data given a prediction.
- Explainable predictive models uncover significant features, graphs, documents for explaining prediction results.

An example event

2014 Venezuelan National Students Protest

major protests began with student marches led by opposition leaders in 38 cities.

Feb. 12

2014 Venezuelan National Students Protest

Opposition Leader, López, called upon students to peacefully protest.

major protests began with student marches led by opposition leaders in 38 cities.

Feb. 1

Feb. 12

2014 Venezuelan National Students Protest

López, alongside María Corina Machado launched a campaign to remove Maduro from office.

Jan. 23

Opposition Leader, López, called upon students to peacefully protest.

Feb. 1

major protests began with student marches led by opposition leaders in 38 cities.

Feb. 12

2014 Venezuelan National Students Protest

Murder of former Miss Venezuela, Monica Spear.

Former presidential candidate Henrique Capriles shook the hand of President Maduro

Attempted rape of a young student on a university campus in San Cristóbal

The harsh police response to their initial protest

January

López, alongside María Corina Machado launched a campaign to remove Maduro from office.

Jan. 23

Opposition Leader, López, called upon students to peacefully protest.

Feb. 1

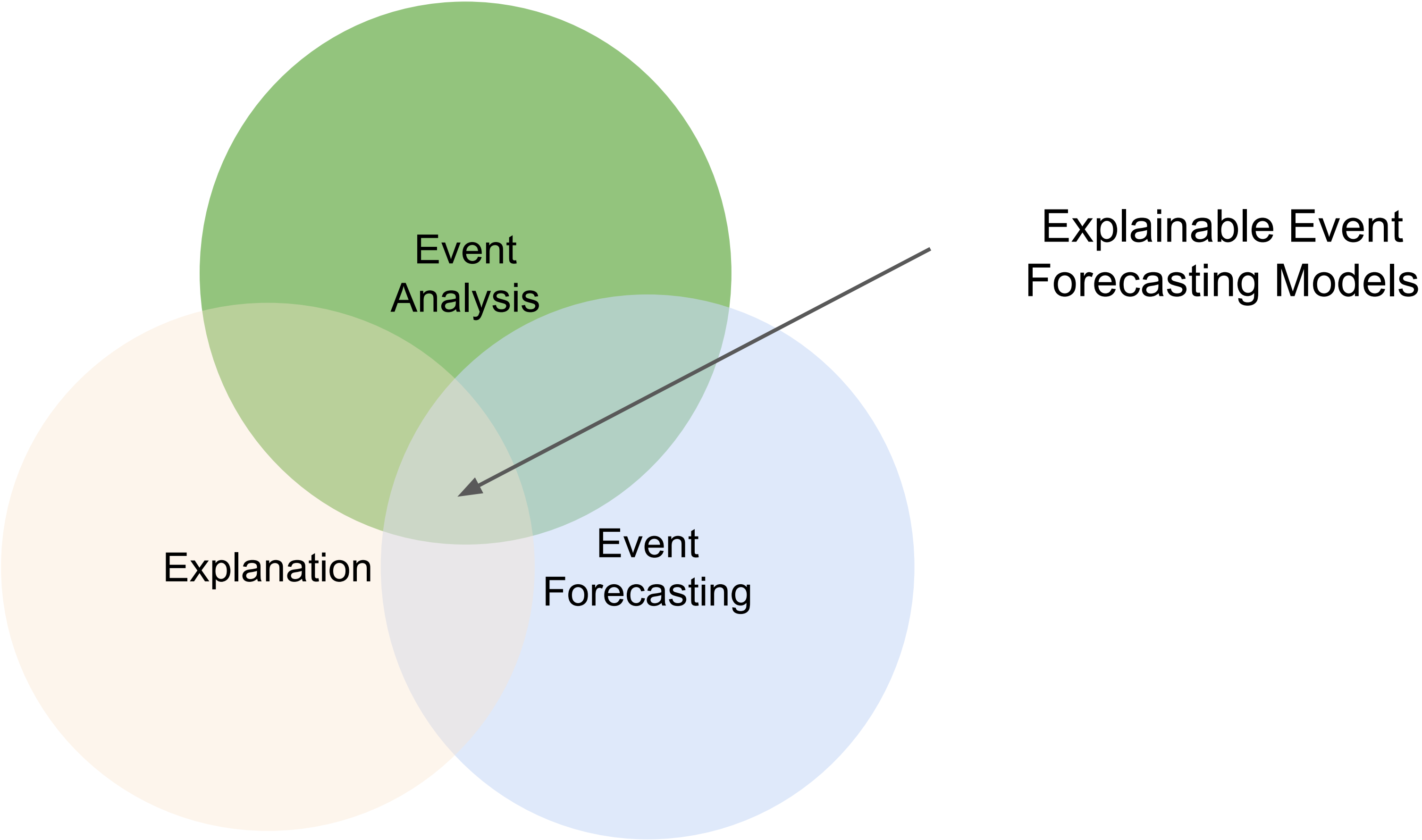
major protests began with student marches led by opposition leaders in 38 cities.

Feb. 12

If social scientists need to do this a lot ……

http://phdcomics.com/

Overview


Event
Analysis
Explanation
Event
Forecasting
Explainable Event
Forecasting Models

Early approaches

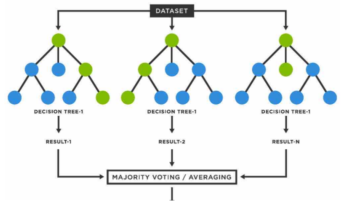
Statistical methods

- A probabilistic model [Radinsky et al. 2013]
  - Assume that events in the real-world are generated by a probabilistic model that also generates news reports corresponding to these events.
- A threshold-based approach [Manrique et al. 2013]
  - The model forecasts civil unrest relying on a search in volume of event-related terms and their momenta.
- Hidden markov model [Qiao et al. 2017]
  - A HMMs-based social unrest (SU) event prediction framework: two HMMs are trained, with one for SU-prone sequences and one for SU-free sequences.

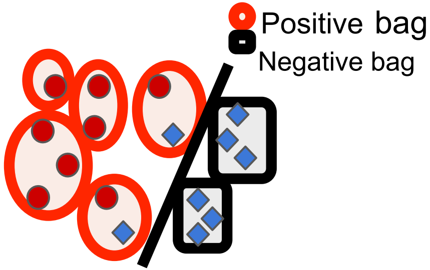
Radinsky, Kira, and Eric Horvitz. **Mining the web to predict future events**. *WSDM*. 2013.
Manrique, Pedro, et al. **Context matters: Improving the uses of big data for forecasting civil unrest: Emerging phenomena and big data**. *IEEE International Conference on Intelligence and Security Informatics*. 2013.
Qiao, Fengcai, et al. **Predicting social unrest events with hidden Markov models using GDELT**. *Discrete Dynamics in Nature and Society* 2017.1.

Machine learning methods

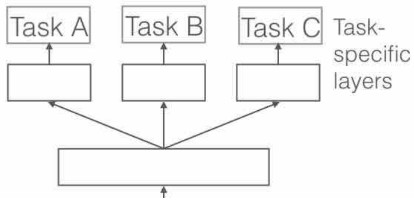
- Random Forest [Kallus 2014]
  - Quantify the predictive signals/features



- Multi-Instance Learning [Ning et al. 2016]
  - Incomplete knowledge about labels in training data; Propagate bag level supervision to individual document



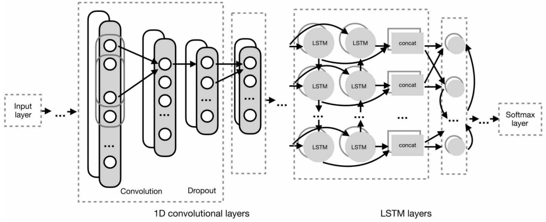
- Multi-task learning [Ning et al. 2018]
  - Explicitly enforces pairs of cities with similar event patterns in the past to learn similar vectors.



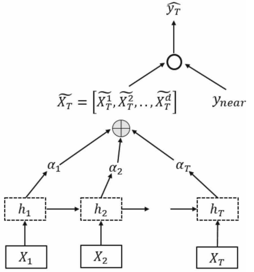
Kallus, Nathan. **Predicting crowd behavior with big public data**. World Wide Web. 2014.
Ning, Yue, et al. **Modeling precursors for event forecasting via nested multi-instance learning**. KDD 2016.
Ning, Yue, et al. **STAPLE: Spatio-temporal precursor learning for event forecasting**. *SDM* 2018.

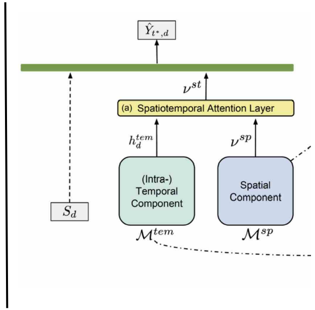
Early deep learning methods

- Recurrent neural networks (RNNs) [Meng et al. 2019]
  - Use recurrent neural networks (RNN, LSTM) to capture temporal information of event signals to predict the civil unrest events.



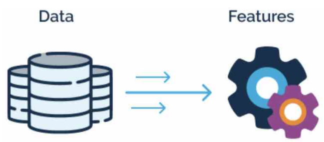
- Attention models [Wang et al. 2018, Ertugrul et al. 2019]
  - Use attention methods to study different contributions of data points in the time series for predicting civil unrest and explaining feature importance.





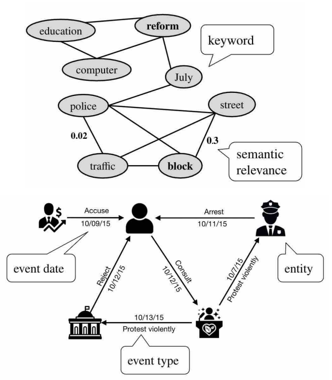
Meng, Lu, et al **Leveraging heterogeneous data sources for civil unrest prediction**. *Social, Cultural, and Behavioral Modeling* (2019).
Wang, Xiuling, et al. **Unrest news amount prediction with context-aware attention lstm**. *PRICAI 2018*
Ertugrul, Ali Mert, et al. **Activism via attention: interpretable spatiotemporal learning to forecast protest activities**. *EPJ Data Science* 8.1 (2019).

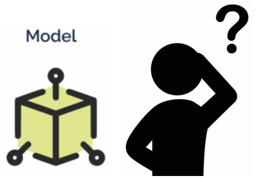
Limitations of these works



Reliance on feature selection and engineering

Limitations of using grid-like or homogeneous data





Limited model ability to understand predicted events

Limited causal analysis in event prediction

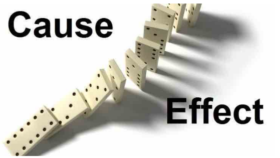
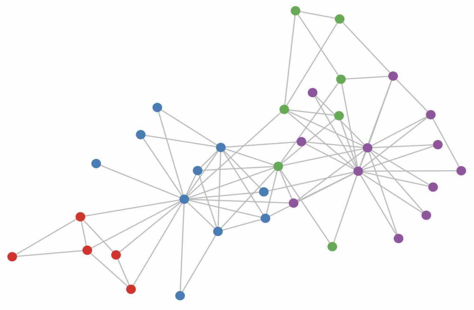


Image credits: https://splashbi.com/what-is-feature-engineering/

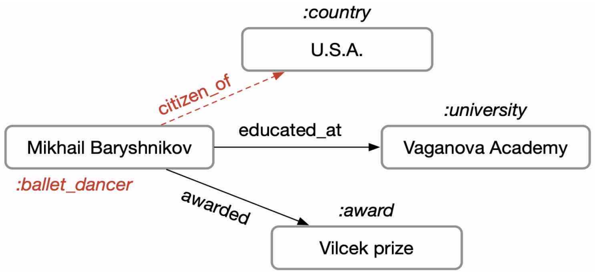
Part 1: Graph Neural Network (GNN)-based methods

Graph Neural Networks (GNNs)

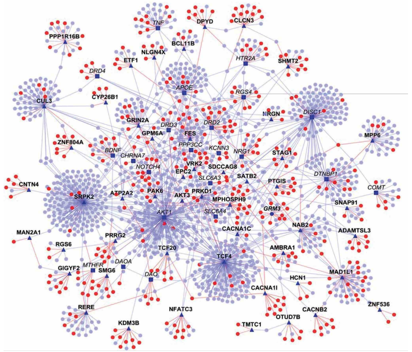
GNNs are a class of neural networks that operate on graph-structured data. GNNs are powerful for graph representation learning in diverse domains.



Social Networks



Knowledge Graphs



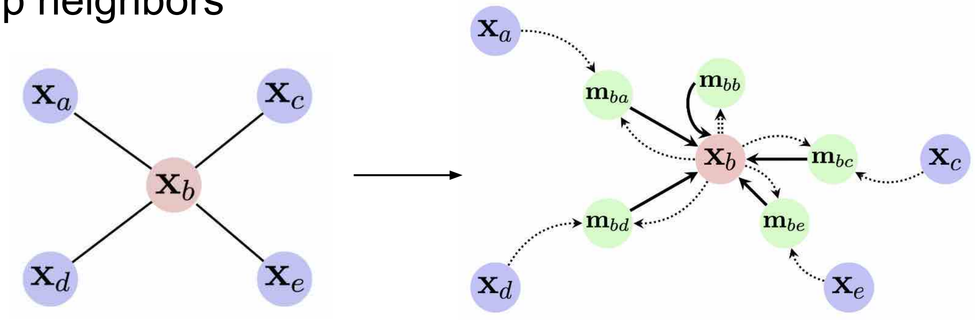
Protein Interaction Networks

Standard CNN and RNN architectures cannot work effectively on graph-structured data.

How do GNNs work?

A message-passing-framework perspective

- Node representation updates by aggregating messages sent from its neighbors
- K-layer GNNs take into account K-hop neighbors

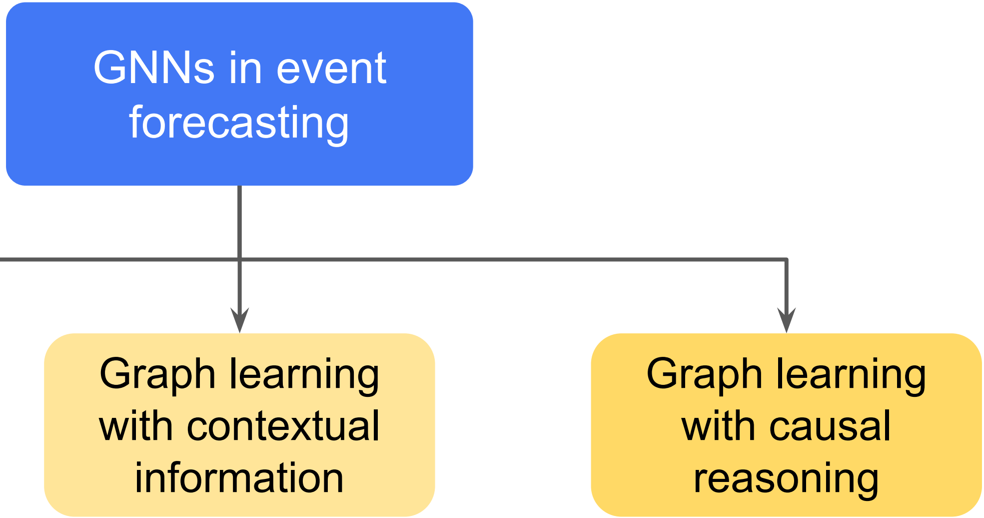


Representative GNN models

- Graph Convolutional Network (GCN)
- Graph Attention Network (GAT)
- Relational Graph Convolutional Networks (R-GCNs)

Kipf, Thomas N., and Max Welling. **Semi-supervised classification with graph convolutional networks**. *arXiv:1609.02907* (2016).
Veličković, Petar, et al. **Graph attention networks**. arXiv preprint arXiv:1710.10903 (2017).
Schlichtkrull, Michael, et al. **Modeling relational data with graph convolutional networks**."ESWC 2018.

GNNs in event forecasting

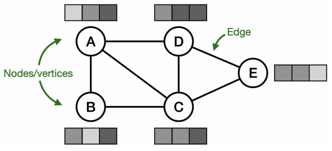


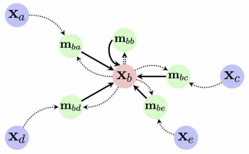
Vanilla graph learning

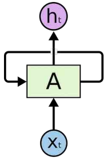
Vanilla graph learning

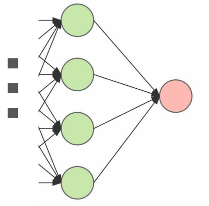
Concentrate on graph operations, which leverage node and/or edge embeddings techniques.

Graph construction → Graph learning → Temporal dependency learning → Output prediction





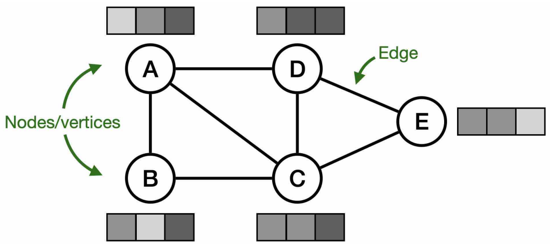




Graph construction

Define graph structure

- Historical event data $(G_{t-k+1}, \ldots, G_t)$ where k is the sequence length.
- $G_t = (V_t, E_t)$ is the graph at time t with the node set *V* and edge set *E*.



Define features for graph elements

- Word embeddings for node features at time t denoted as $\mathbf{X}_t^{N \times d}$, where *N* is the number of words and d is the feature dimension.

Graph learning

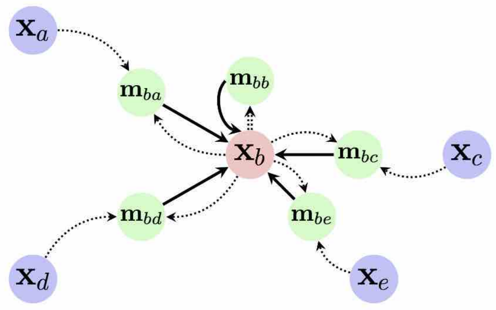
Update node and/or edge representations based on information from neighboring nodes.

Updated representation of node i

Neighbor information extractor

$$\mathbf{H}_t^{(l)}[i] \leftarrow \underset{\forall j \in \mathcal{N}_t(i), \forall e \in E_t(j,i))}{\mathrm{AGG}} \left(\mathrm{EX}\left(\mathbf{H}_t^{(l-1)}[i]; \mathbf{H}_t^{(l-1)}[j], e\right)\right)$$

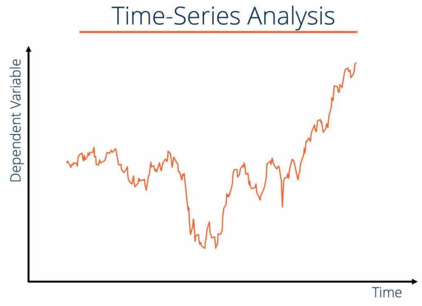
Gathers the neighborhood information of source nodes, e.g., mean, max.

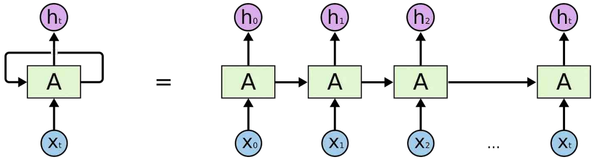


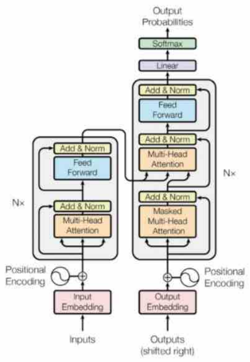
Temporal dependency learning

This step involves capturing **temporal dependencies** from past events to effectively forecast future events.

Techniques commonly employed include time series analysis methods, recurrent neural networks, attention mechanisms, or customized temporal methods.

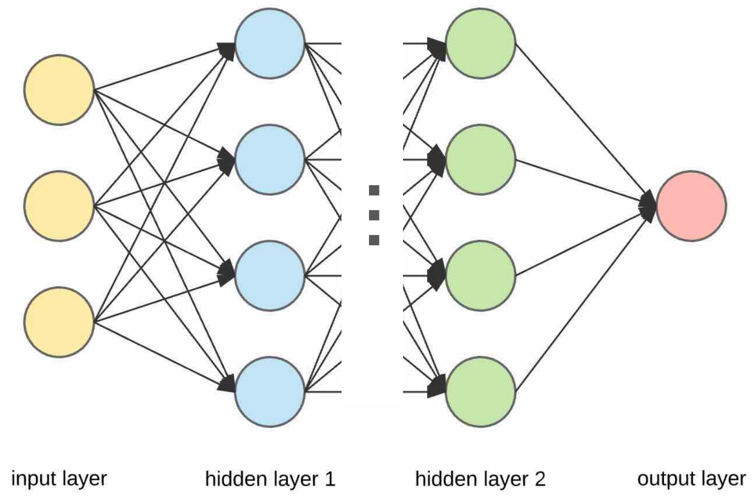






Output prediction

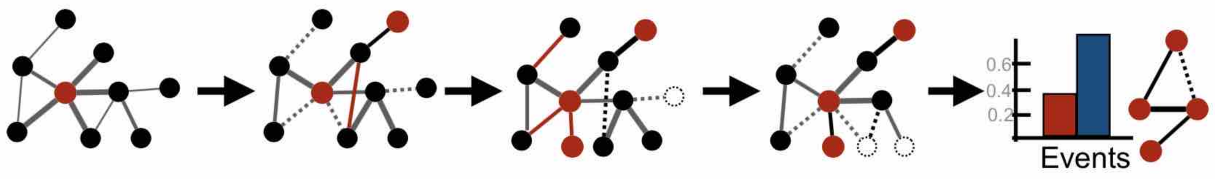
Use the learned node or graph representations for downstream tasks, such as predicting **binary events** or **concurrent** events, by applying an appropriate output layer.



E.g., applying sigmoid to predict the binary occurrence of events.

Learning Dynamic Context Graphs for Predicting Social Events [Deng et al. KDD19]

Propose a novel **graph-based model** for predicting events and identifying relevant event subgraphs.

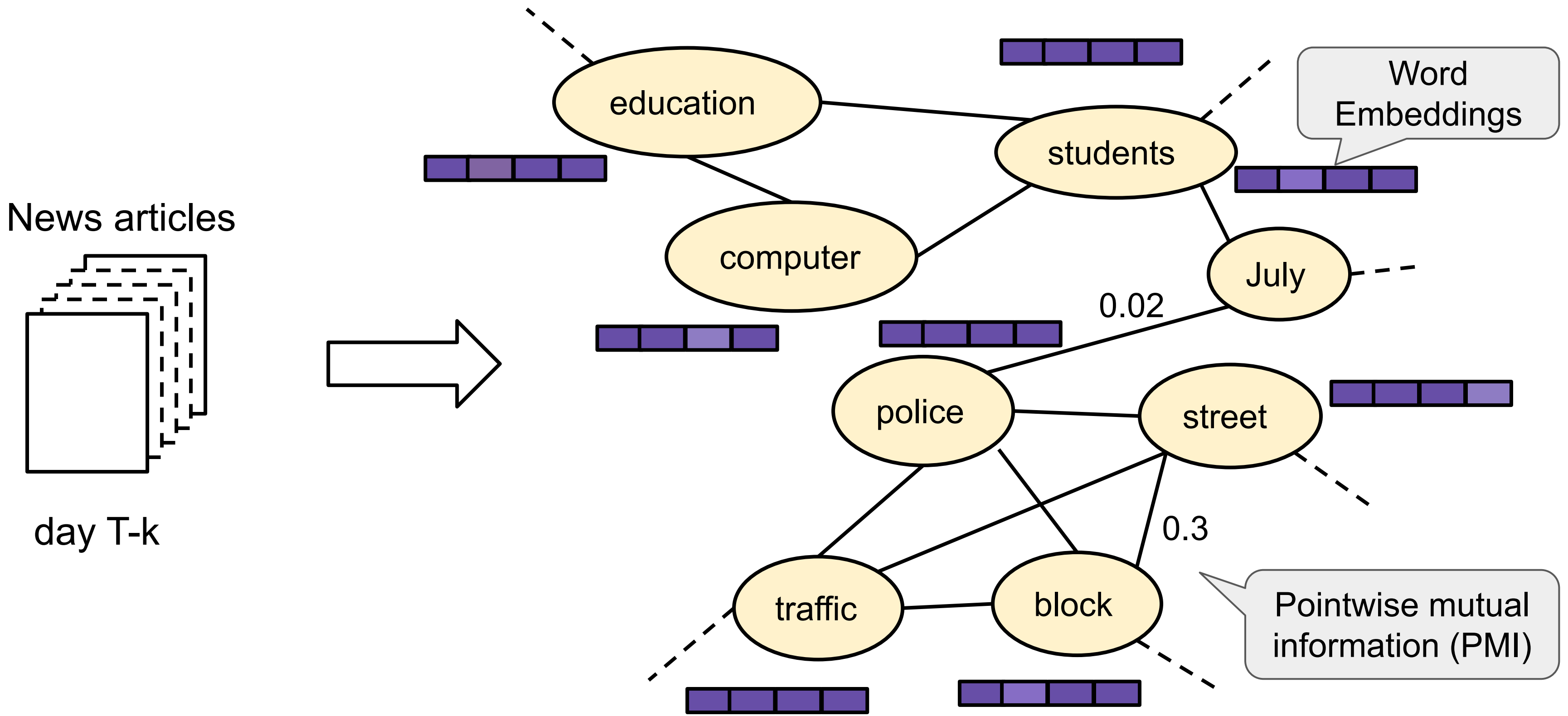


Motivation

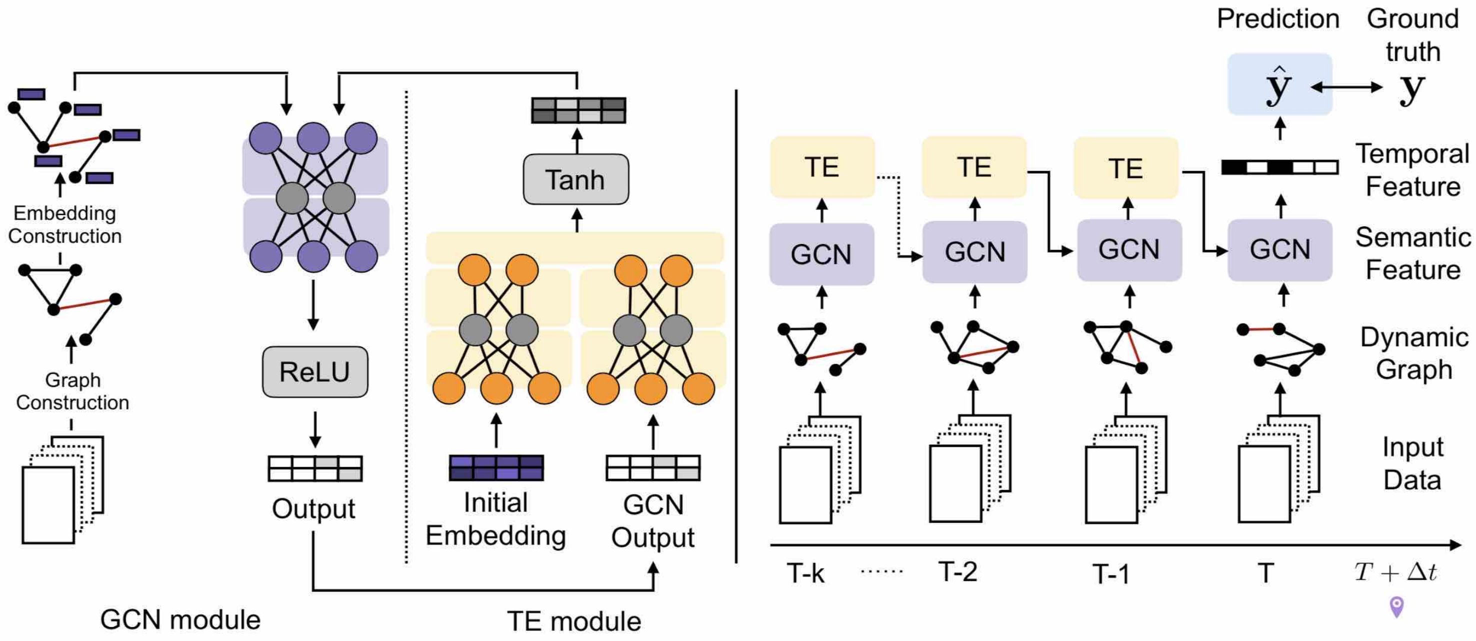
- Improve the interpretability of event forecasting algorithms by **providing supporting evidence/clue**, e.g., subgraphs.
- Graph structured data encode rich information and easy to understand.

Deng, Songgaojun, Huzefa Rangwala, and Yue Ning. **Learning dynamic context graphs for predicting social events**. *KDD.* 2019.

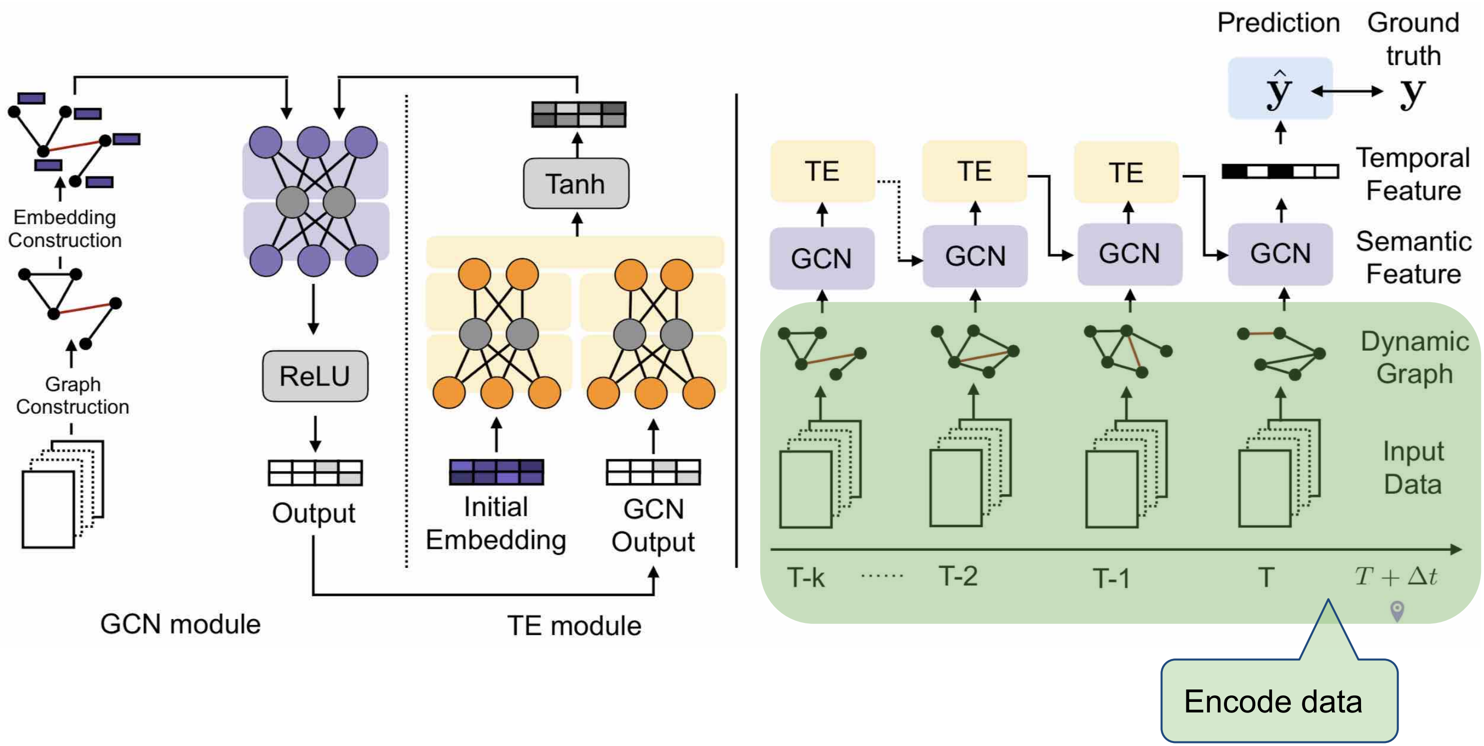
Encoding documents into graphs



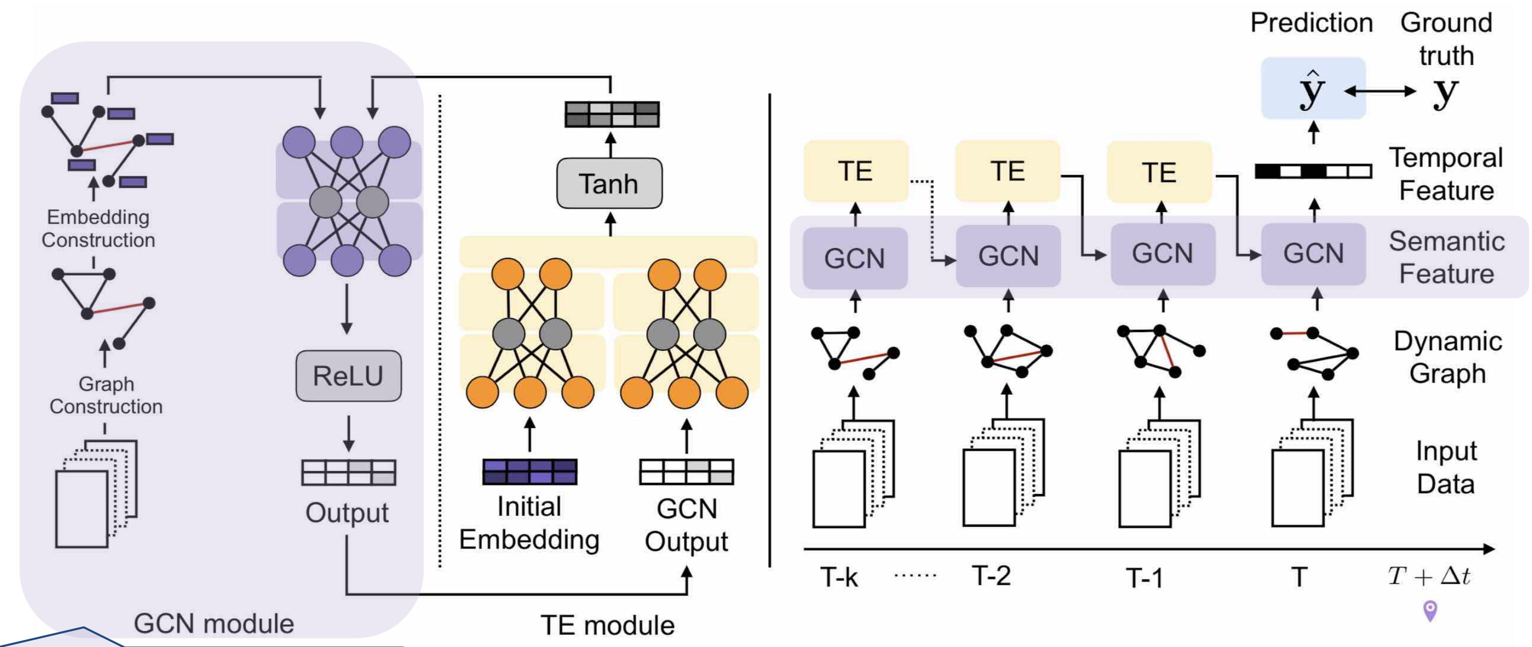
Model framework: DynamicGCN



Model framework: DynamicGCN

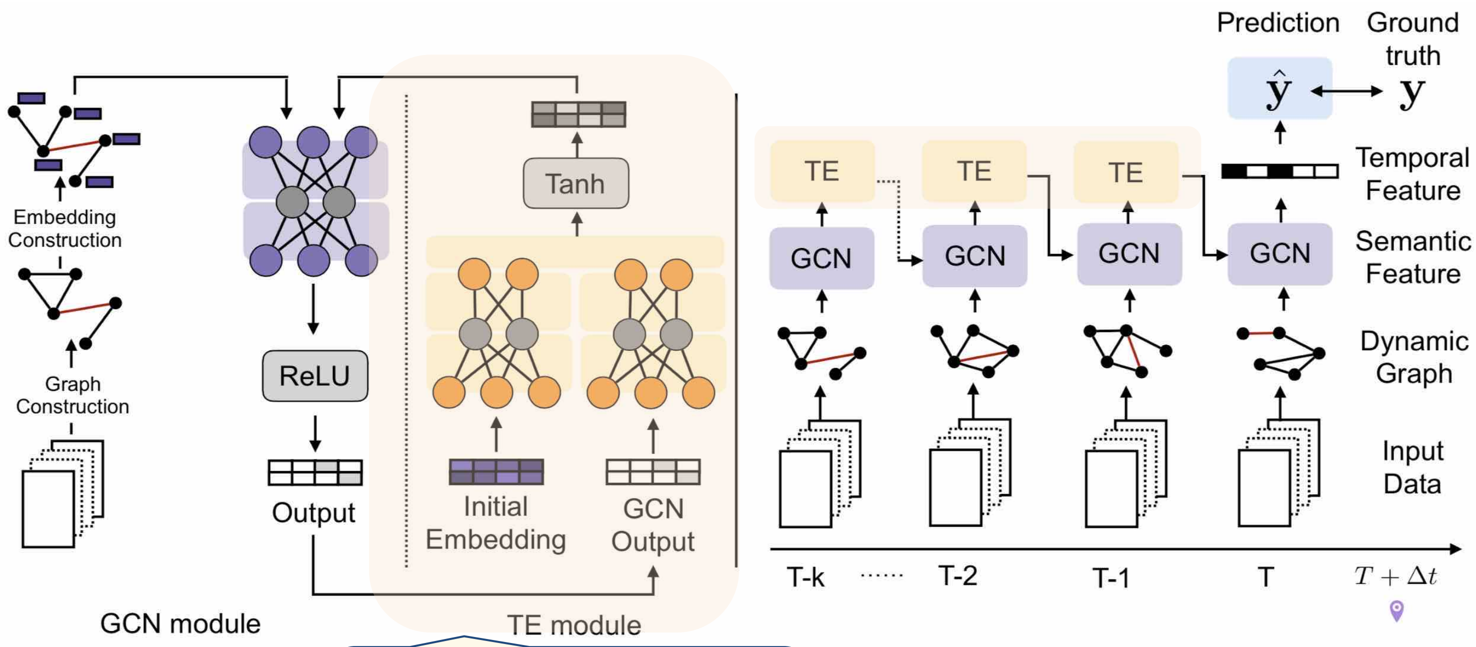


Model framework: DynamicGCN



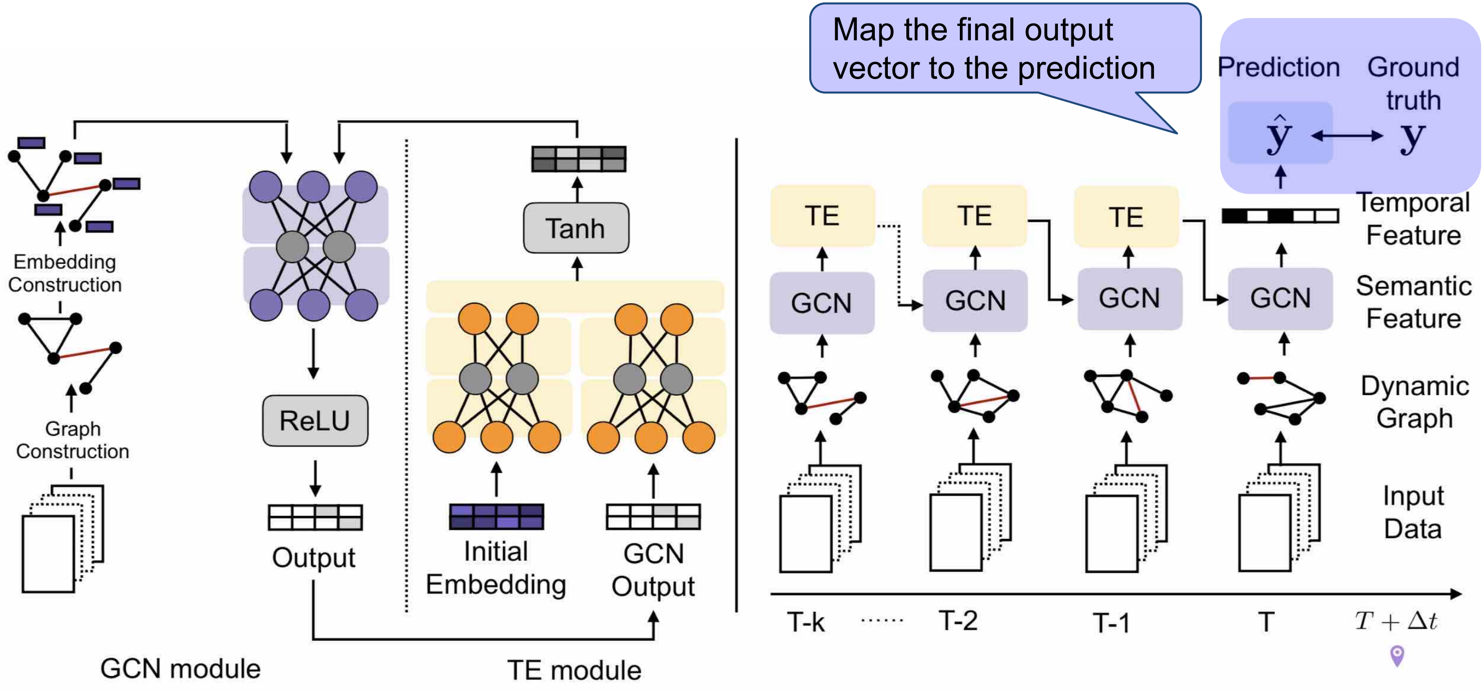
Learn the node representation by involving the semantic information from neighbors

Model framework: DynamicGCN



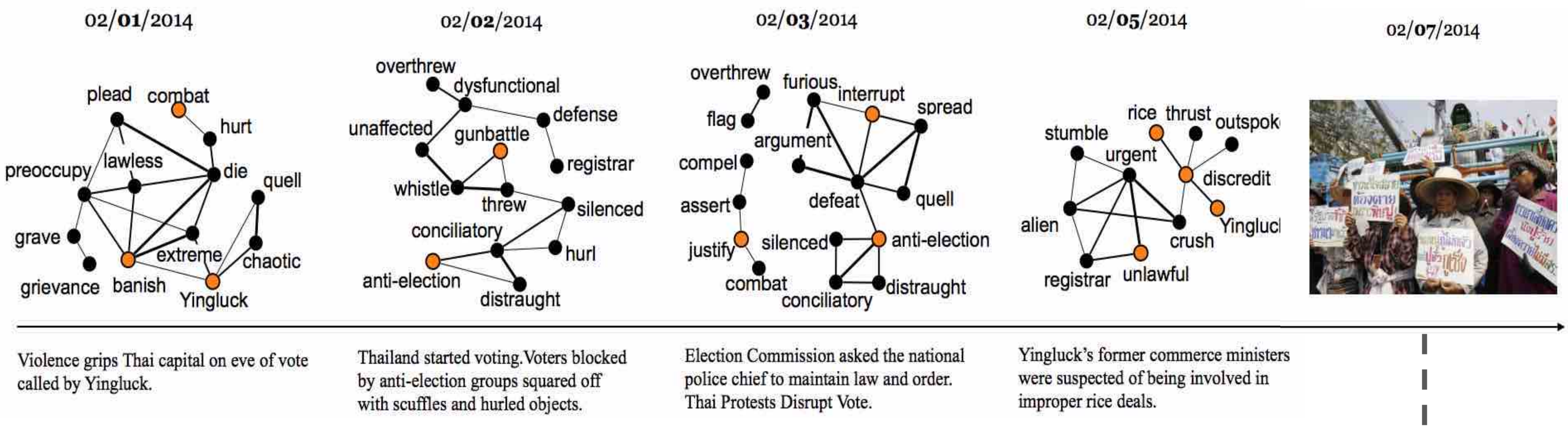
Integrate both **semantic** information and **contextual** relational information

Model framework: DynamicGCN



A case study

Context subgraphs generated from the trained model.



A possible fraud involving **rice** traders and some **politicians.**

The election related to **Yingluck** was ever **interrupted**

Coffee Break

(15 minutes)

Next: Part 1: Graph Neural Network (GNN)-based methods

- Graph learning with contextual information

Graph learning with contextual information

- Context in graph construction
- Context in graph learning

Graph learning with contextual information

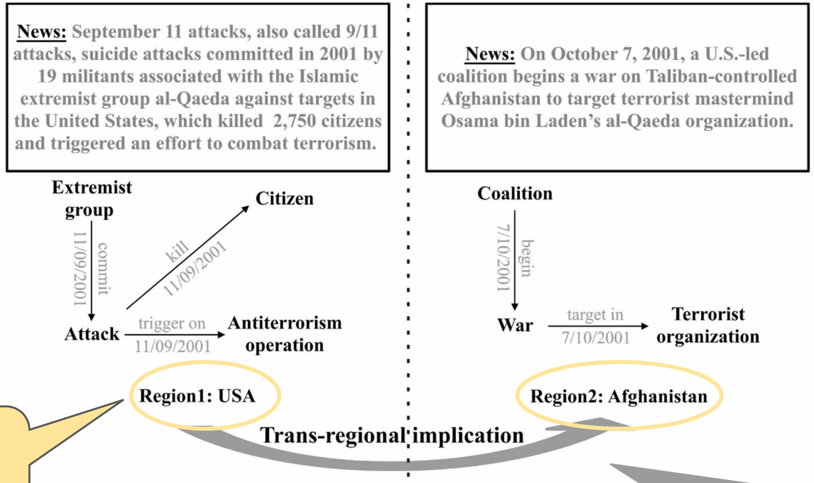
- **Context in graph construction**
- Context in graph learning

Spatial-temporal knowledge graph network for event prediction [Huai et al. 2023]

Introduced **additional spatial and textual information** in graph construction, and proposed a spatial and temporal knowledge graph neural network (STKGN)

Motivation

- Cause of the incident is **not local**.
- **Trans-regional implications** are behind the occurrences of events.



Region information as a set of nodes

Fig. 1. An example of trans-regional

Connect the two nodes

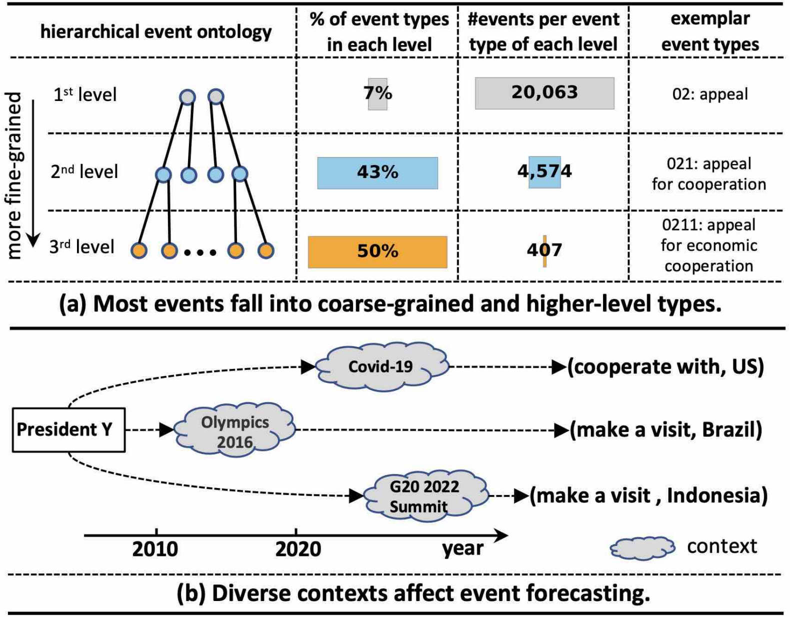
Huai, Zepeng, Guohua Yang, and Jianhua Tao. **Spatial-temporal knowledge graph network for event prediction**. Neurocomputing 553 (2023).

Context-aware event forecasting via graph disentanglement [Ma et al. KDD2023]

Introduce **specific categorical context** in graph construction, assuming the availability of external prior knowledge.

Motivation

- Most events fall into general types in the event ontology, and tend to be **coarse-grained**.
- Events defined by a fixed ontology **fail to retain out-of-ontology** contextual information.



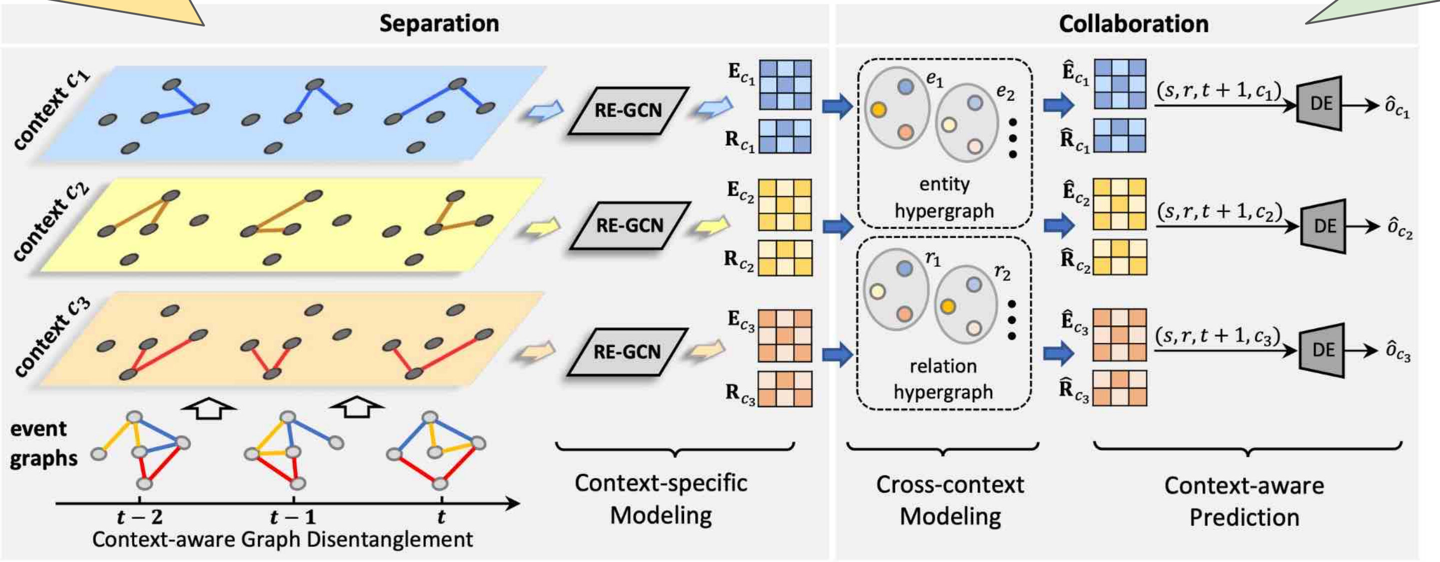
Ma, Yunshan, et al. **Context-aware event forecasting via graph disentanglement**. KDD. 2023.

Leverage contextual information

Define **context** as a **categorical value** and assign it to each event.

1. Use the ***context*** as a prior guidance to separate the event graphs.
2. Capture the context-specific patterns.

1. Construct **hypergraphs**.
2. GNN learns the collaborative associations among contexts.



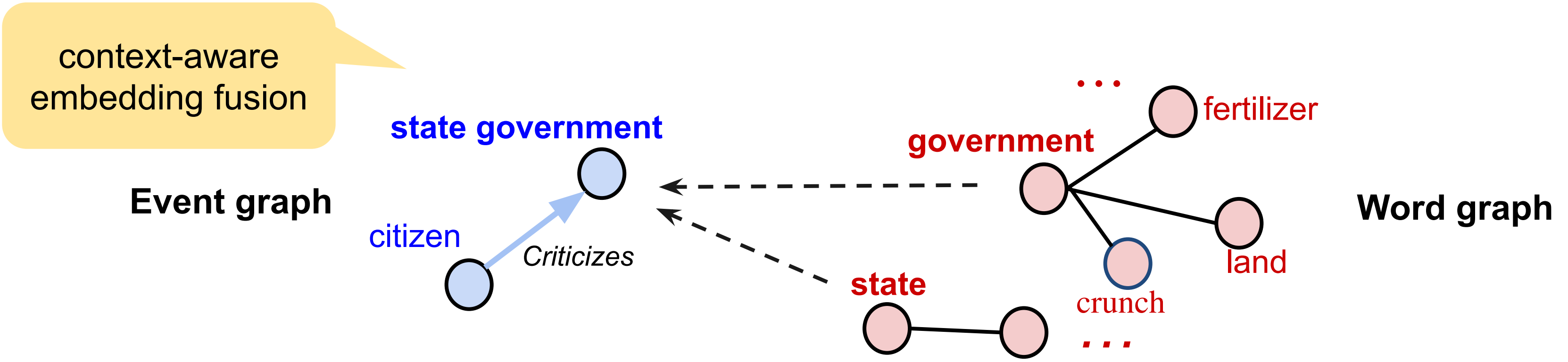
Graph learning with contextual information

- Context in graph construction
- **Context in graph learning**

Dynamic knowledge graph based multi-event forecasting

[Deng et al. KDD2020]

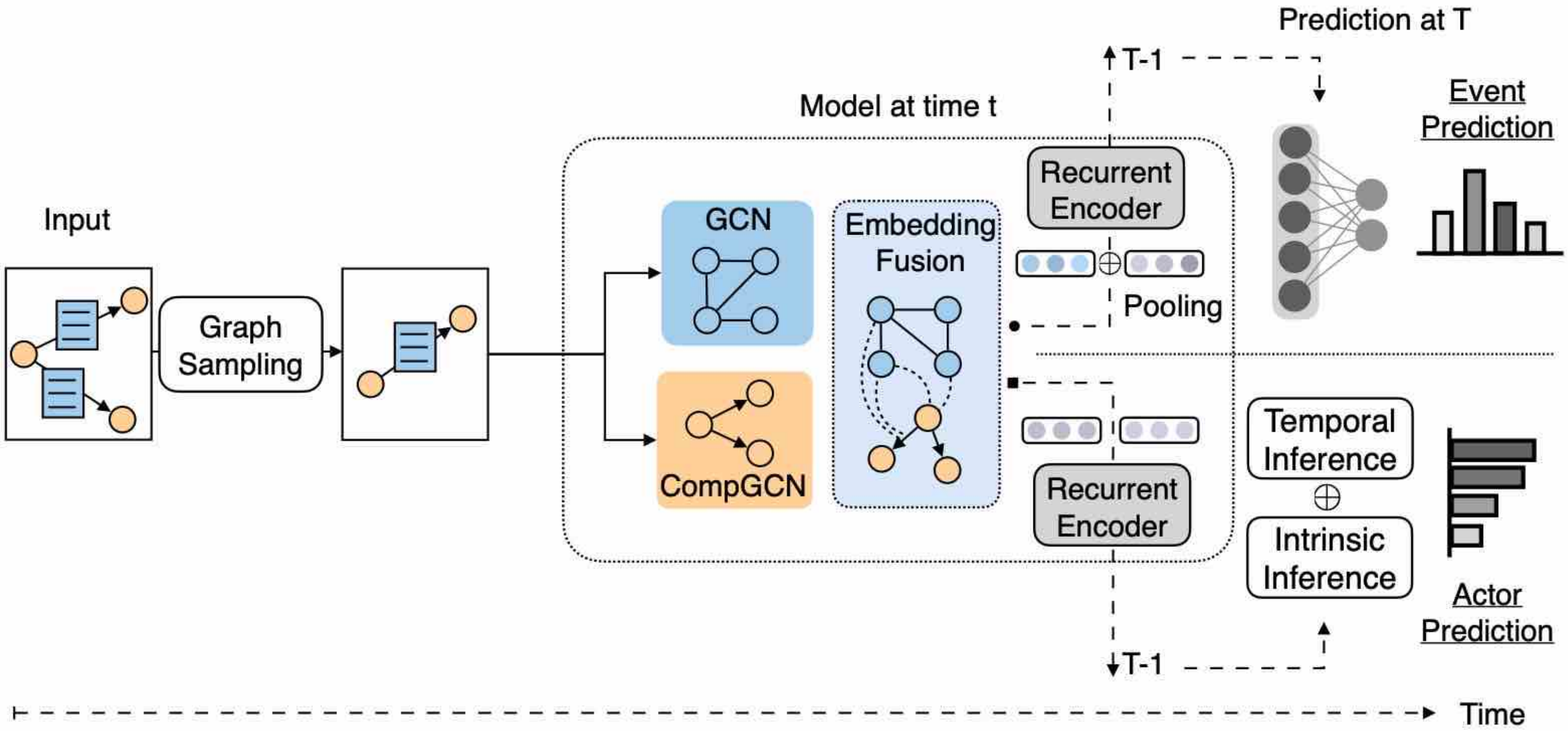
An event: (**Citizen**, *Criticizes*, **State government**, 02/26/2015) “A Politician attacked the government on various fronts such as *fertilizer crunch* and *land acquisition act*.”



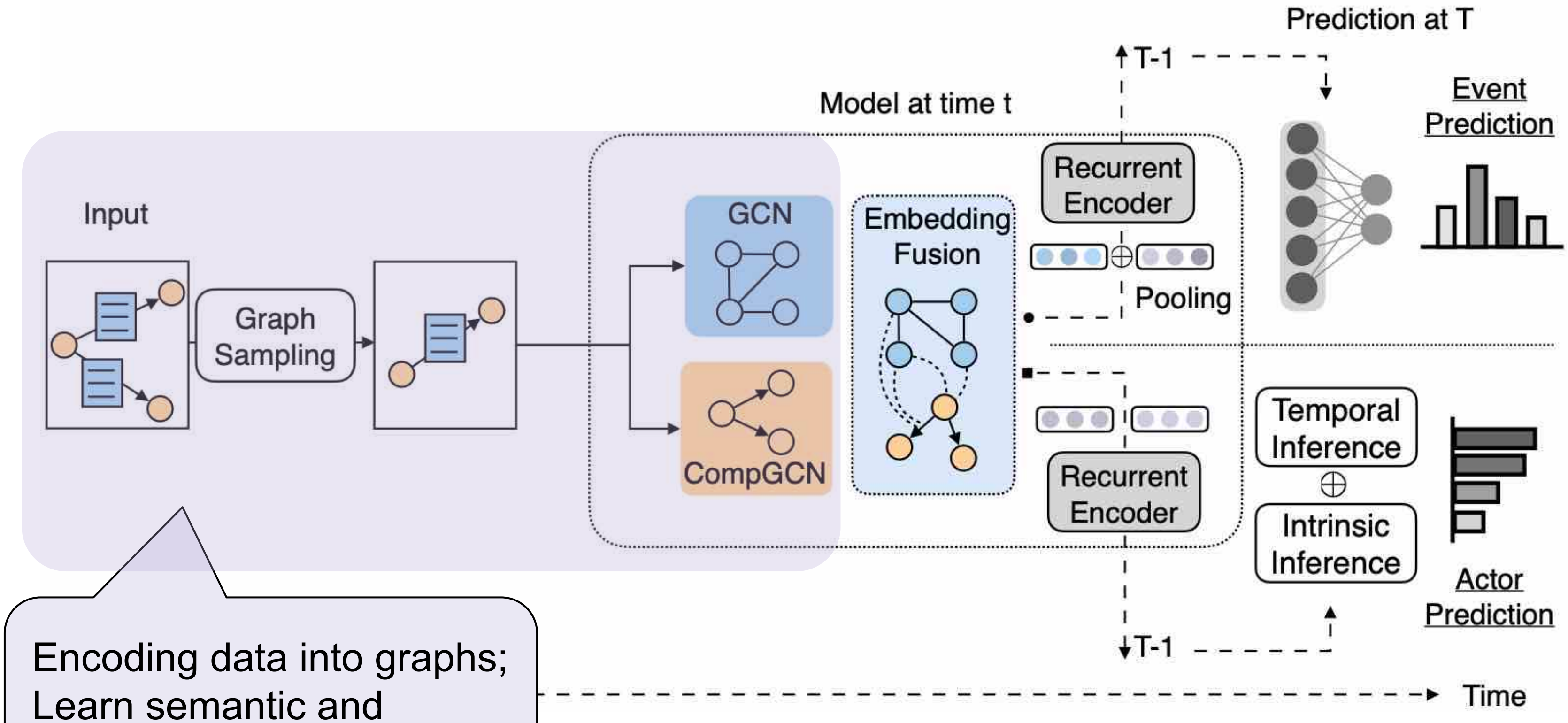
entity: **state government** ← relevant words : **[government, state]**

Deng, Songgaojun, Huzefa Rangwala, and Yue Ning. **Dynamic knowledge graph based multi-event forecasting**. *KDD*. 2020.

Model framework: Glean

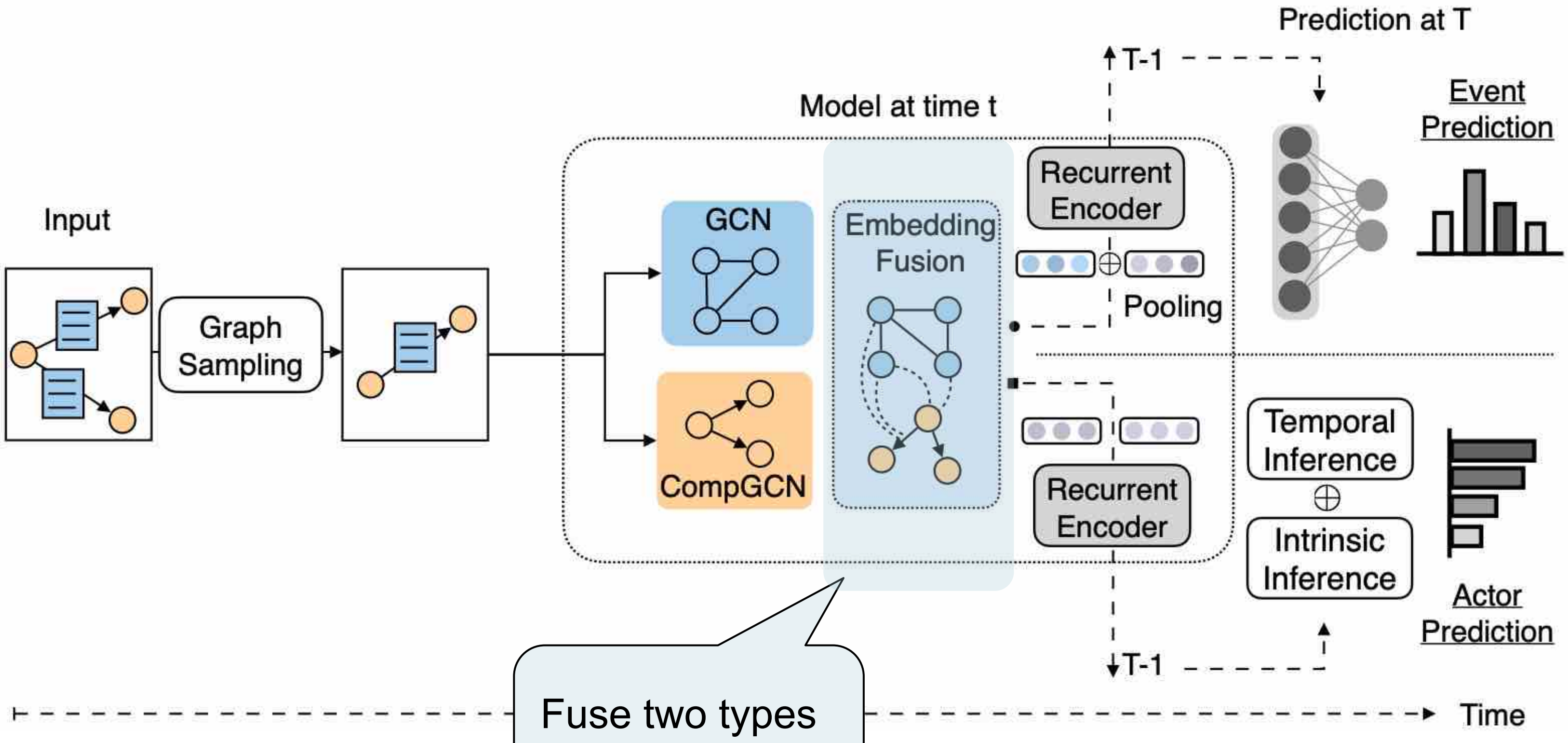


Model framework: Glean



Encoding data into graphs;
Learn semantic and
relational embedding

Model framework: Glean



Fuse two types of embeddings

Context-aware embedding fusion

An event: (**Citizen**, *Criticizes*, **State government**, 02/26/2015) "A Politician attacked the government on various fronts such as *fertilizer crunch* and *land acquisition act*."

Relational embedding learned from event graphs.

(State government)

(state/government)

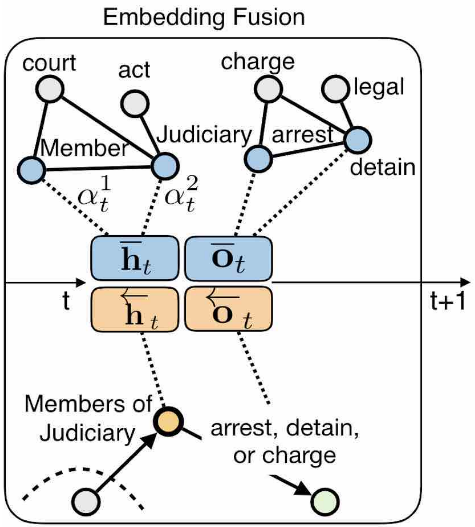
Semantic embedding learned from word graphs.

$$\alpha_{t,(i,\omega)} = \frac{\exp\left(\mathrm{Attn}\left(\overleftarrow{\mathbf{h}}_{t,(i)}, \bar{\mathbf{h}}_{\omega}\right)\right)}{\sum_{\varphi \in \mathcal{W}_i} \exp\left(\mathrm{Attn}\left(\overleftarrow{\mathbf{h}}_{t,(i)}, \bar{\mathbf{h}}_{\varphi}\right)\right)} \in \mathbb{R}, \qquad (7)$$

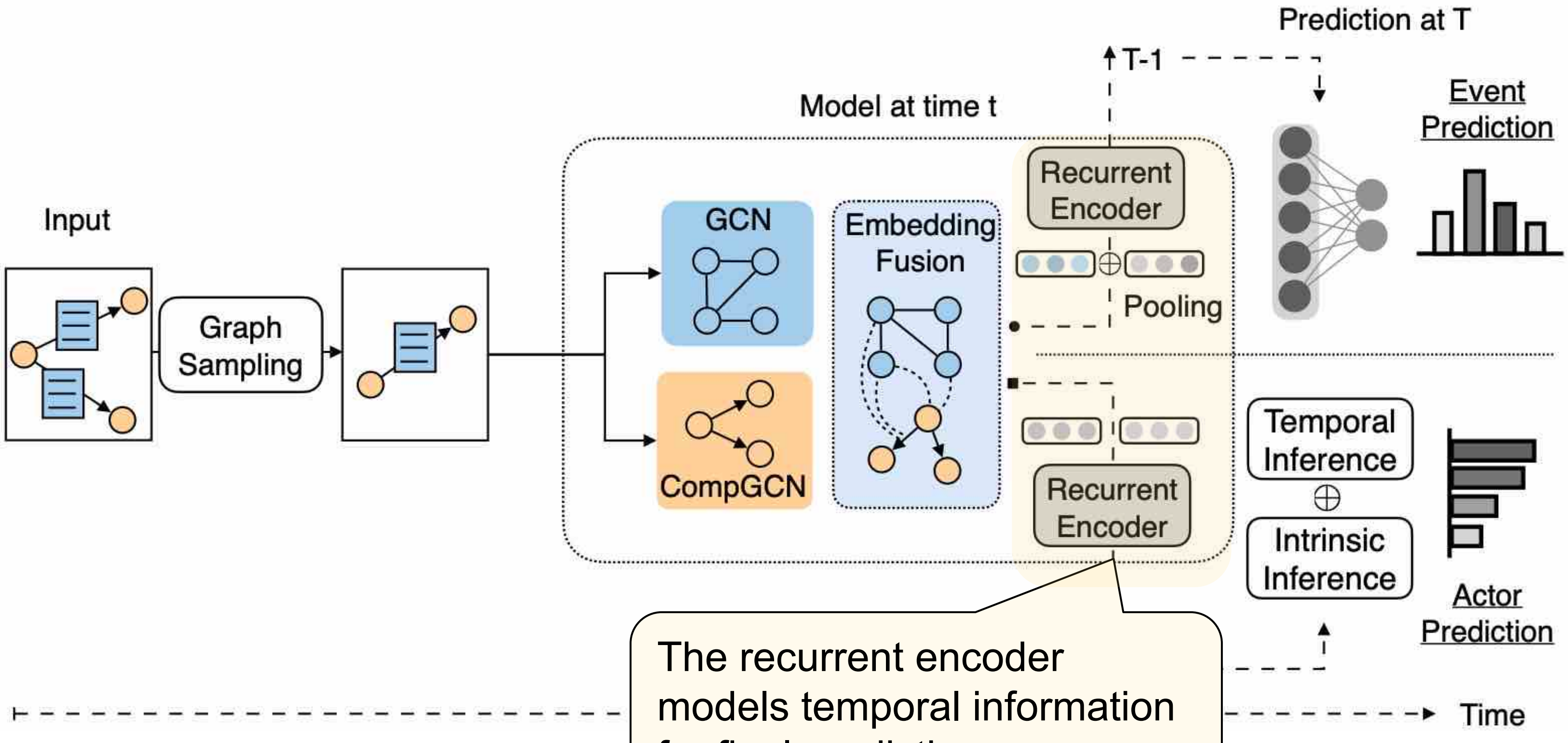
$$\mathbf{h}^{\spadesuit}_{t,(i)} = \tanh\left(\mathbf{W}_{\alpha}^{\top} \cdot \left[\underbrace{\overleftarrow{\mathbf{h}}_{t,(i)}}_{\text{rel.}}; \underbrace{\sum_{\omega \in \mathcal{W}_i} \alpha_{t,(i,\omega)} \bar{\mathbf{h}}_{\omega}}_{\text{semantic}}\right]\right) \in \mathbb{R}^d, \qquad (8)$$

Fused embedding that enhances the information of entities (or event types) from words.

$|\mathcal{W}_i| \geq 1$: the set of words related to entity i



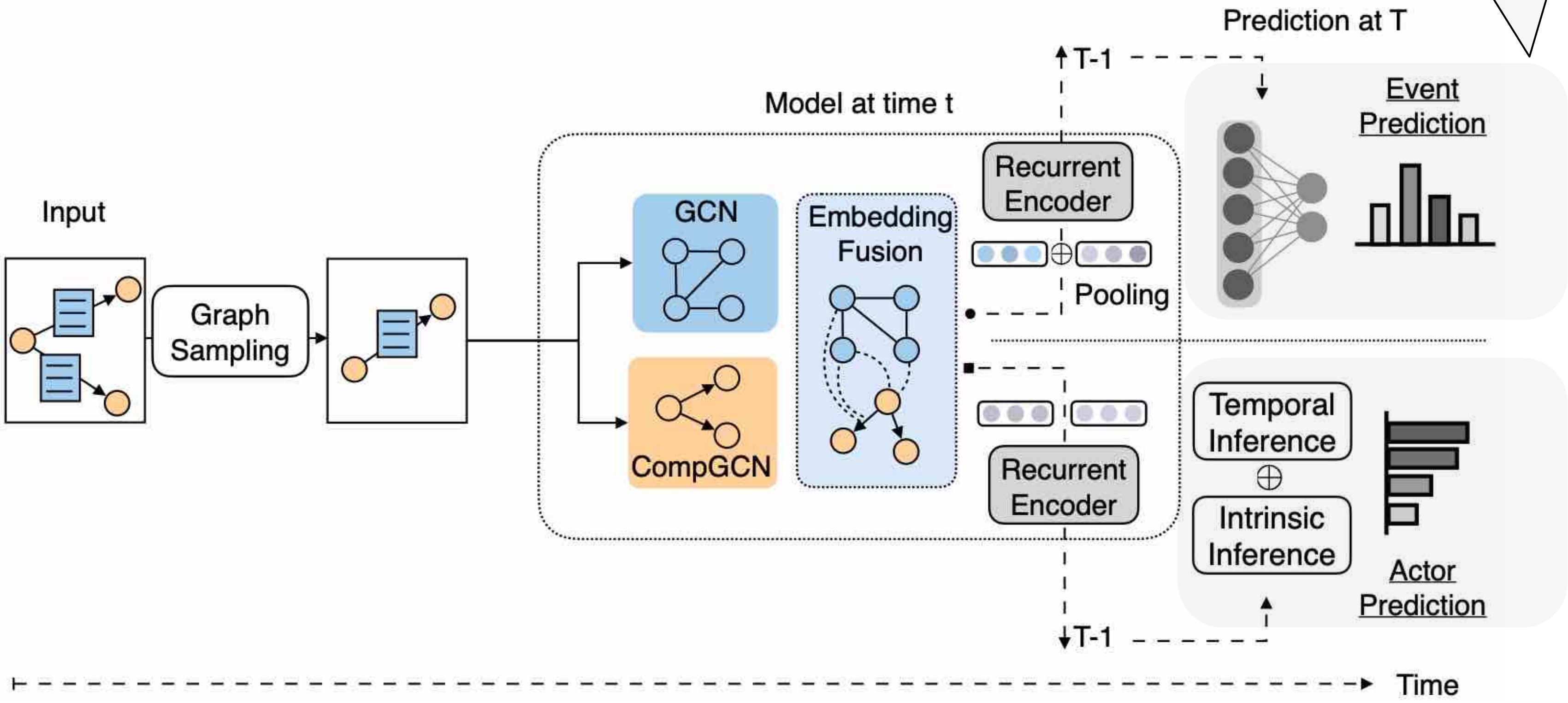
Model framework: Glean



The recurrent encoder models temporal information for final predictions.

Model framework: Glean

Prediction model



Experimental results

Glean is the best and "fusion" is helpful.

Multi-event prediction

Method	India			Russia		
	F1	F2	Recall	F1	F2	Recall
DNN	52.49	54.65	56.38	53.81	58.44	62.61
MLkNN	52.33	54.27	55.77	51.38	55.29	58.62
BRkNN	50.36	53.05	56.00	47.46	51.53	56.64
MLARAM	33.68	33.93	34.10	25.67	26.27	26.71
DynGCN	41.80	42.57	43.19	52.81	56.77	60.14
T-GCN	60.73	64.14	67.20	56.36	61.86	67.66
RENET[1]	55.10	57.26	58.99	54.47	58.98	63.02
RENET[2]	58.44	61.46	64.18	55.85	60.86	65.66
$\text{Glean}_{-\text{fusion}}$	65.91	70.87	75.80	**58.92**	65.60	73.47
Glean	**66.69**	**71.95**	**77.31**	**58.92**	**65.64**	**73.57**
% relative gain	9.8%	10.9%	15.0%	4.5%	6.1%	8.7%

Multi-actor prediction

Method	India			Russia		
	H @ 1	3	10	1	3	10
DNN	2.09	11.01	33.87	1.46	9.72	36.40
RENET[3]	8.87	21.57	39.85	16.52	22.31	40.21
tRGCN	9.74	22.74	41.04	18.83	30.79	44.62
tCompGCN	9.62	21.91	40.53	18.27	30.20	44.79
$\text{Glean}_{-\text{temp}}$	13.39	24.50	43.68	18.24	31.15	43.27
$\text{Glean}_{-\text{fusion}}$	13.95	27.03	**45.73**	20.25	**34.64**	**48.10**
Glean	**14.01**	**27.17**	**45.73**	**20.49**	34.36	**48.10**
% relative gain	4.6%	10.9%	4.7%	8.8%	11.2%	7.4%

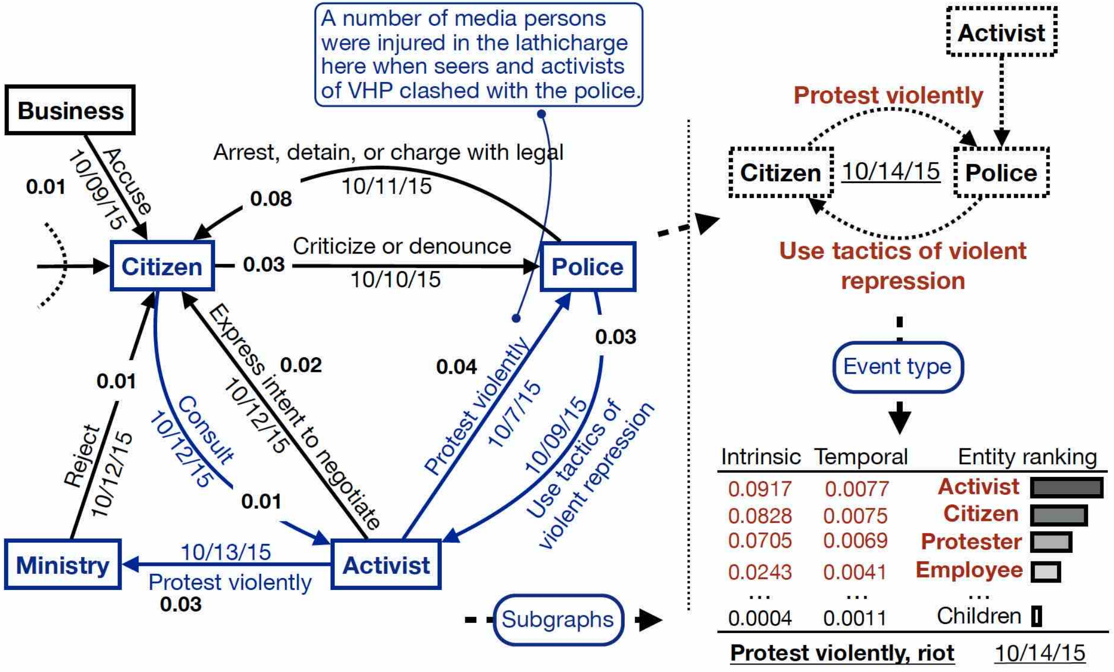
Case study

Identifying important historical events.

The blue part represents the subgraph sampled for actor prediction.

- Select entities that partially retain their feature variables after a max pooling layer.

The red font indicates the model prediction.

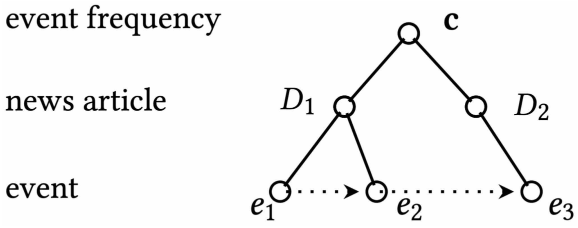


Understanding event predictions via contextualized multilevel feature learning [Deng et al. CIKM2021]

Introduce **contextualized multilevel feature learning** for interpretable event forecasting, i.e., providing example-level explanations.

Motivation

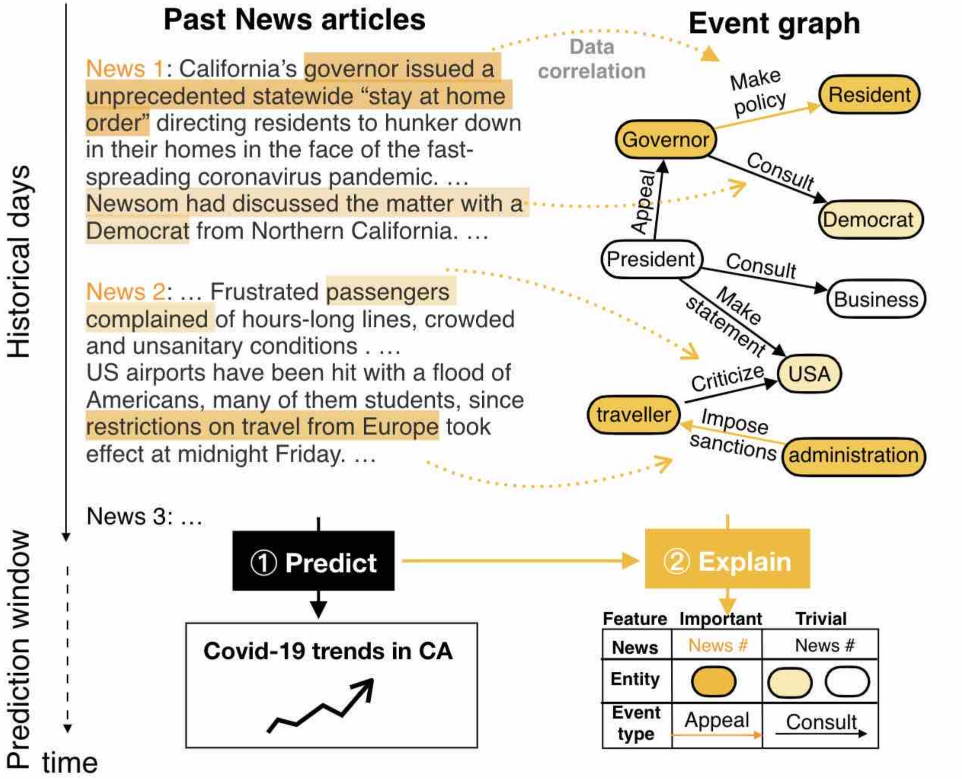
- Event data (i.e., ICEWS) can be organized in **multiple levels**.
- Combining multilevel features can provide **rich explanations**.



Deng, Songgaojun, et al. **Understanding event predictions via contextualized multilevel feature learning**. *CIKM*. 2021.

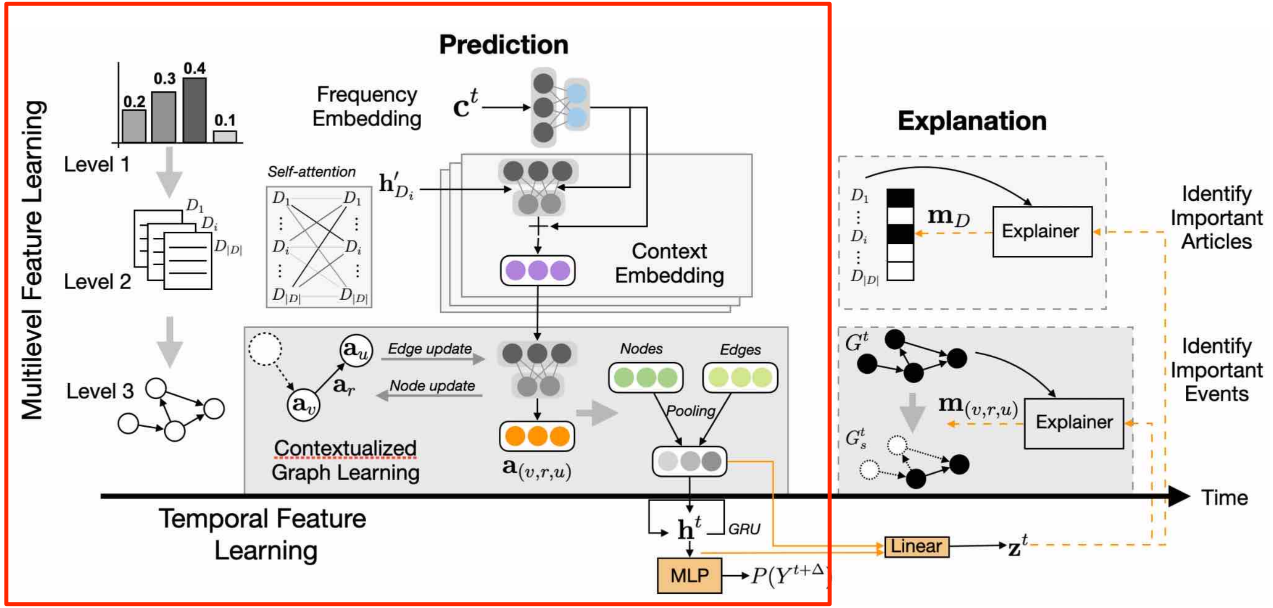
A motivating example

Predicting and explaining Covid-19 trends in CA.



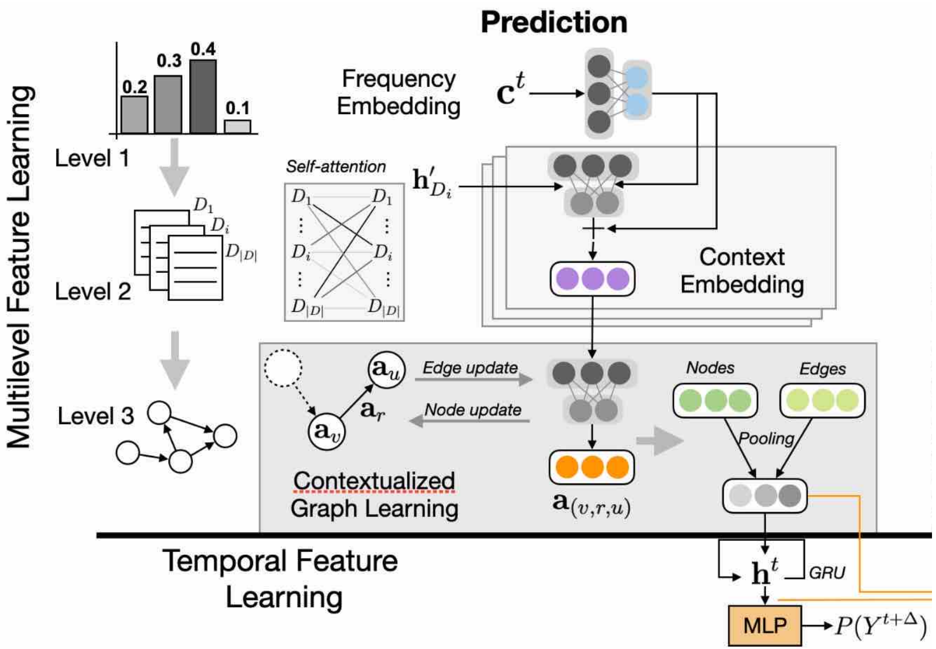
Model framework: CMF (Contextualized Multilevel Feature learning)

- A **predictor** fuses hierarchical and heterogeneous data, and learns sequential features.



Prediction

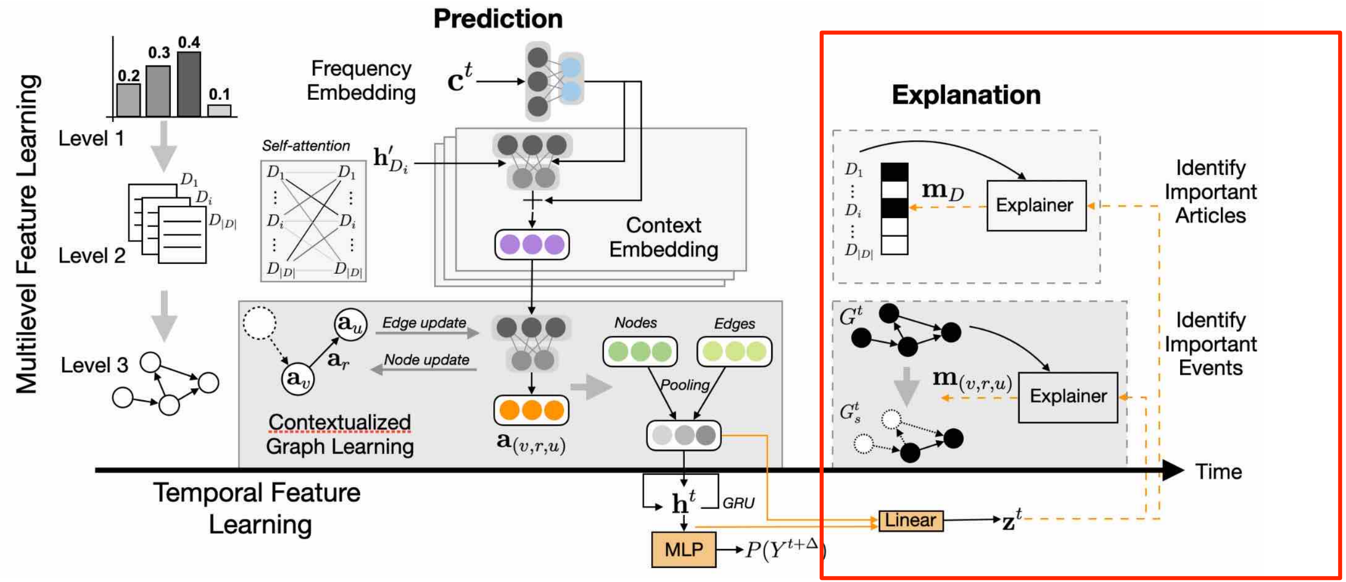
1. **Multilevel feature learning** that hierarchically models heterogeneous data.
   - *Level 1 Frequency Embedding*
   - *Level 2 Context Embedding*
   - *Level 3 Contextualized Graph Learning*



2. GRU **learns sequential information** across different historical time steps.

Model framework: CMF (Contextualized Multilevel Feature learning)

- An **explainer** provides temporal and multilevel explanations.



Multilevel explanation

The learning objective:

- Identify a subset of **key news articles** and a **subgraph of key events** over the historical time window.

Formalize the **feature importance** using Mutual Information (MI):

$$\underset{\{D_s\}^{t-w+1:t}}{\text{Max}} \; MI\left(\hat{y}^{t+\Delta}, \{\mathcal{D}_s\}^{t-w+1:t}\right)$$
$$= H\left(\hat{y}^{t+\Delta}\right) - H\left(\hat{y}^{t+\Delta} \middle| \{\mathcal{D}_s\}^{t-w+1:t}\right)$$

Since direct approximation is intractable, we relax the objective using parameterized neural networks to obtain $\{\mathcal{D}_s\}^{t-w+1:t}$, i.e., learn important scores for articles and events.

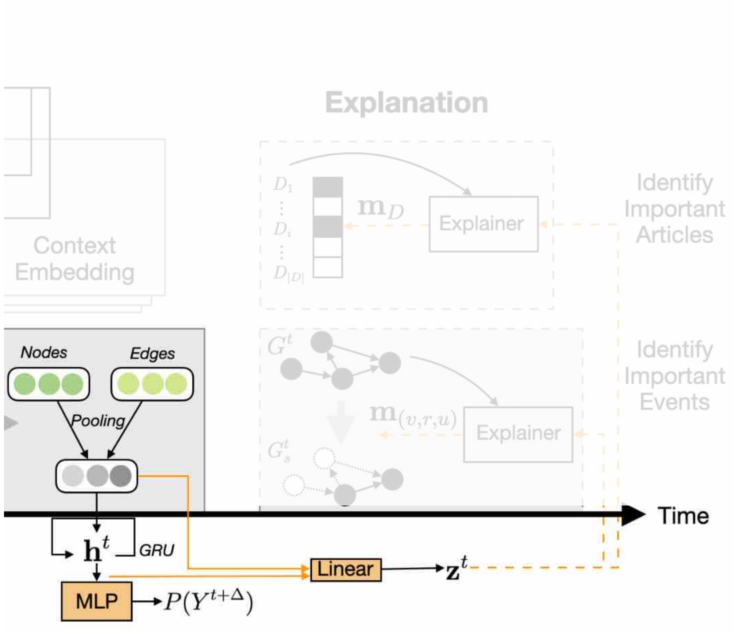
Multilevel explanation

1. Reference embedding, a guideline for selecting important features at each historical time step.

Averaged node and edge embeddings

$$\mathbf{z}^{\tau} = f_{z_1}\left(\boxed{\mathbf{h}^{t}} \oplus \boxed{\bar{\mathbf{u}}^{\tau} \oplus \bar{\mathbf{r}}^{\tau}}\right), \quad t - w + 1 \leq \tau \leq t$$

Final feature vector (GRU)



Multilevel explanation

1. Reference embedding, a guideline for selecting important features at each historical time step.

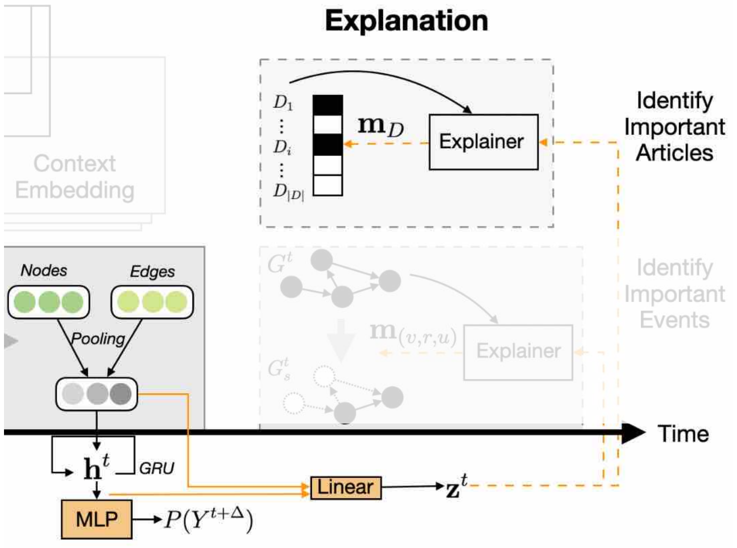
Averaged node and edge embeddings

$$\mathbf{z}^{\tau} = f_{z_1}\left(\boxed{\mathbf{h}^{t}} \oplus \boxed{\bar{\mathbf{u}}^{\tau} \oplus \bar{\mathbf{r}}^{\tau}}\right), \quad t - w + 1 \leq \tau \leq t$$

Final feature vector (GRU)

2. Semantic explanations which identify **key documents** at each historical time step.

$$\alpha_{D_i} = \mathbf{v}^{\top}\mathrm{Tanh}\big(\mathbf{W}_a(\mathbf{h}_{D_i} \oplus \mathbf{z}^{t})\big)$$



Multilevel explanation

1. Reference embedding, a guideline for selecting important features at each historical time step.

Averaged node and edge embeddings

$$\mathbf{z}^{\tau} = f_{z_1}\left(\boxed{\mathbf{h}^t} \oplus \boxed{\bar{\mathbf{u}}^{\tau} \oplus \bar{\mathbf{r}}^{\tau}}\right), \quad t - w + 1 \leq \tau \leq t$$

Final feature vector (GRU)

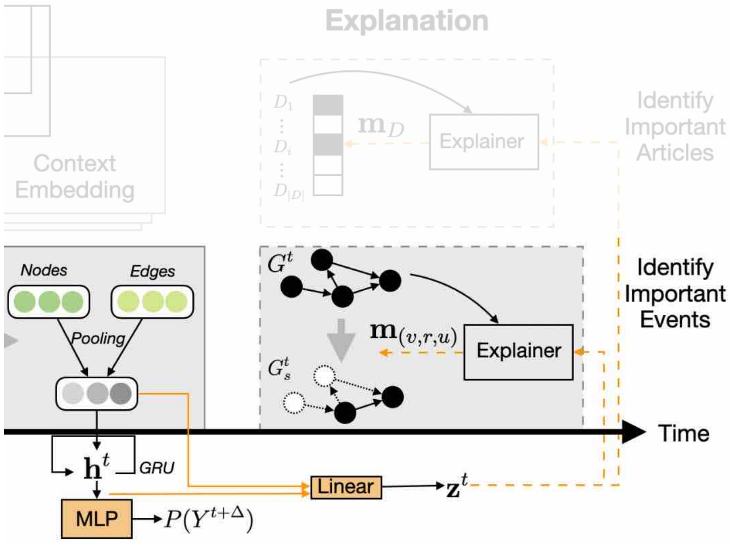
2. Semantic explanations which identify **key documents** at each historical time step.

$$\alpha_{D_i} = \mathbf{v}^{\top}\mathrm{Tanh}\big(\mathbf{W}_a(\mathbf{h}_{D_i} \oplus \mathbf{z}^t)\big)$$

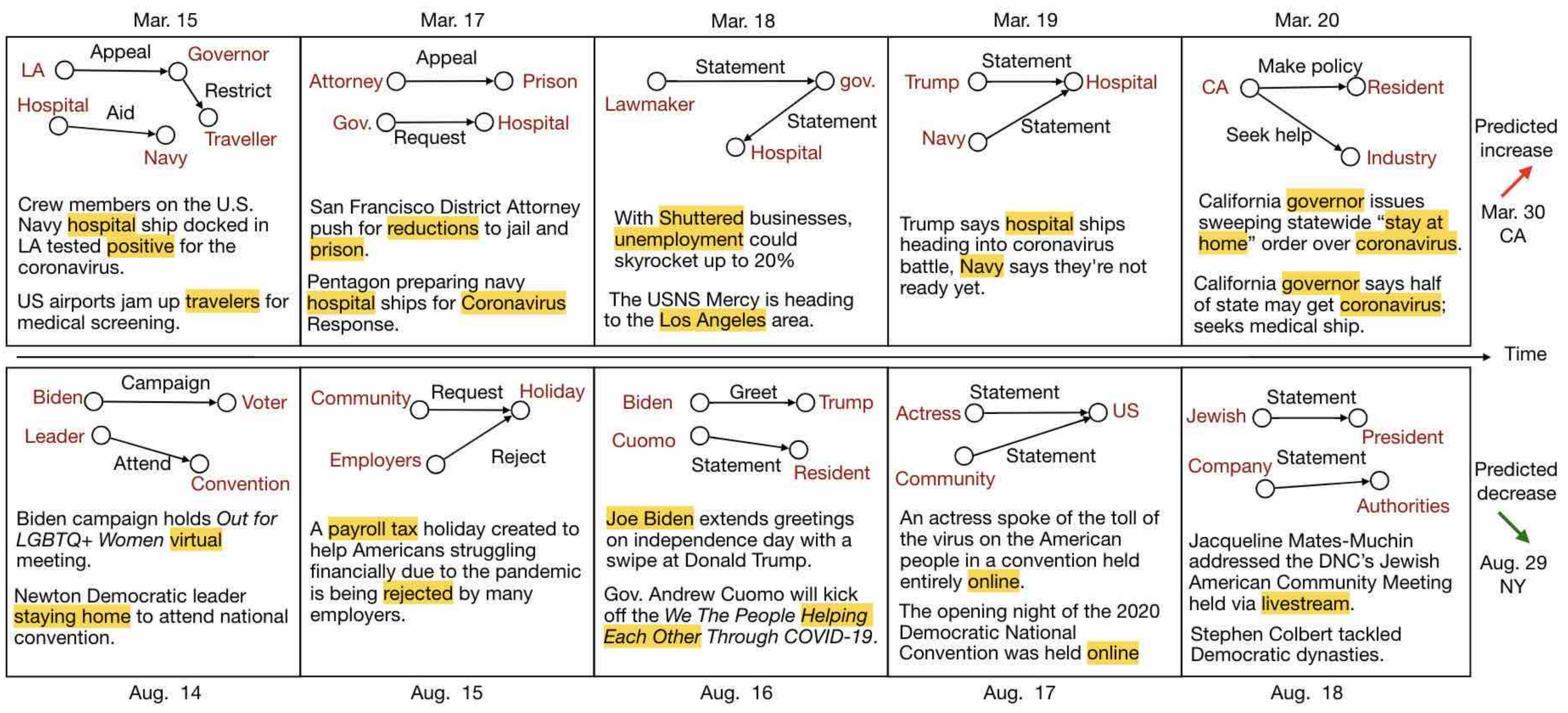
3. Relational explanations which detect **key events** by applying edge masks.

$$\alpha_{(v,r,u)} = \mathrm{Tanh}\Big(f_{z_2}\big(\mathbf{z}^t \oplus f_{ex}\boxed{(\mathbf{a}_v \oplus \mathbf{a}_r \oplus \mathbf{a}_u)}\big)\Big)$$

Event tuple info.



Two explanation examples on the Covid-19 dataset



Text-enhanced multi-granularity temporal graph learning for event prediction [Han et al. ICDM2022]

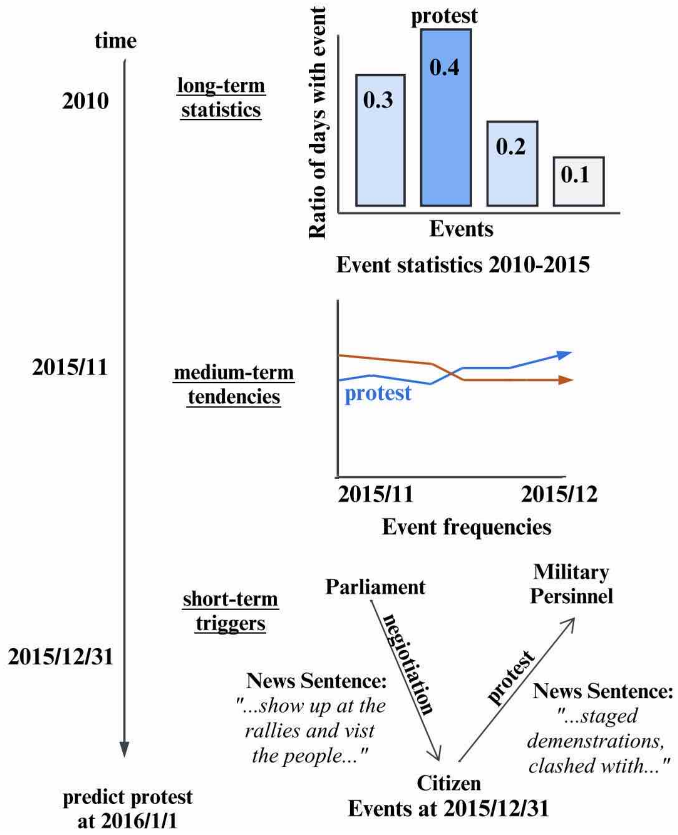
Develop a graph-based model that learns entities and event types embeddings through **different historical hierarchies.**

Motivation

- Existing methods rely on **Markov assumption** that event's probability only depends on **very recent history** – a condition not always true in real life.

Han, Xiaoxue, and Yue Ning. **Text-enhanced multi-granularity temporal graph learning for event prediction**. *ICDM*. IEEE, 2022.

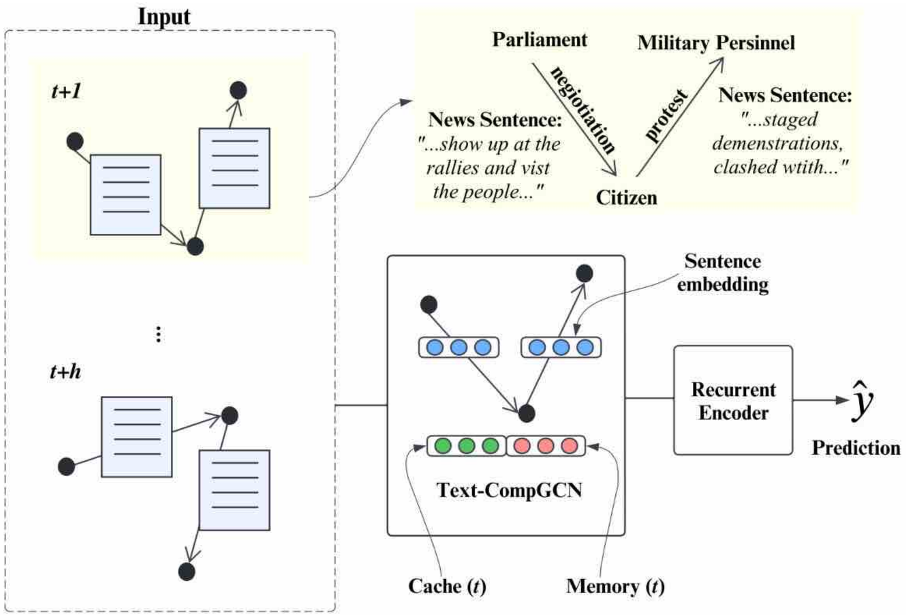
An intuitive example



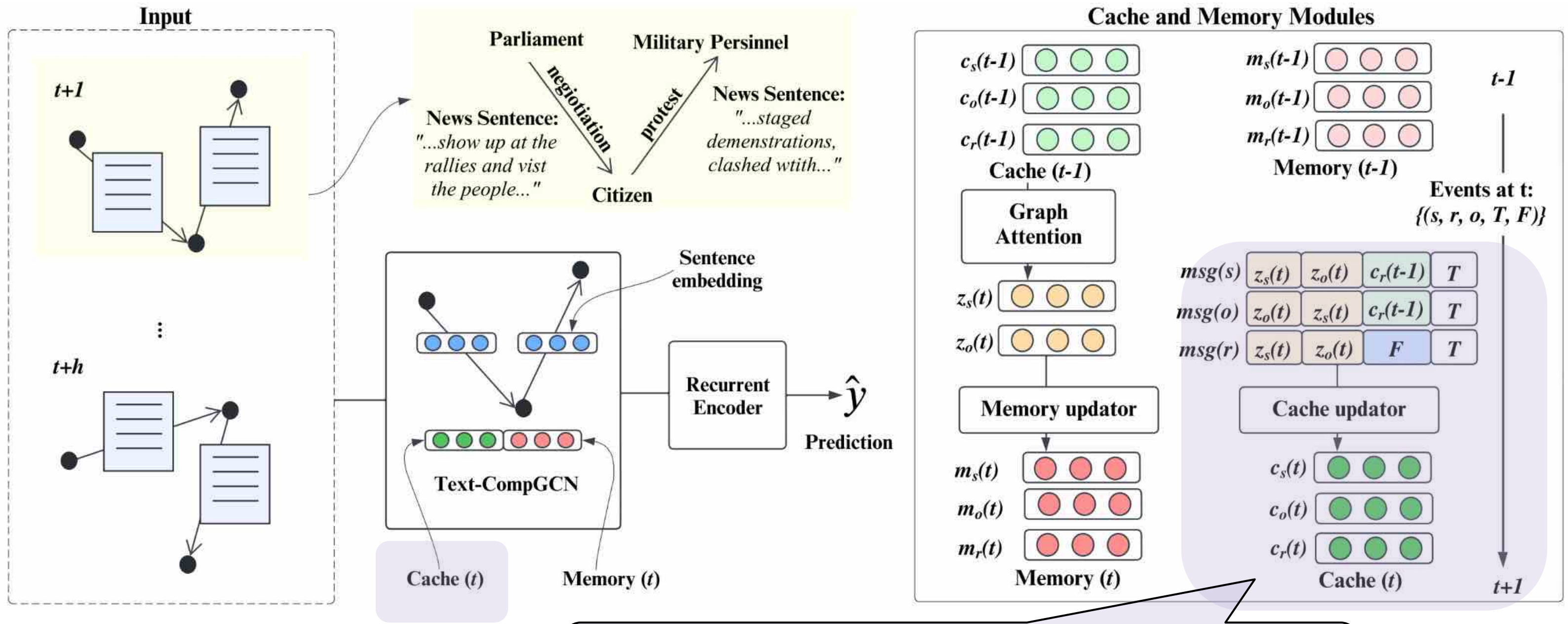
To predict if protests would happen tomorrow in a city, three potential questions:

- **Long-term statistics**: Is this society in a stable status? *Annual statistics can act as prior knowledge for the prediction.*
- **Mid-term tendencies**: Do people tend to protest these days? *Factors can be current economy and unemployment rates, and a look-back period like few months.*
- **Short-term triggers**: Any recent events that may trigger a protest?

Model framework: MTG (Multiple Temporal Granularities)

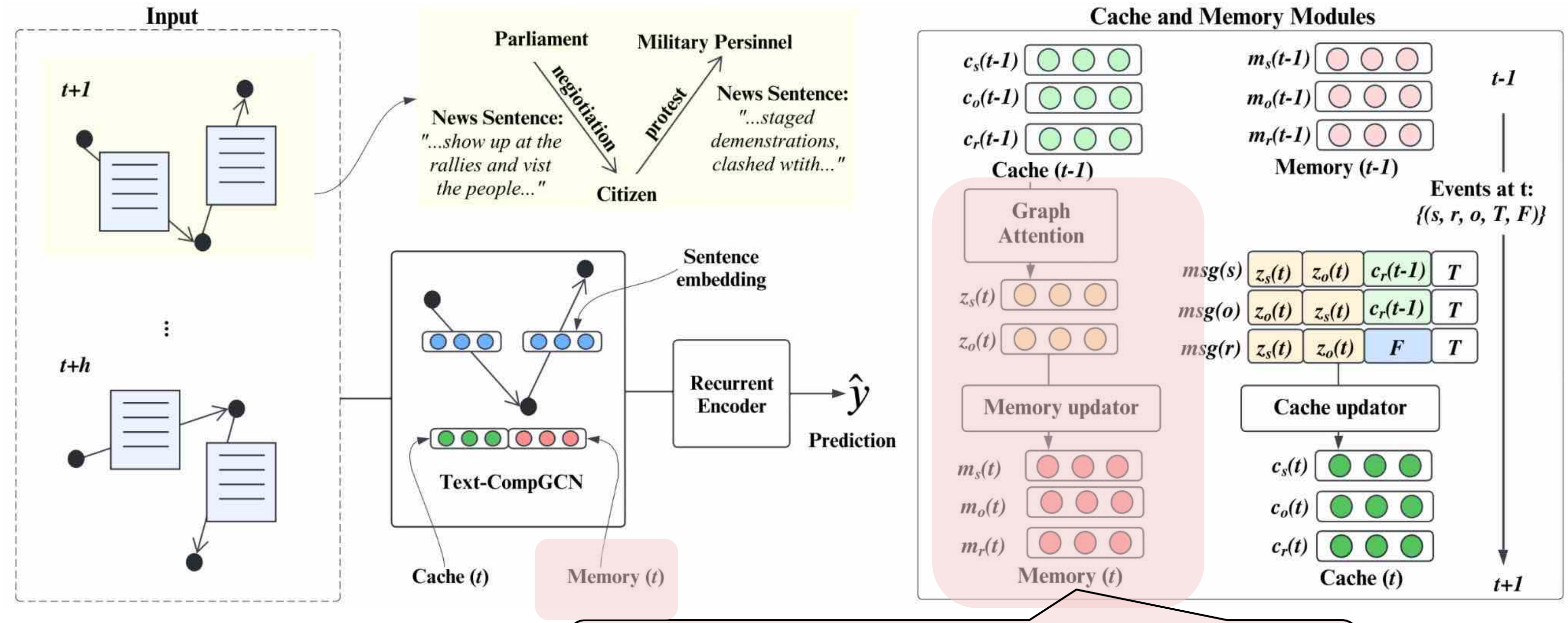


Model framework: MTG (Multiple Temporal Granularities)



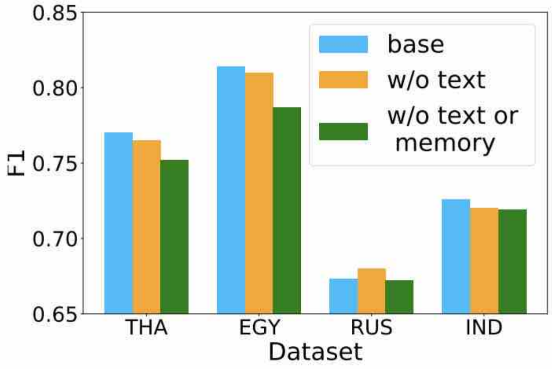
A RNN-based **cache** module models the mid-term tendencies, similar to RNN.

Model framework: MTG (Multiple Temporal Granularities)

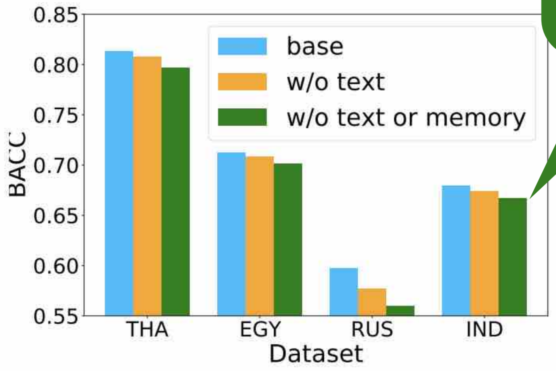


Memory module models the long-term statistics, using a memory matrix.

Impact of the text feature and the memory module



(a) F1



(b) BACC

Long-term information is helpful!

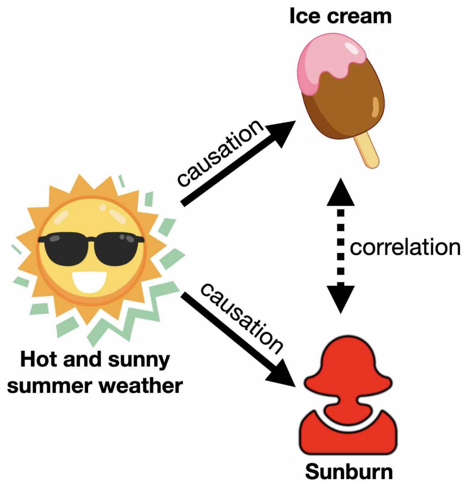
Graph learning with causal reasoning

Two-stage approaches:

- Discover causal information
- Use them to assist in forecasting

Causality

Causality is an influence by which one event contributes to the occurrence of another event.



Correlations v.s. causation

Causal reasoning is a promising direction for improving prediction accuracy and interpretability in event forecasting.

Robust Event Forecasting with Spatiotemporal Confounder Learning [Deng et al. KDD22]

Leverage **estimated event-related causal effects as prior knowledge** for event prediction

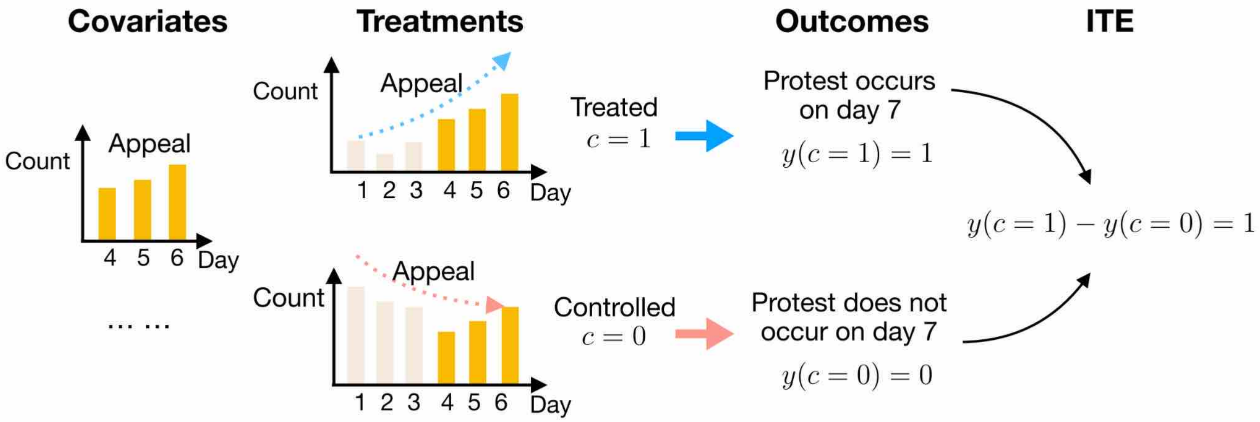
Motivation

- Existing event studies focus on correlation analysis.
- **Causal effect learning** has shown advantages in improving predictions in recommender systems, disease diagnosis, and computer vision.
- Studying causal effects of events might contribute to more robust predictions of events (e.g., less susceptible to noise in data).

Deng, Songgaojun, Huzefa Rangwala, and Yue Ning. **Robust event forecasting with spatiotemporal confounder learning**. *KDD*. 2022.

Problem formulation

- Estimate **individual treatment effects (ITE)** of multiple pre-defined treatment events (e.g., appeal, demand, etc.) on a target event (i.e., protest).

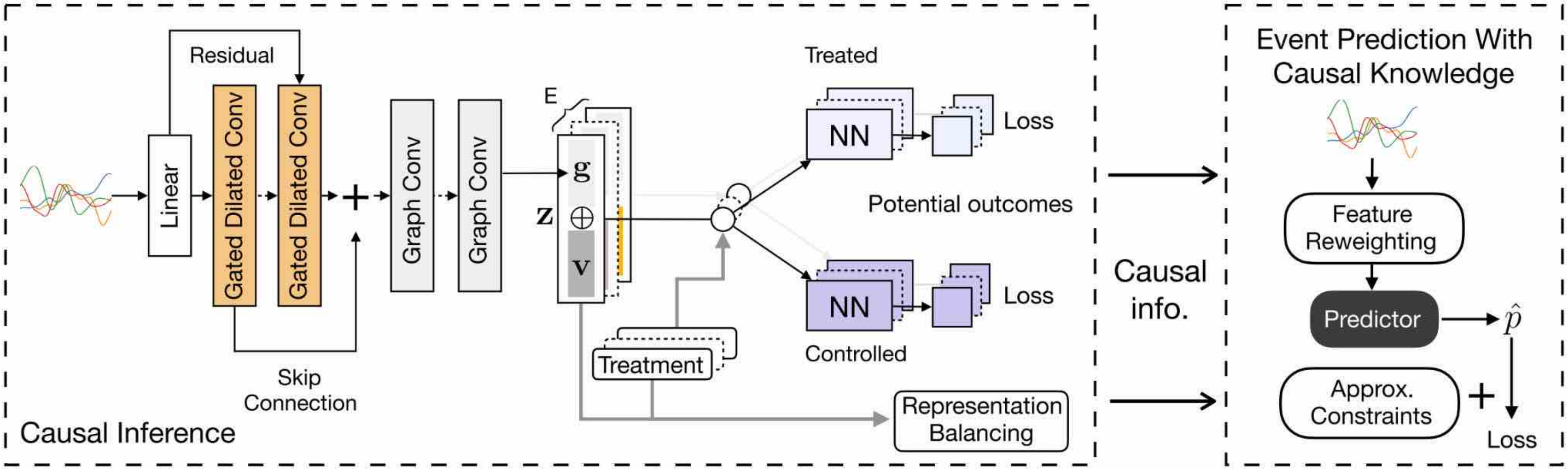


How does increased "*Appeal*" causally impact "*Protest*" at a time and location?

- Predict future events with the estimated causal information.

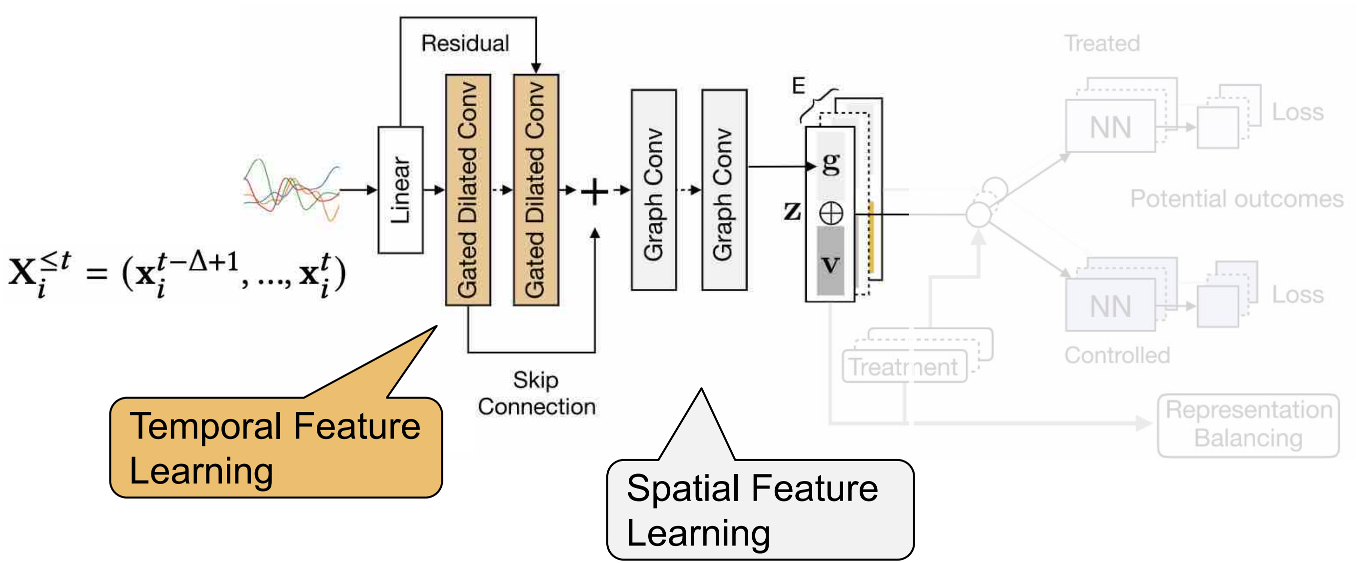
Proposed method

CAPE, which **incorporates causal inference into the prediction of future event** occurrences in a spatiotemporal environment.



Causal Inference

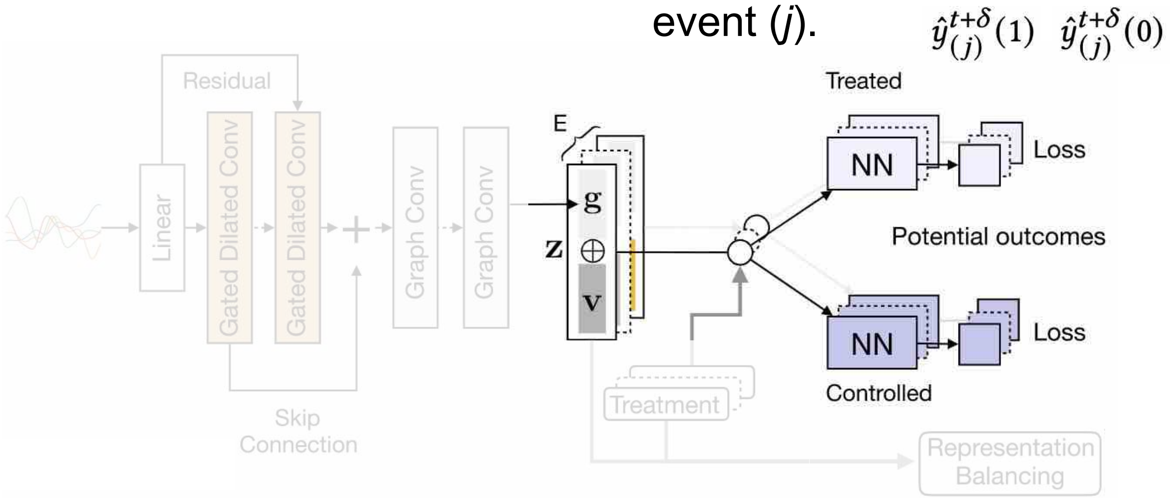
1. A spatiotemporal model learns hidden confounder representations using covariates.



Causal Inference

1. A spatiotemporal model learns hidden confounder representations using covariates.

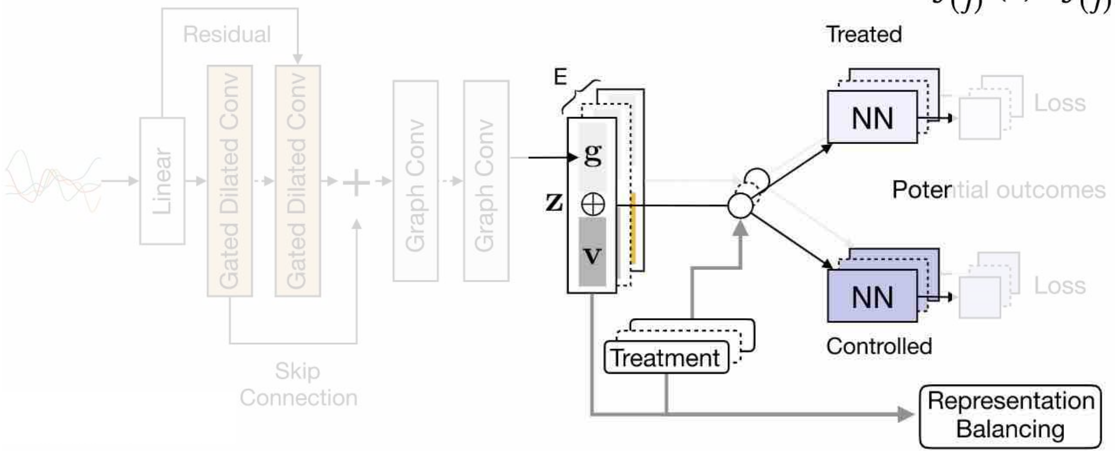
2. A multi-task framework estimates two potential outcomes for each treatment event (j).



Causal Inference

1. A spatiotemporal model learns hidden confounder representations using covariates.

2. A multi-task framework estimates two potential outcomes for each treatment event (*j*). $\hat{y}_{(j)}^{t+\delta}(1)$ $\hat{y}_{(j)}^{t+\delta}(0)$

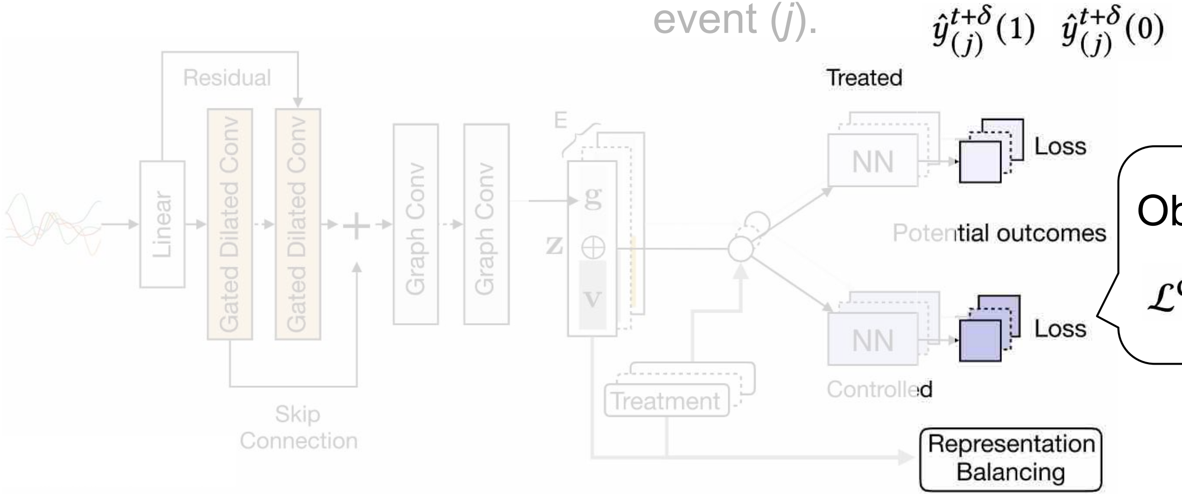


3. Representation balancing forces hidden confounders of treated and controlled groups to be similar. [Johansson, *et al.* 2016]

Causal Inference

1. A spatiotemporal model learns hidden confounder representations using covariates.

2. A multi-task framework estimates two potential outcomes for each treatment event (*j*).



Objective Function

$$\mathcal{L}^{\text{cau}} = \mathcal{L}^{\text{fact}} + \mathcal{L}^{\text{disc}}$$

3. Representation balancing forces hidden confounders of treated and controlled groups to be similar. [Johansson, *et al.* 2016]

Event Prediction with Causal Knowledge

Feature Reweighting

- Reweight input features **x** using gating variables $\beta^{t+\delta}$ obtained from estimated ITEs which quantify causal effects.

$$\tilde{\mathbf{x}}^t = \mathrm{FFN}(\mathbf{x}^t) \odot \boldsymbol{\beta}^{t+\delta} + \mathbf{x}^t$$

Approximation constraints

- A penalty is applied if a prediction exceeds the boundaries defined by estimated causal information

Event prediction

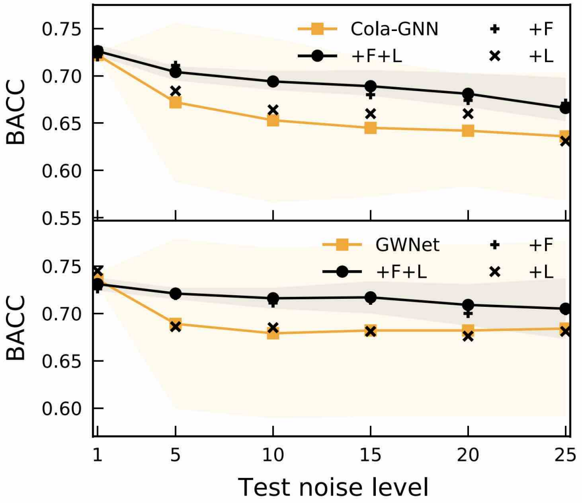
$$\mathcal{L}^{\mathrm{cstr}} = \sum_{t \in T} \sum_{i \in M} \mathrm{ReLU}\left(l_i^{t+\delta} - \hat{p}_i^{t+\delta}\right) + \mathrm{ReLU}\left(\hat{p}_i^{t+\delta} - u_i^{t+\delta}\right)$$

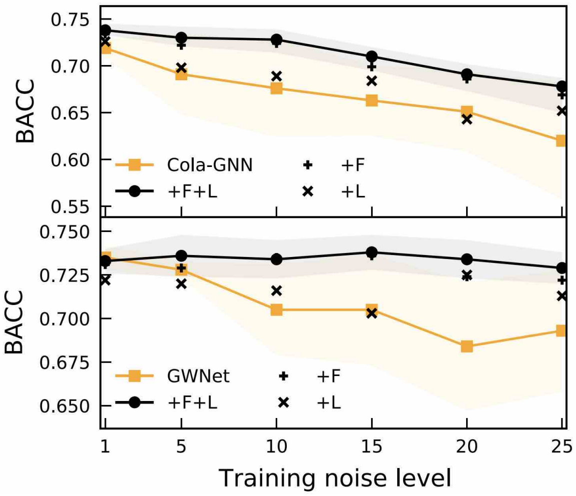
Min of predicted potential outcomes for all treatment events.

Max of predicted potential outcomes for all treatment events.

Robustness test in event prediction

Poisson noise is added into the test (left) and training sets (right) while keeping the others noise-free on the India dataset.





Adding both modules (+F+L) leads to a higher average BACC and lower variance.

Causality Enhanced Societal Event Forecasting With Heterogeneous Graph Learning [Deng et al. ICDM22]

Study the **causal factors of societal event** in the form of **topics**, and leverages those topics in dynamic heterogeneous contexts for predicting events.

Motivation

- Knowing the causes of past events can help humans reason about future events.
- Incorporating **causal information into graph learning** can enhance the representation learning of graph nodes by broadcasting causal information.

Deng, Songgaojun, Huzefa Rangwala, and Yue Ning. **Causality enhanced societal event forecasting with heterogeneous graph learning**. *ICDM*. IEEE, 2022.

Preliminaries

Heterogeneous graph

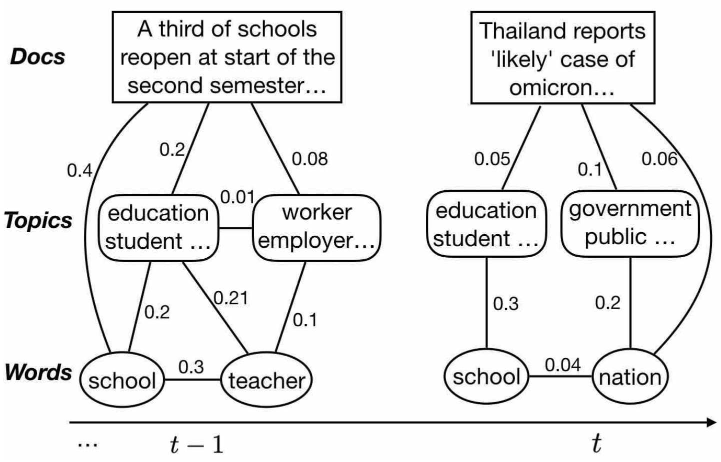
- A directed graph $G \subseteq (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R})$ with multiple types of nodes and edges.

Dynamic Heterogeneous Context Graph

- A type of heterogeneous graph where edges have timestamps $G[t] \subseteq (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R}, \mathcal{T})$

  E.g., a timestamped edge (e, t) = ((u, v), t) denotes the connection of two nodes u and v at time t.

Edge definition: word-word: PMI
word-doc: TF-IDF
topic-topic: Similarity
topic-doc: Probability
topic-word: Probability



An example of the dynamic heterogeneous context graph with three types of nodes: **word**, **topic**, and **docs**.

Preliminaries

Average Treatment Effect (ATE)

- The difference in the mean outcome between the treatment and control groups

Propensity Score Matching (PSM) [Caliendo, et al. 2008]

- A statistical matching technique that estimates the ATE.
- Reduce the bias due to confounding variables that could be found in treatment effect estimation.

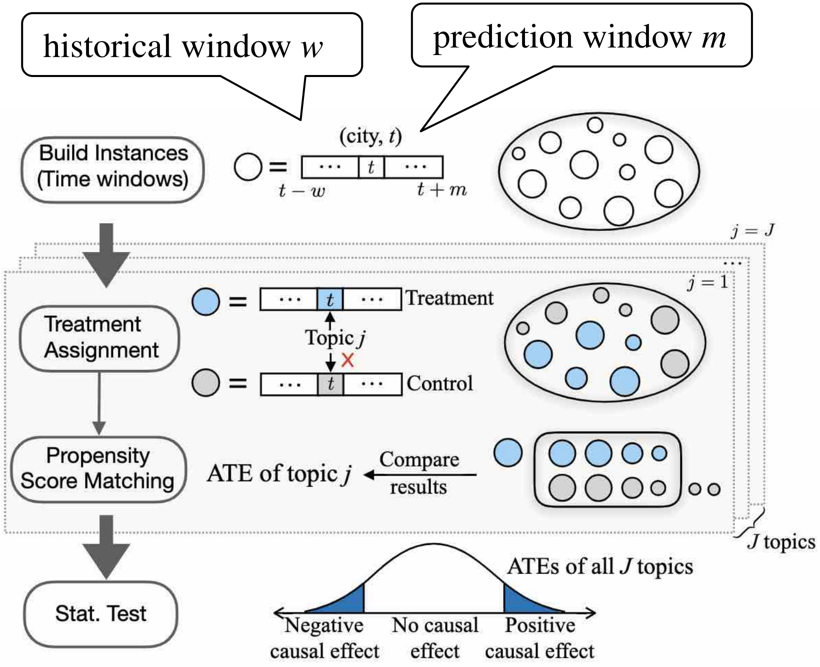
Problem Statement

- Predict the event by taking a dynamic heterogeneous context graph as input. **Topics** that might cause events (e.g., protest) are discovered and identified in the input graph.

Caliendo et al.. **Some practical guidance for the implementation of propensity score matching**. *Journal of economic surveys* 2008.

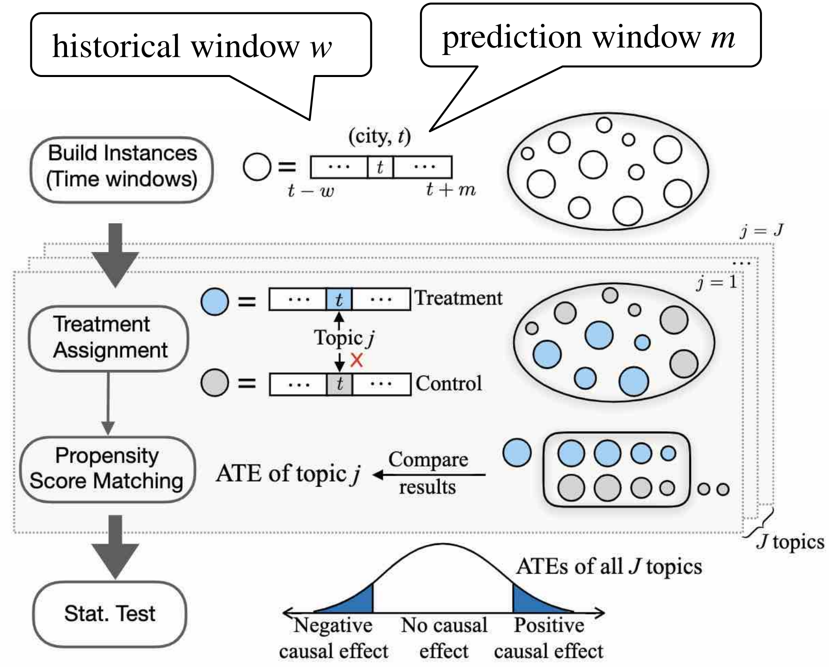
Methodology - Causal analysis

Discovering causal topics



Methodology - Causal analysis

Discovering causal topics



Social environments are changing!

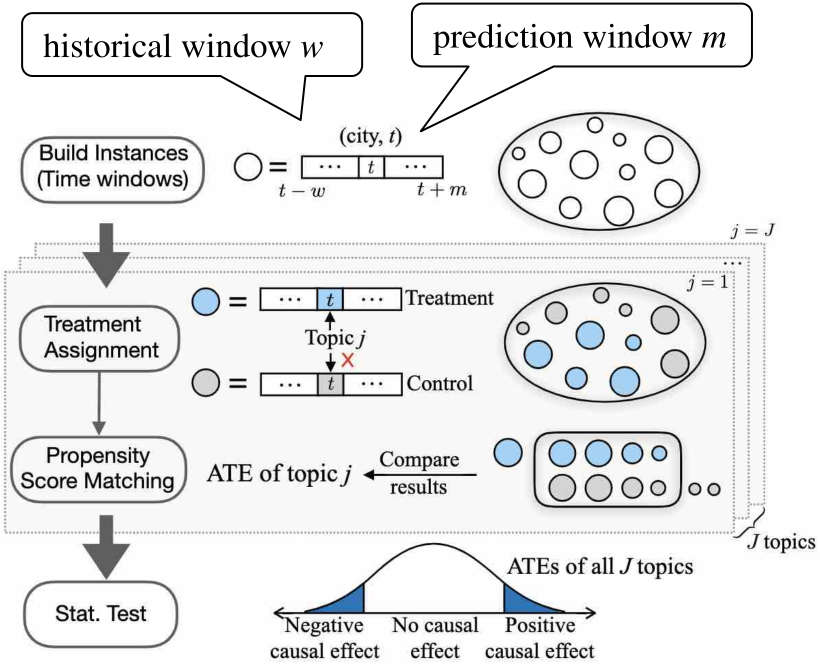
Event factors may change!

Protest factor: *Education* *Election* *Foreign Affairs*

2012 → 2016

Methodology - Causal analysis

Discovering causal topics



Social environments are changing!

Event factors may change!

Protest factor: *Education* *Election* *Foreign Affairs*

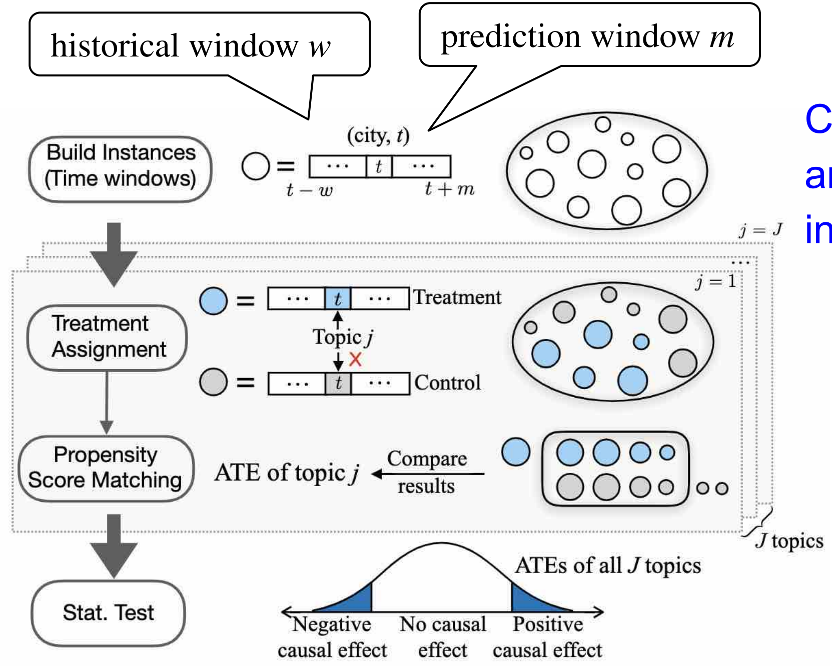
2012 2016

Discovering evolving and multi-view causal topics

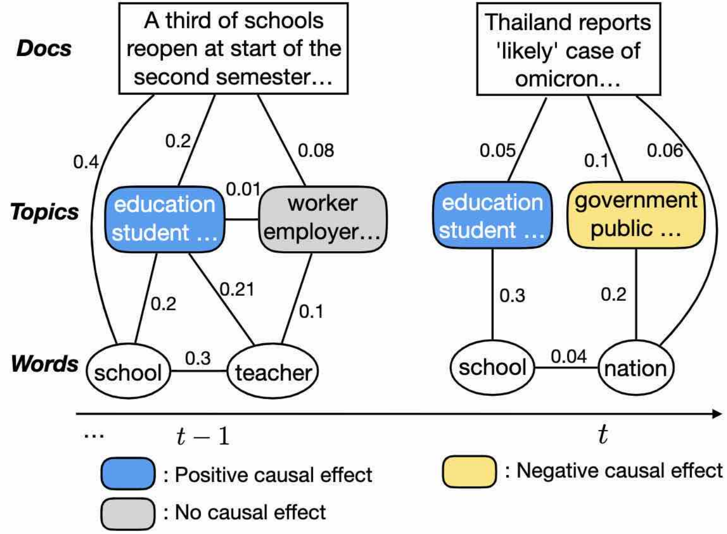
- Obtain time-sensitive causal topics for each season (i.e., 3 months) by using time windows in the past.
- Vary the prediction window size m (e.g., 3, 7, and 14) to obtain topics that have long-term and short-term effects.

Methodology - Causal analysis

Discovering causal topics

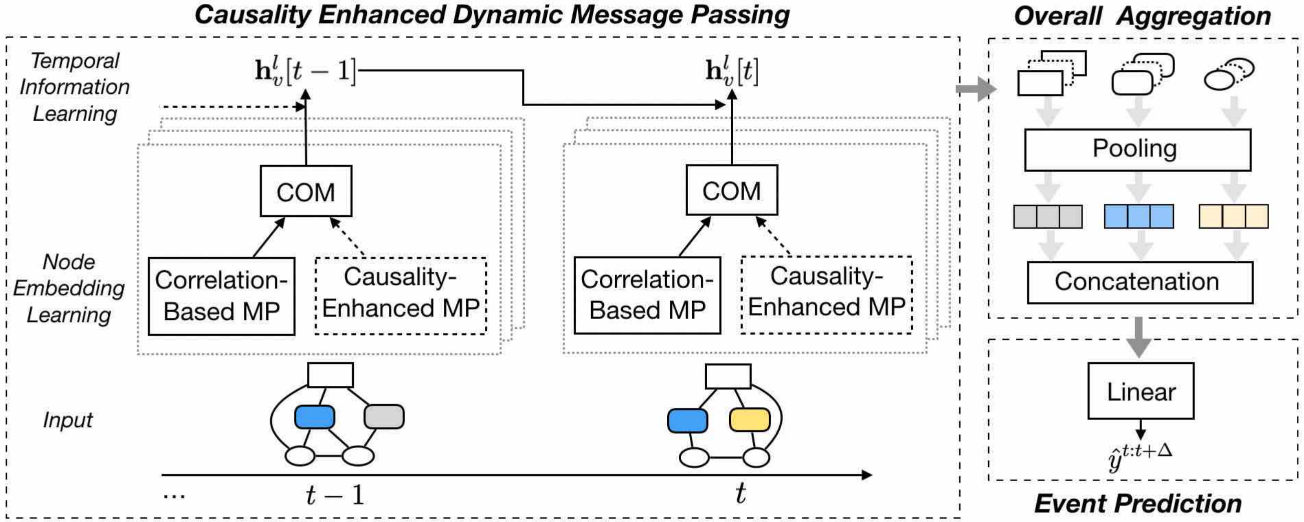


Causal topics are identified in input graphs



Methodology - Event prediction

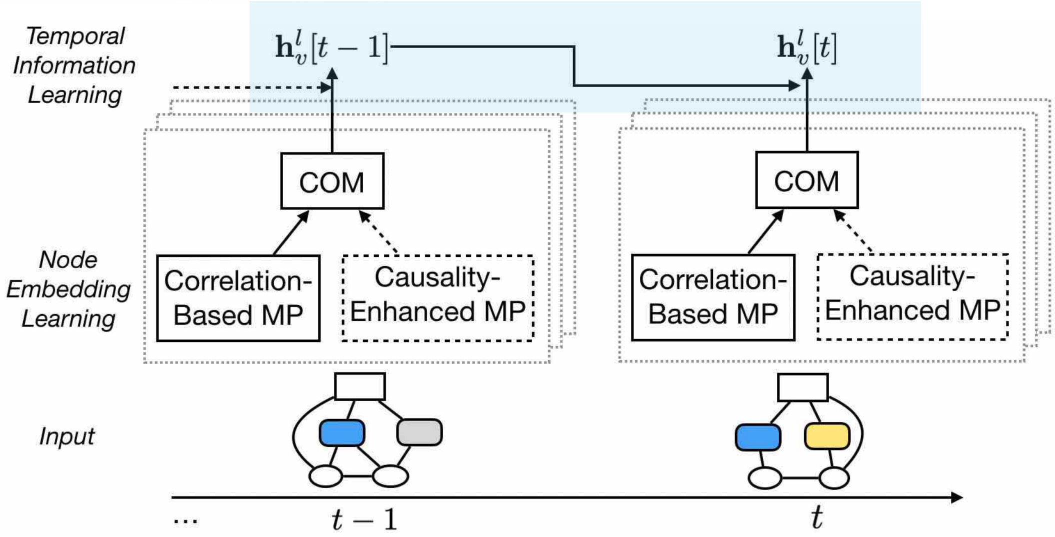
Dynamic **h**eterogeneous **g**raph model with **c**ausality enhanced node representations (HGC)



Causality Enhanced Dynamic Message Passing

1. A temporal information learning module (TEM) incorporates node embedding learned in the past graph.

$$\mathbf{h}_v^l[t] = \alpha_{\tau(v)} \cdot \tilde{\mathbf{h}}_v^l[t] + (1 - \alpha_{\tau(v)}) \cdot \mathbf{h}_v^l[< t]$$



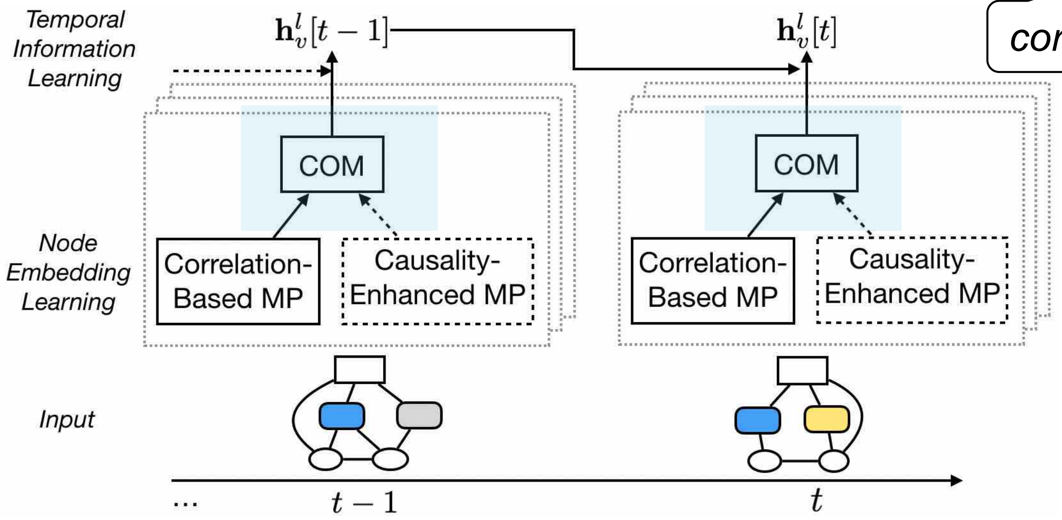
Causality Enhanced Dynamic Message Passing

1. A temporal information learning module (TEM) incorporates node embedding learned in the past graph.

$$\mathbf{h}_v^l[t] = \alpha_{\tau(v)} \cdot \tilde{\mathbf{h}}_v^l[t] + (1 - \alpha_{\tau(v)}) \cdot \mathbf{h}_v^l[< t]$$

2. A combination module (COM) incorporates causal information.

$$\tilde{\mathbf{h}}_v^l[t] = \mathrm{ReLU}(\underbrace{\mathbf{o}_v^l[t]}_{\text{correlation}} + \underbrace{\mathbf{c}_v^l[t]}_{\text{Causal}})$$



Causality Enhanced Dynamic Message Passing

1. A temporal information learning module (TEM) incorporates node embedding learned in the past graph.

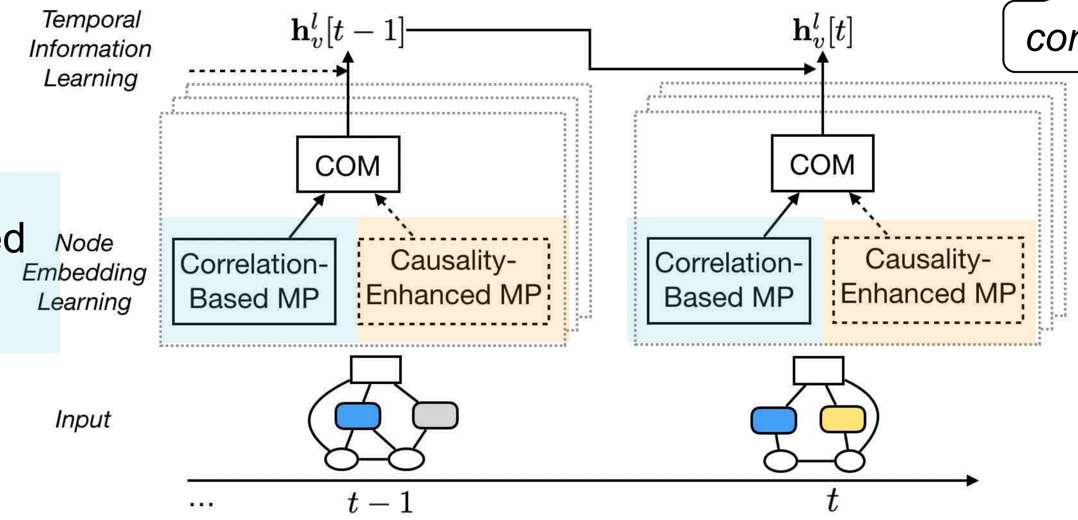
$$\mathbf{h}_v^l[t] = \alpha_{\tau(v)} \cdot \tilde{\mathbf{h}}_v^l[t] + (1 - \alpha_{\tau(v)}) \cdot \mathbf{h}_v^l[< t]$$

2. A combination module (COM) incorporates causal information.

$$\tilde{\mathbf{h}}_v^l[t] = \mathrm{ReLU}(\mathbf{o}_v^l[t] + \mathbf{c}_v^l[t])$$

correlation

Causal



3. Correlation-based message passing.

4. Causality-aware message passing.

Experimental results of event prediction

	Train ratio	Metric	GAT	EvolveGCN	RGCN	HGT	**HGC**
THA	60%	F1	0.713±0.038	0.717±0.019	0.754±0.014	0.803±0.03	**0.839±0.023**
		BACC	0.767±0.029	0.765±0.015	0.795±0.012	0.838±0.024	**0.867±0.019**
	40%	F1	0.662±0.028	0.628±0.060	0.719±0.013	0.765±0.025	**0.796±0.019**
		BACC	0.711±0.019	0.698±0.033	0.759±0.007	0.800±0.021	**0.826±0.015**
AFG	60%	F1	0.512±0.056	0.576±0.043	0.599±0.008	0.650±0.029	**0.700±0.013**
		BACC	0.641±0.015	0.673±0.025	0.684±0.011	0.721±0.021	**0.758±0.010**
	40%	F1	0.544±0.071	0.541±0.029	0.610±0.023	0.629±0.028	**0.683±0.011**
		BACC	0.657±0.036	0.635±0.012	0.686±0.023	0.711±0.019	**0.751±0.009**

Graph types the model handles:

GAT	EvolveGCN	RGCN	HGT	HGC
Static homo	Dynamic homo	Static homo	Static hetero	**Dynamic hetero**

Causal topic analysis

Qualities of causal topics

	Sig. level	#Pos	#Neg	F1	BACC
THA	99%	2	1	0.839±0.037	0.868±0.031
	95%	4	2	**0.847±0.022**	**0.875±0.020**
	90%	5	3	0.829±0.023	0.859±0.020
	80%	8	7	0.834±0.009	0.863±0.007
AFG	99%	3	1	**0.700±0.013**	**0.758±0.010**
	95%	4	1	0.688±0.035	0.750±0.027
	90%	5	2	0.694±0.021	0.754±0.017
	80%	7	7	0.660±0.053	0.734±0.034

The average # of topics per sample that has a positive/negative causal effect on future protests.

There is a trade-off between

- involving fewer causal topics with high confidence or
- involving more causal topics that may sacrifice confidence.

Challenges and considerations

GNN-based models can capture intricate interactions and dependencies in complex event data, thereby enhancing prediction accuracy.

Limitations

- The **computational demands** for training GNNs on massive graphs pose scalability challenges, when considering the vast volume of event data.
- **Generalizing to unseen** graphs. Low ability to capture relevant patterns or dependencies in unfamiliar event contexts, such as underrepresented regions with limited training data.

Large language models!

Coffee Break

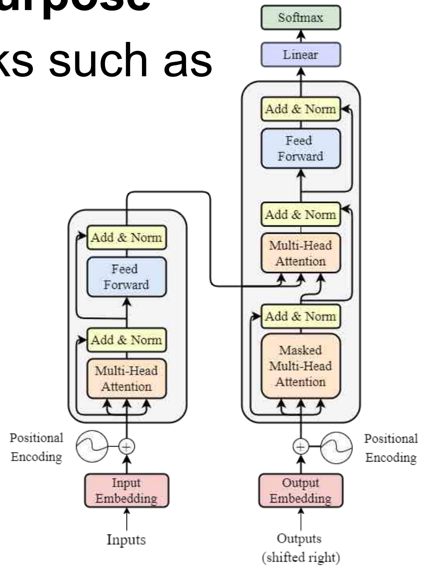
(15 minutes)

Next: Part 2: Large Language Model (LLM)-based methods

Part 2: Large Language Model (LLM)-based methods

Large language models

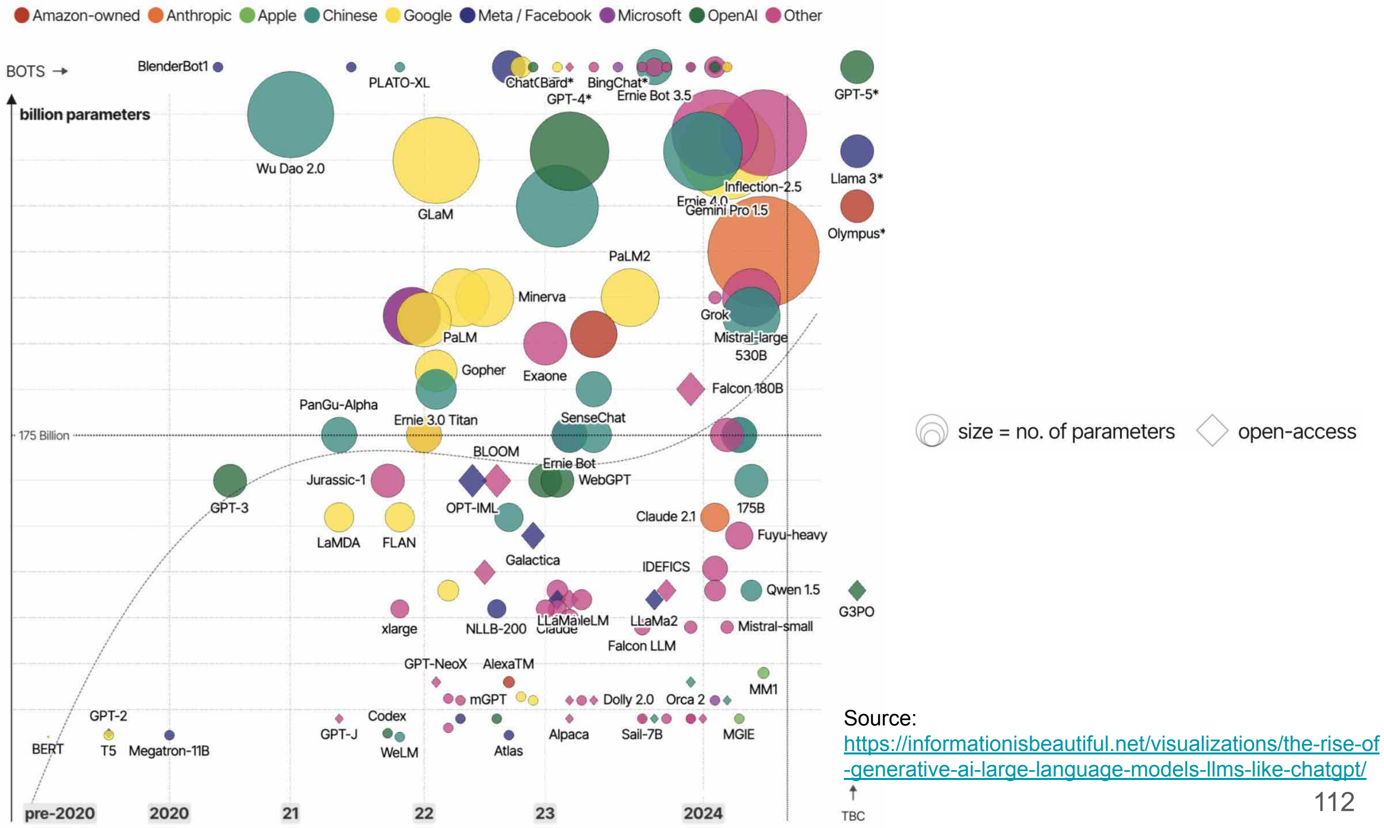
A computational model notable for its ability to achieve **general-purpose language generation** and other natural language processing tasks such as classification.



A brief history

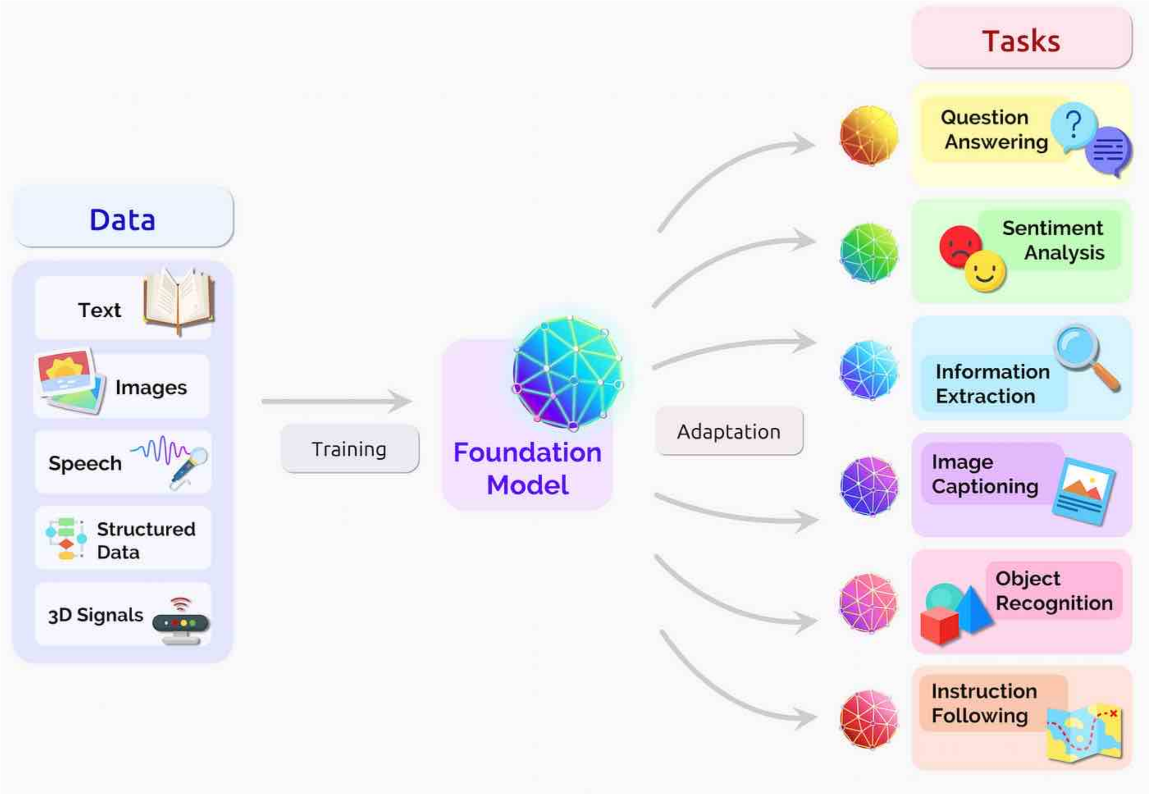
- 2017 Attention Is All You Need
- 2018 BERT was introduced and quickly became "ubiquitous"
- 2018 OpenAI introduced Decoder only GPT-1
- 2019-2022 OpenAI released GPT-2, GPT-3
- Since 2022, source-available models gain popularity, e.g, LLaMA
- 2023, GPT-4 was praised for its increased accuracy and multimodal capabilities
- …

The rise of large language models (LLMs)



Source:
https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-language-models-llms-like-chatgpt/

Overview of LLM applications



Source: https://arxiv.org/pdf/2108.07258

LLMs for event prediction

Language Models Can Improve Event Prediction by Few-shot Abductive Reasoning [Shi et al. NeurIPS 2024]

Propose the first work that **integrates LLMs into event prediction.**

Motivation

- **Event** data usually comes with **texts**, how to effectively use textual information for event prediction?
- LLMs have shown astonishing performance on various reasoning tasks.
- **Can LLMs reason about real-world events** and help improve the prediction performance of event sequence models.

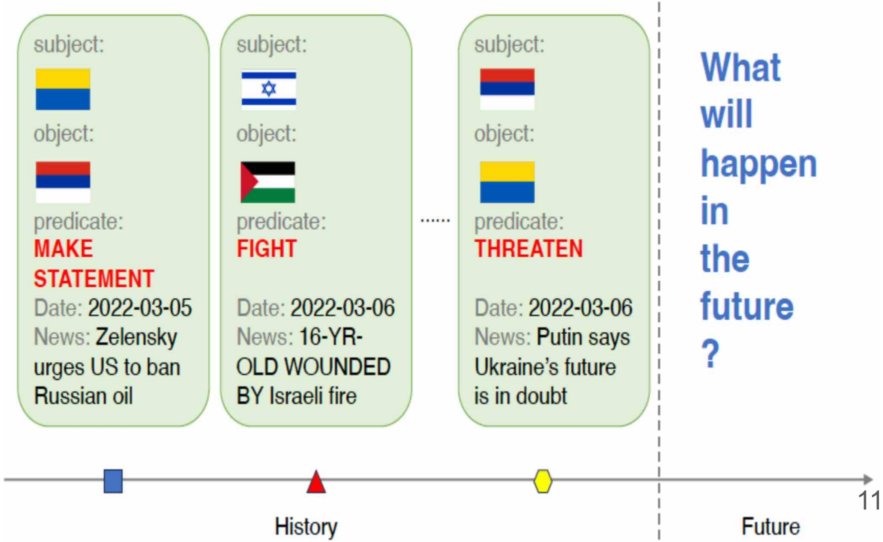
Shi, Xiaoming, et al. **Language models can improve event prediction by few-shot abductive reasoning**. *NeurIPS* 37 (2023).

Event sequence modeling

Suppose given an event sequence $(t_1, k_1), (t_2, k_2), \ldots$, where $0 < t_1 < t_2 < \ldots$ are **times** of occurrence and each $k_i \in \mathcal{K}$ is a discrete **event type**.

Goal: Predict the **next event** for a given history of events.

$$\mathcal{H}_i = (t_1, k_1), \ldots, (t_{i-1}, k_{i-1})$$



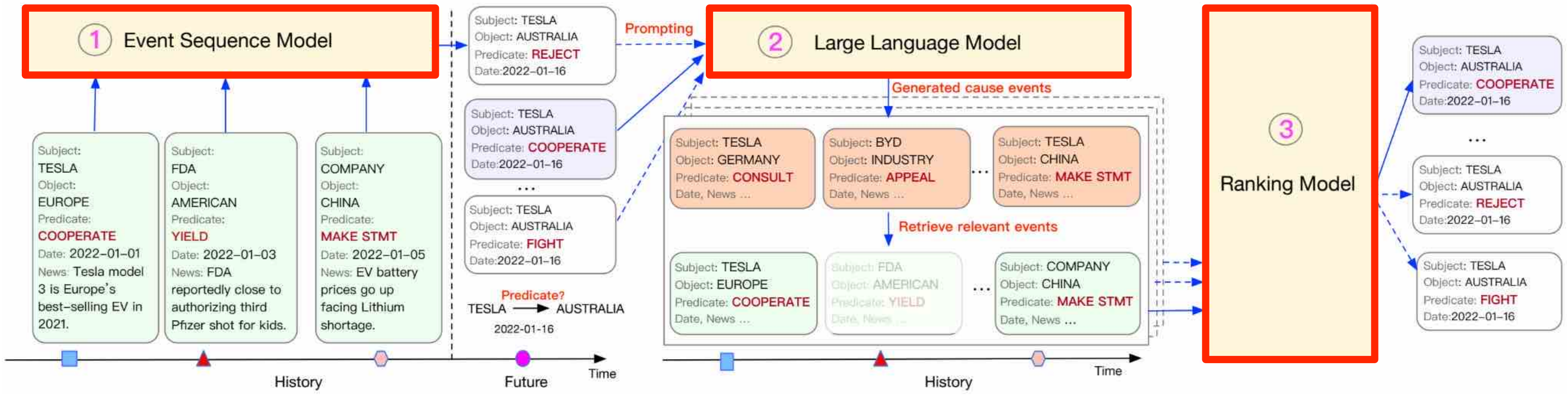
It consists of two subtasks:

- Predict **time *t*** of the next event
- Then predict **type *k*** of the next event

LAMP: Large Language Model in Event Prediction

LAMP has three key components

1. A base event sequence model **proposes candidate predictions**.
2. A large language model **performs abductive reasoning**.
3. A ranking model **learns to scores predictions**.



Phase-I: Proposing Predictions

Propose candidate predictions based on history $\mathcal{H}_i = (t_1, k_1), \ldots, (t_{i-1}, k_{i-1})$

Transformer based model
ANHP [Yang et al 2022]

- **Time** prediction
  - Draw *L* i.i.d. samples from the base model $\hat{t}_i^{(1)}, \ldots, \hat{t}_i^{(L)}$
  - Usually, we use the **average** of the samples $\hat{t}_i = \frac{1}{L}\sum_{\ell=1}^{L} \hat{t}_i^{(\ell)}$

- **Type** prediction given the ground-truth time t_i
  - Find *M* event types $\hat{k}_i^{(1)}, \ldots, \hat{k}_i^{(M)}$ ordered by their **intensity** at the time from the base model.
  - Usually, we select the type with the **highest** intensity.

Base model is imperfect and and we want more informed predictions.

Yang, C., Mei, H., and Eisner, J. **Transformer embeddings of irregularly spaced events and their participants**. ICLR, 2022.

Phase-I: Proposing Predictions

Propose candidate predictions based on history $\mathcal{H}_i = (t_1, k_1), \ldots, (t_{i-1}, k_{i-1})$

- **Time** prediction
  - Draw *L* i.i.d. samples from the base model $\hat{t}_i^{(1)}, \ldots, \hat{t}_i^{(L)}$
  - Treat all the *L+1* samples as **candidates**, including the average $\hat{t}_i = \frac{1}{L}\sum_{\ell=1}^{L} \hat{t}_i^{(\ell)}$

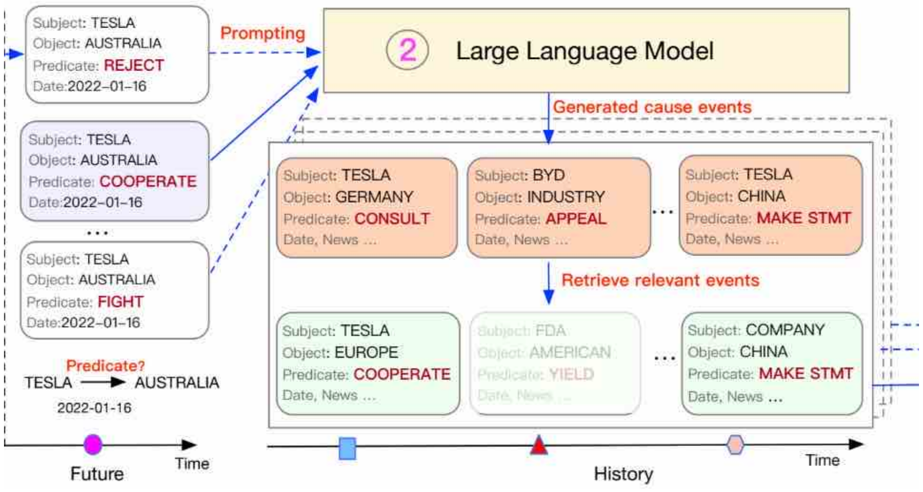
- **Type** prediction given the ground-truth time t_i
  - Find *M* event types $\hat{k}_i^{(1)}, \ldots, \hat{k}_i^{(M)}$ ordered by their **intensity** at the time from the base model.
  - Keep most probable *M* full events $\{(\hat{t}_i^{(\ell)}, \hat{k}_i^{(\ell,m)})\}_{m=1}^{M}$ for each time proposal $\hat{t}_i^{(\ell)}$

How to deal with all the (*t*, *k*) event candidates?

Phase-II: Prompting LLM to Perform Abductive Reasoning

For each proposed event (t, k), the framework **selects a set of previous events** from its full history as its supporting evidence **e**(t, k), using an **LLM** (e.g., GPT-3.5)

- Prompt the LLM to imagine some **possible cause events** that could explain the occurrence of this proposal.
- Search for the most similar ones if not exactly match.
  - Cosine similarity of embeddings of actual and cause events. (subject-predicate-object, time, text)



Prompt formats

Explain the task, concepts, examples and post the query.

```
I want you to do the reasoning over
social events. I given you an effect
event and you give me four or five
cause events. An effect event is an
event that happens. A cause event is
believed to be one of the causes that
have triggerred the effect event to
happen. Each event consists of a time,
a type (that includes subject,
predicate, object), and a news headline
 describing the event.

The predicates are restricted to the 20
 options below.
1. MAKE STATEMENT
⋮  // Full list are in Appendix E.4.
20. ENGAGE IN MASS VIOLENCE

Now I give you 10 examples. In each
example, the first event is the effect
and the next several events are the
causes that happened earlier.
⋮  // Examples are in Listing 2.
Now please generate possible causes for

effect
predicate: CONSULT
time: 2022-07-05
subject: CHINA PM
object: YELLEN
```

Listing 1: Format of our LLM prompt.

Prompt formats

Here, each example includes 1 effect and 5 cause events.

```
I want you to do the reasoning over
social events. I given you an effect
event and you give me four or five
cause events. An effect event is an
event that happens. A cause event is
believed to be one of the causes that
have triggerred the effect event to
happen. Each event consists of a time,
a type (that includes subject,
predicate, object), and a news headline
 describing the event.

The predicates are restricted to the 20
 options below.
1. MAKE STATEMENT
⋮  // Full list are in Appendix E.4.
20. ENGAGE IN MASS VIOLENCE

Now I give you 10 examples. In each
example, the first event is the effect
and the next several events are the
causes that happened earlier.
⋮  // Examples are in Listing 2.
Now please generate possible causes for

effect
predicate: CONSULT
time: 2022-07-05
subject: CHINA PM
object: YELLEN
```

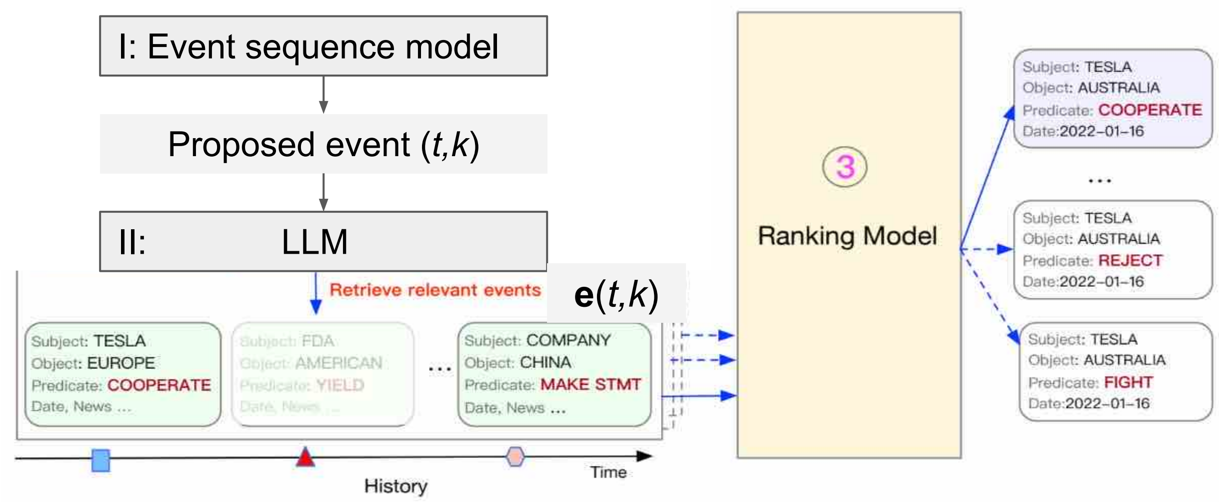
Listing 1: Format of our LLM prompt.

```
## Example 1

effect
predicate: APPEAL
time: 2022-04-23
subject: GERMANY
object: GREEN PROJECT

reasoning:
------------------------
cause event 1
predicate: REDUCE RELATIONS
time: 2022-04-21
subject: EUROPE
object: RUSSIA
headline: Europe determined to ban
Russian energy exports.

cause event 2
predicate: DISAPPROVE
time: 2022-03-16
subject: EUROPE
object: RUSSIAN
headline: Europe can endure painful
transition to live without Russian oil.

⋮  // Other causes are in Appendix E.4.

## Example 2
⋮  // Other examples in Appendix E.4.
```

Listing 2: Few-shot examples in our prompt.

Phase-III: Ranking Proposals

Score each proposed event (t, k) based on the compatibility with its retrieved evidence $\mathbf{e}(t, k)$.



Phase-III: Ranking Proposals

Score each proposed event (t, k) based on the compatibility with its retrieved evidence $\mathbf{e}(t, k)$.

- Calculate the score of each proposed event (t, k)

$$s_{\text{event}}(t, k) \stackrel{\text{def}}{=} \exp\left(c\left((t, k), \mathbf{e}(t, k)\right)\right)$$

A **high** score means the proposal is **strongly supported** by its retrieved evidence; more likely to happen at time *t.*

- Measure the overall belief for candidate time t, given the most probable M events at time t $\{(t, k^{(m)})\}_{m=1}^{M}$

$$s_{\text{time}}(t) \stackrel{\text{def}}{=} \sum_{m=1}^{M} s_{\text{event}}(t, k^{(m)})$$

Time prediction: time with highest $s_{\text{time}}(\hat{t}^{(\ell)})$ **Type prediction**: type with highest $s_{\text{event}}(t, \hat{k}^{(m)})$

Model architecture and training

Model architecture

- Function *c* has a continuous-time Transformer architecture [Yang et al. ICLR 2022].
  - Its attention mechanism learns to disregard irrelevant retrieved events.
  - its sophisticated handling of time may be helpful (e.g., recent evidence events matters more than ancient events)

Training

- Train the ranking model by maximizing the objective

$$J \stackrel{\text{def}}{=} J_{\text{actual}} + \beta J_{\text{no}}$$

Scores of the events that have actually happened.

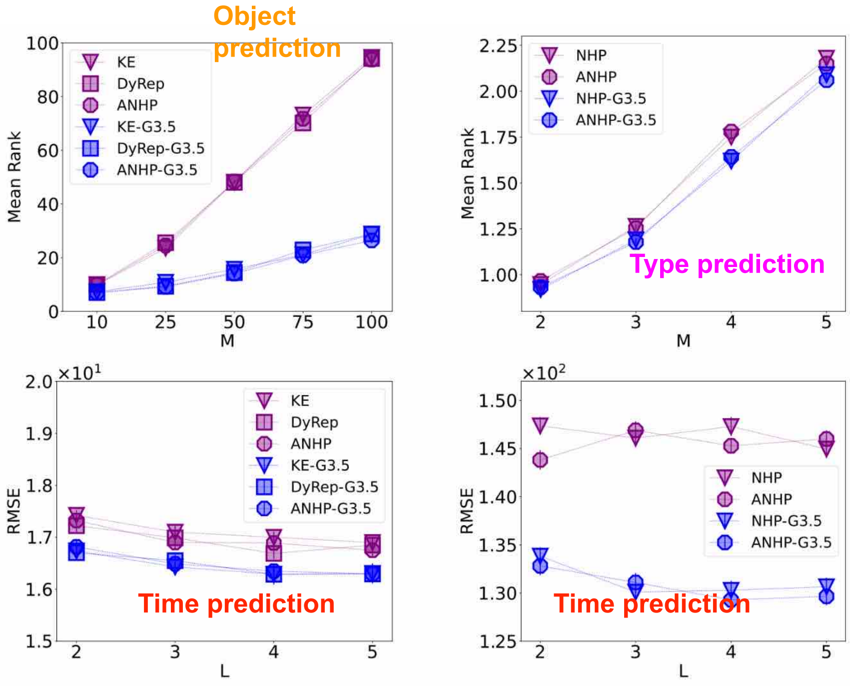
Negative scores of the non-events at sampled times.

Yang, C., Mei, H., and Eisner, J. **Transformer embeddings of irregularly spaced events and their participants**. ICLR 2022.

Experiments on real-world datasets

Mean Rank (MR) ↓: average rank of the ground-truth type/object in the list.

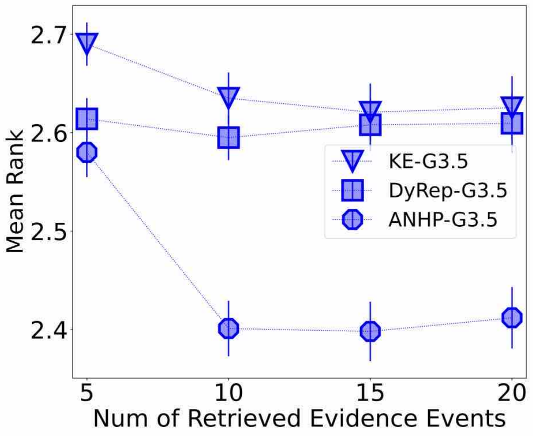
RMSE ↓: how close top-ranked time prediction is to the ground-truth time.

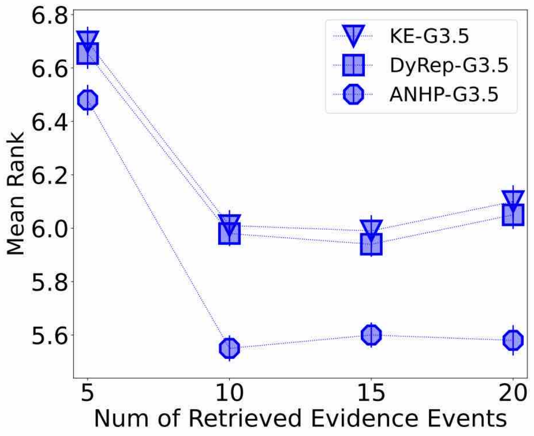


(b) ICEWS. (c) Amazon Review.

How many evidence events do we need?

Evidence events are generated by LLM at Phase II.
Effect of the number of evidence events on predicate (left) and object (right) prediction on GDELT.



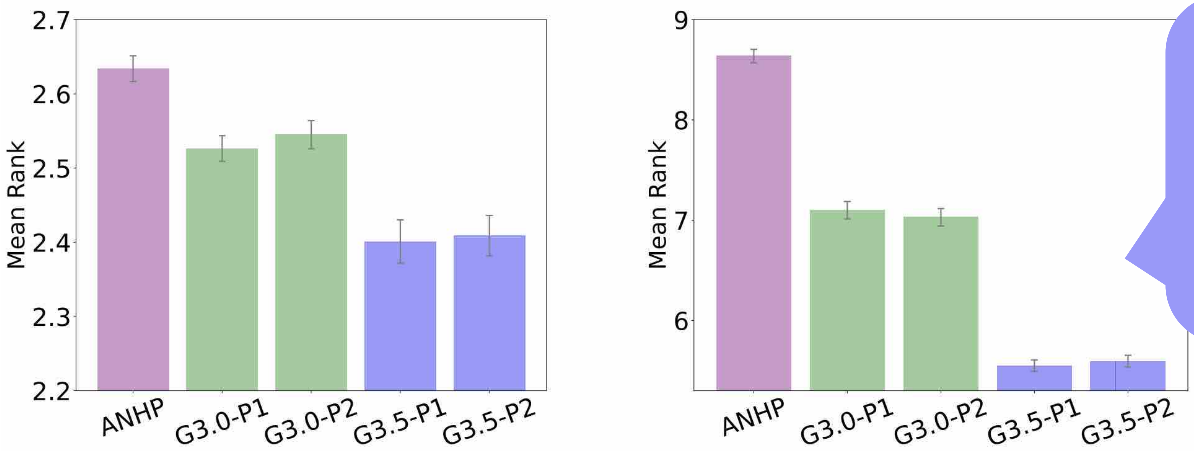


Retrieve too many “evidences” event might not be helpful.

Prompt design – choice of demonstrations

P1: the set of demonstrations used throughout the main experiments.

P2: randomly replace half of the demonstrations in P1.



Different demonstrations only have a slight effect on performance. (Results are on GDELT)

Does the GPT already have the ability to do reasoning?

About data leakage

GPT models are trained on online text data up to 2021.

For GDELT or ICEWS, the data used in experiments are from 2022.

For Amazon review, the data are from 2014 to 2016. To verify data leakage, a conversation was made with GPT-3.5

input: We are using Amazon Review Data, published by Jianmo Ni, et al., 2018, which is firstly used in the paper "Justifying recommendations using distantly-labeled reviews and fined-grained aspects", EMNLP 2019.

The dataset includes millions of reviews (ratings, text, helpfulness votes), product metadata (descriptions, category information, price, brand, and image features) in the range May 1996 - Oct 2018. Do you remember any of the review information if I give you a review id?

GPT-3.5 output: As an AI language model, I am not capable of remembering any
re … fic
pi … tion
fr … ven a review ID.

…not capable of remembering any review information …
I can search …

input: can you recall any information with reviewer ID A28GK1G2KDXHRP on date 04 17, 2014.

GPT-3.5 output: I apologize, but the Amazon Review dataset does not include information about the specific date when a review was posted. It only includes the year and month when the review was posted. Therefore, I cannot provide you
wi … April
17

I apologize,... I cannot provide you with any information about reviews …

Large Language Models as Event Forecasters [Zhang et al. 2024]

- Existing challenges and motivations for leveraging LLMs
  - **L**arge languag**e** models **a**s event **f**orecasters (LEAF)

- LEAF: LLMs for object prediction
  - Object prediction as a ranking task
  - Object prediction as a generative task

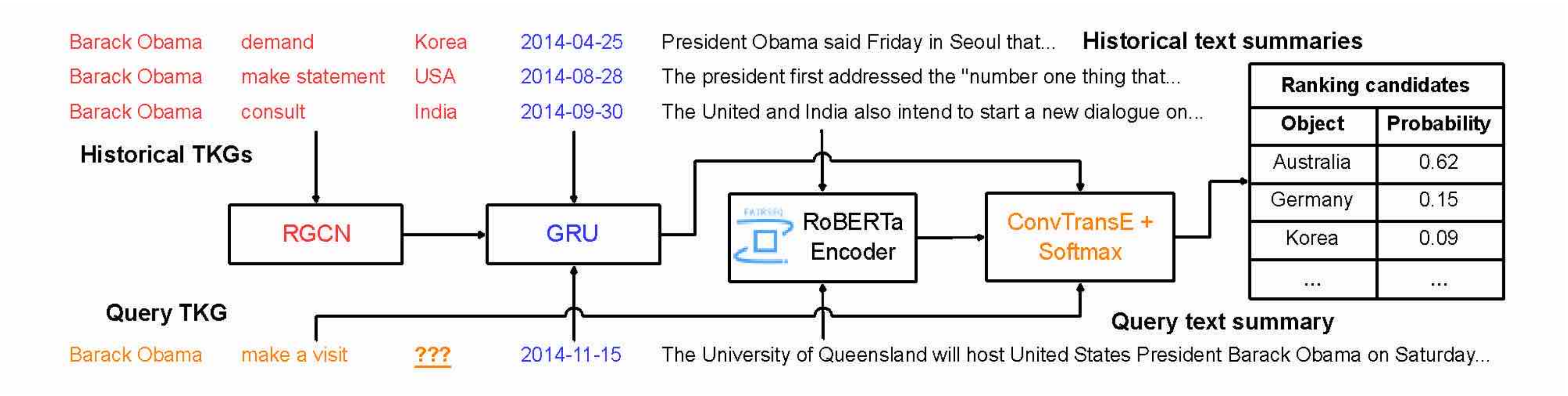
- LEAF: LLMs for multi-event forecasting
  - Quintuple-level prompt encoding and multi-label binary classification

Libo Zhang and Yue Ning. 2024. **Large Language Models as Event Forecasters**. arXiv 2024.

Challenges and motivations

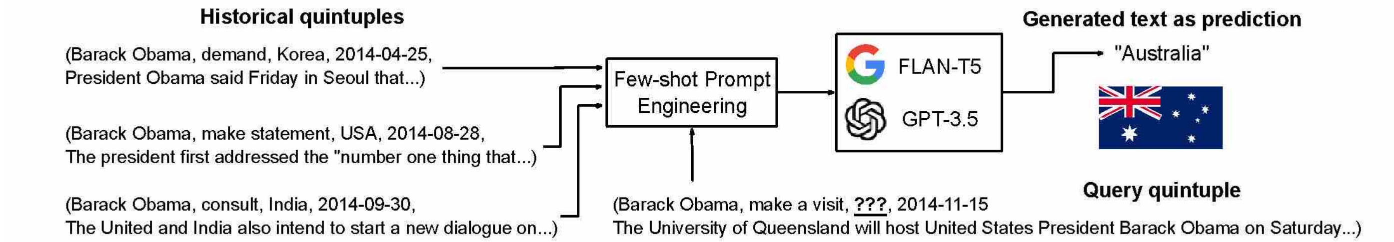
- Underestimation of individual TKG quintuple contextualization
  - Contextual meaning of each short phrase
- Unfamiliar domain-specific knowledge for closed-source LLMs
  - Trade-off between detailed prompt engineering and token-related pricing
- Limited maximum input context length for open-source LLMs
  - Misaligned goals between prediction accuracy and instruction fine-tuning

Object prediction as a ranking task



- Encoder-only LLM: RoBERTa
- The RoBERTa encoder is separately fine-tuned with masked language modeling loss by simply concatenating text summaries from all TKG quintuples
- Only RGCN, GRU, and ConvTransE are involved during ranking optimization

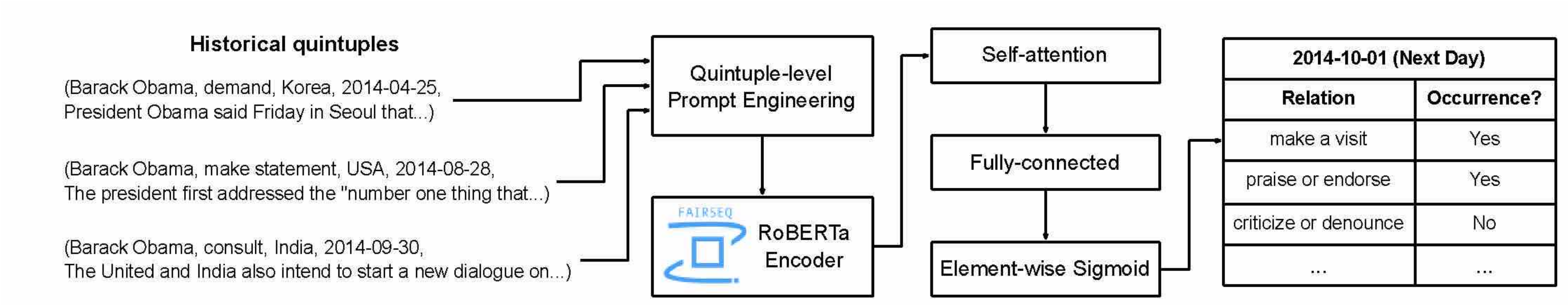
Object prediction as a generative task



- Encoder-decoder LLM: FLAN-T5, or
- Decoder-only generative LLM: GPT-3.5
- A standard question-answering task
- The question is formulated as a prompt, with in-context learning examples and the query to be completed
- The answer is the ground-truth object
- Feed text in, and get text out

Name	Prompt Template
Original	I ask you to perform an object prediction task after I provide you with five examples. Each example is a knowledge quintuple containing two entities, a relation, a timestamp, and a brief text summary. Each knowledge quintuple is strictly formatted as (subject entity, relation, object entity, timestamp, text summary). For the object prediction task, you should predict the missing object entity based on the other four available elements. Now I give you five examples. ## Example 1 (⟨SUBJECT 1⟩, ⟨RELATION 1⟩, ⟨MISSING OBJECT ENTITY⟩, ⟨TIMESTAMP 1⟩, ⟨TEXT SUMMARY 1⟩) \n The ⟨MISSING OBJECT ENTITY⟩ is: ⟨OBJECT 1⟩ \n ⋮ ## Example 5 (⟨SUBJECT 5⟩, ⟨RELATION 5⟩, ⟨MISSING OBJECT ENTITY⟩, ⟨TIMESTAMP 5⟩, ⟨TEXT SUMMARY 5⟩) \n The ⟨MISSING OBJECT ENTITY⟩ is: ⟨OBJECT 5⟩ \n Now I give you a query: (⟨SUBJECT 6⟩, ⟨RELATION 6⟩, ⟨MISSING OBJECT ENTITY⟩, ⟨TIMESTAMP 6⟩, ⟨TEXT SUMMARY 6⟩) \n Please predict the missing object entity. You are allowed to predict new object entity which you have never seen in examples. The correct object entity is:
Zero-shot	Remove all five in-context learning examples in the original prompt.
No-text	Remove all text summaries of five in-context learning examples and the query in the original prompt.

LLMs for multi-event forecasting



- Encoder-only LLM: RoBERTa
- Each historical quintuple is formulated as a prompt for RoBERTa encoding
- Self-attention mechanism is applied to weigh the significance differences among different quintuples within one historical day
- The dimension tracing back to multiple days is collapsed through aggregation
- Each relation's occurrence potential is evaluated through element-wise Sigmoid activation

Experiments: object prediction, ranking

Table 2: Model architectures for object prediction

Model	GNN	RNN	Decoder	How to handle text?
ConvTransE (Shang et al. 2019)	N/A	N/A	ConvTransE × 1	N/A
SeCoGD, LDA, 5 (Ma et al. 2023)	RGCN × 5	GRU × 5	ConvTransE × 5	Context clusters
Baseline w/o LLM	RGCN × 1	GRU × 1	ConvTransE × 1	N/A
LEAF-OP (ours)	RGCN × 1	GRU × 1	ConvTransE × 1	Query embeddings

Table 3: Object prediction results as a ranking task on the ICEWS dataset

Model	Afghanistan			India			Russia		
Metric: Hits@	1	3	10	1	3	10	1	3	10
Historical sequence length = 3 (back to 3 days), fine-tuned RoBERTa-base as an encoder-only LLM									
Baseline w/o LLM	0.1538	0.3137	0.5408	0.1704	0.3125	0.4952	0.1332	0.2345	0.3767
SeCoGD, LDA, 5	0.1878	0.3570	0.5740	0.2064	0.3554	0.5357	0.1768	0.2909	0.4351
ConvTransE	0.1235	0.2704	0.4916	0.1521	0.2821	0.4600	0.1009	0.1791	0.3078
LEAF-OP (ours)	0.3691	0.5630	0.7317	0.3675	0.5507	0.7233	0.3751	0.5390	0.6831
Historical sequence length = 7 (back to 7 days), fine-tuned RoBERTa-base as an encoder-only LLM									
Baseline w/o LLM	0.1551	0.3298	0.5544	0.1724	0.3175	0.5004	0.1335	0.2349	0.3781
SeCoGD, LDA, 5	0.1833	0.3652	0.5862	0.2056	0.3516	0.5352	0.1661	0.2823	0.4433
ConvTransE	0.1243	0.2759	0.4909	0.1544	0.2911	0.4740	0.0973	0.1804	0.2971
LEAF-OP (ours)	**0.3861**	**0.5884**	**0.7664**	**0.3935**	**0.5831**	**0.7454**	**0.3861**	**0.5590**	**0.7077**

- Evaluation metrics: Hits @ 1, 3, and 10
- Incorporating more historical knowledge enhances the prediction performance, but is inefficient in terms of memory and time

Experiments: object prediction, generative

Table 6: Object prediction results as a generative task (for the first 5000 test samples)

ICEWS dataset	**Afghanistan**			**India**			**Russia**		
Metric: ROUGE−	1	2	L	1	2	L	1	2	L
Historical sequence length = 3 (back to 3 days), fine-tuned RoBERTa-base as an encoder-only LLM									
SeCoGD, LDA, 5	0.4137	0.1676	0.4134	0.4916	0.2957	0.4918	0.3326	0.1189	0.3331
LEAF-OP (ours)	0.5320	0.2848	0.5322	0.5709	0.3888	0.5709	0.4671	0.2455	0.4674
Historical sequence length = 7 (back to 7 days), fine-tuned RoBERTa-base as an encoder-only LLM									
SeCoGD, LDA, 5	0.4222	0.1770	0.4221	0.5016	0.2977	0.5022	0.3385	0.1175	0.3385
LEAF-OP (ours)	0.5732	0.3502	0.5728	0.6167	0.4251	0.6168	0.5024	0.2554	0.5015
Prompt engineering: back to 5 samples or no sample, fine-tuned FLAN-T5-base as a generative LLM									
Original prompt (ours)	**0.8789**	**0.5887**	**0.8786**	**0.8666**	**0.7191**	**0.8670**	**0.8154**	**0.4942**	**0.8150**
Zero-shot prompt (ours)	0.8742	0.5805	0.8741	0.8622	0.7128	0.8624	0.8053	0.4874	0.8047
No-text prompt (ours)	0.3940	0.1307	0.3941	0.4810	0.2572	0.4817	0.3522	0.1151	0.3525
Prompt engineering: back to 5 samples, **closed-source GPT-3.5-Turbo-Instruct** as a generative LLM									
Original prompt	0.4097	0.1302	0.4092	0.3644	0.1601	0.3640	0.3480	0.1255	0.3485

- Evaluation metrics: ROUGE scores
- Fine-tuned FLAN-T5 under the question-answering format has the best performance
- Text summaries are important for LLMs to make correct predictions

Experiments: multi-event forecasting

Table 9: Model architectures for multi-event forecasting

Model	GNN	RNN	How to handle text?
DNN	N/A	N/A	N/A
DynGCN	DynGCN	N/A	Word graph only
T-GCN	GCN	GRU	Word graph only
RENET	RGCN	RNN	Event graph only
Glean	CompGCN	GRU	Word graph + event graph
LEAF-MEF (ours)	N/A	N/A	TKG quintuple prompt encoding

Table 10: Multi-event forecasting results, historical sequence length = 7 (back to 7 days)

ICEWS dataset	Afghanistan			India			Russia		
Metric	F1	Recall	Precision	F1	Recall	Precision	F1	Recall	Precision
DNN	55.77	68.14	47.20	52.49	56.38	49.10	53.81	62.61	47.18
Dynamic GCN	50.05	57.75	44.16	41.80	43.19	40.50	52.81	60.14	47.07
Temporal GCN	60.04	76.93	49.23	60.73	67.20	55.40	56.36	67.66	48.29
RENET	60.58	77.75	49.62	58.44	64.18	53.64	55.85	65.66	48.59
Glean	62.48	82.84	**50.15**	66.69	77.31	58.64	58.92	73.57	49.14
LEAF-MEF w/o SA (ours)	60.93	78.32	49.86	59.21	64.76	54.54	56.67	68.38	48.38
LEAF-MEF w/ SA (ours)	**63.63**	**88.69**	49.61	**70.99**	**87.31**	**59.81**	**62.80**	**86.81**	**49.19**

- Evaluation metrics: F1 score, recall, and precision
- The self-attention mechanism before historical aggregation is important for enhancing the forecasting performance

Limitations and potential future work

- An appropriate alignment between the pre-trained LLMs and the dynamic evolution of knowledge graphs as time goes remains to be explored.

- The LLM fine-tuning in LEAF is conducted separately, therefore, to better utilize LLM’s contextualized potential, better customization for combining LLMs with downstream tasks is expected.

- When retrieving historical knowledge, LEAF considers each quintuple wholly, whereas a closer look at the versatile relationships among different subjects, relations, and objects could lead to finer granularity.

Challenges and considerations

LLMs provide powerful tools for human event forecasting by extracting insights and performing reasoning from textual data.

Limitations

- The answers from LLMs may **lack factuality and causality**, as they generate responses based on statistical patterns rather than explicit knowledge.

How to ensure the accuracy and reliability of their outputs?

- Develop methods to enhance reliability
- Critically evaluate LLM outputs in event forecasting applications

LLMs for event data construction

Event encoding systems

Event data can be categorized into:

- **Human-encoded** events depend on human research teams with specific knowledge of the local context.
  - Armed Conflict Location & Event Data (ACLED)

- **Machine-encoded** events rely entirely on automated event encoding systems.
  - Integrated Crisis Early Warning System (ICEWS)
  - Global Database of Events, Language, and Tone (GDELT)

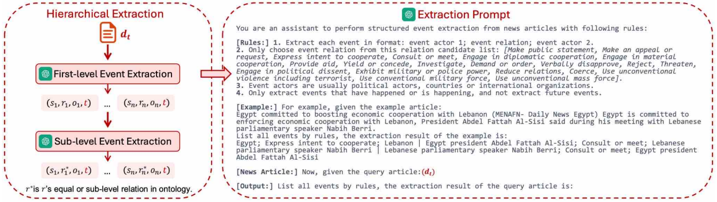
More data can be found: Deng, Songgaojun, at al. **Advances in Human Event Modeling: From Graph Neural Networks to Language Models**. KDD (2024).

LLMs for event data construction

Compared to rule-based machine coding methods, LLMs have **better language comprehension** and can effectively **save labor** compared to manual coding.

A recent work introduces a fully automated pipeline and a new event dataset MidEast-TE [Ma et al. 2024]

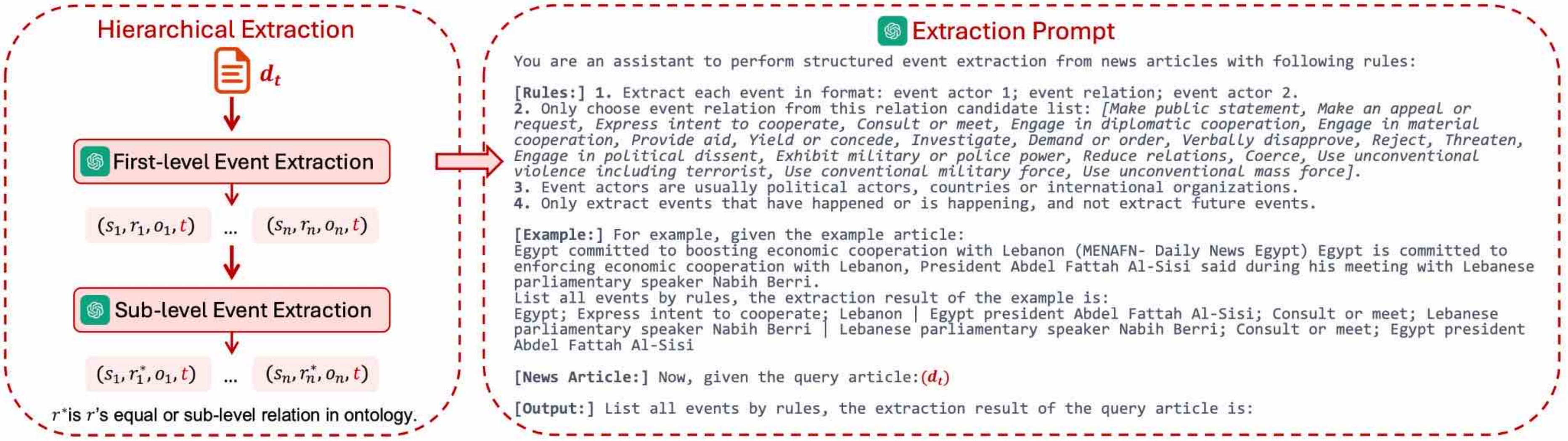
- **utilizes the pre-trained large language models (LLMs) for event extraction** from news articles.



Ma, Yunshan, et al. **SCTc-TE: A Comprehensive Formulation and Benchmark for Temporal Event Forecasting**. *arXiv* (2024).

Hierarchically extracts structured events with LLMs

- Hierarchically extraction due to **input length limitation** of LLMs.
- Follow the **three-level relation hierarchy in CAMEO** (event encoding scheme) from coarse-grain to fine-grain.



Ma, Yunshan, et al. **SCTc-TE: A Comprehensive Formulation and Benchmark for Temporal Event Forecasting**. *arXiv* (2024).

Entity linking with LLMs

No predefined entity set during LLM based event extraction, thus multiple entities may correspond to the same one, e.g., U.S.A and United States.

Linking method

- Apply a K-means clustering to group original entities into multiple groups.
- Ask the GPT-4 to perform entity linking for each cluster.
    - Entity cluster size is relatively small, thus the cost of using GPT-4 is marginal.

Accuracy might not be guaranteed

Ma, Yunshan, et al. **SCTc-TE: A Comprehensive Formulation and Benchmark for Temporal Event Forecasting**. *arXiv* (2024).

Dataset evaluation

The events in MidEast-TE are **more fine-grained**, because of

- A larger entity set.
- The distribution of events in different levels (less low level events and more high level events)

Table 4: Event extraction results comparison.

Dataset	$\lvert\mathcal{R}\rvert$	$\lvert\mathcal{E}\rvert$	#atomic events	% of different levels		
				1st	2nd	3rd
GDELT-TE	239	1,555	1,201,881	38.93	57.19	3.88
MidEast-TE	234	2,794	455,877	21.24	64.82	13.94

Evaluating LLMs for event forecasting

A benchmark for evaluating LLMs for event forecasting

MIRAI, a novel benchmark designed to systematically **evaluate LLM agents as temporal forecasters** in the context of international events.

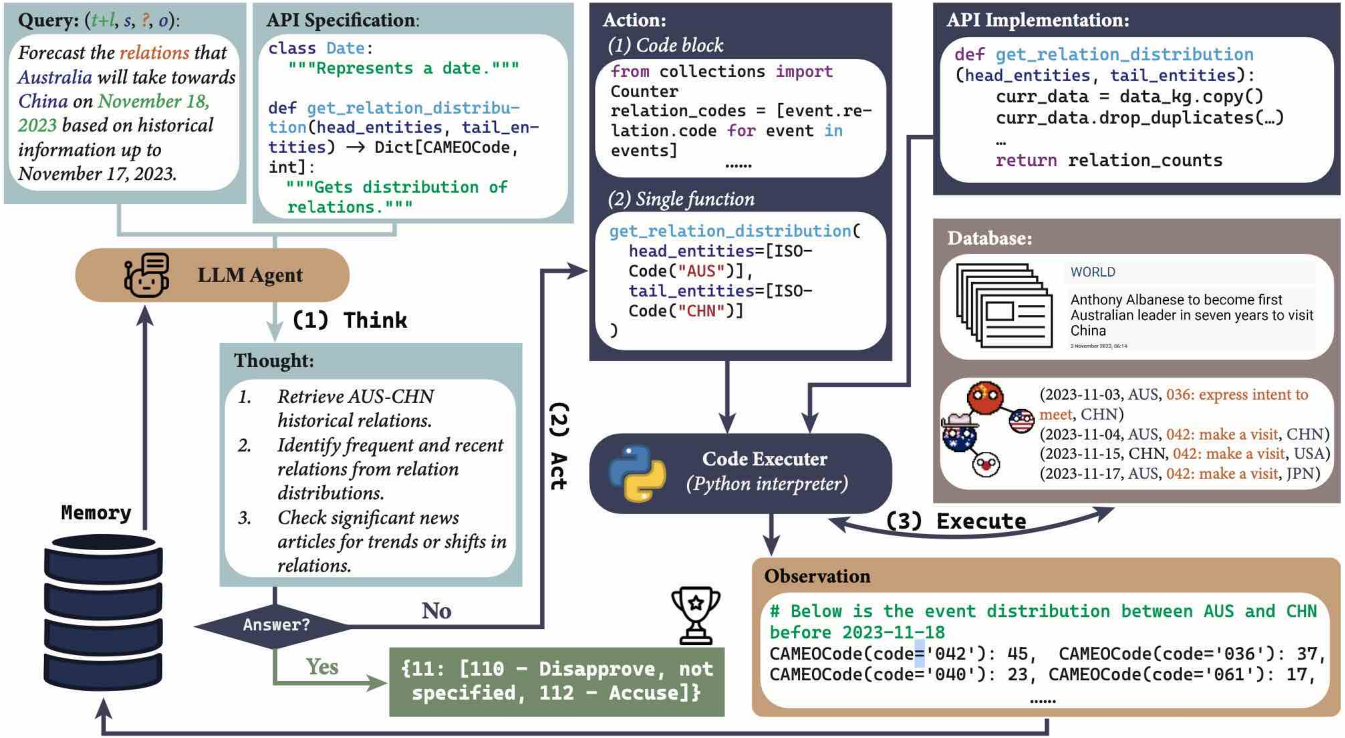
Motivation:

- Increasing interests have been put into employing LLM agents for event prediction.
- A lack of a rigorous benchmark of LLM agents' forecasting capability and reliability.

Ye, Chenchen, et al. **MIRAI: Evaluating LLM Agents for Event Forecasting**. *arXiv* (2024).

The pipeline

Overview of the LLM agent's interaction with the multi-source environment using the **ReAct** [Yao et al 2022] strategy for forecasting a query event.



Yao, Shunyu, et al. **React: Synergizing reasoning and acting in language models**. ICLR 2023.

Different agent tools and the tool-use strategies

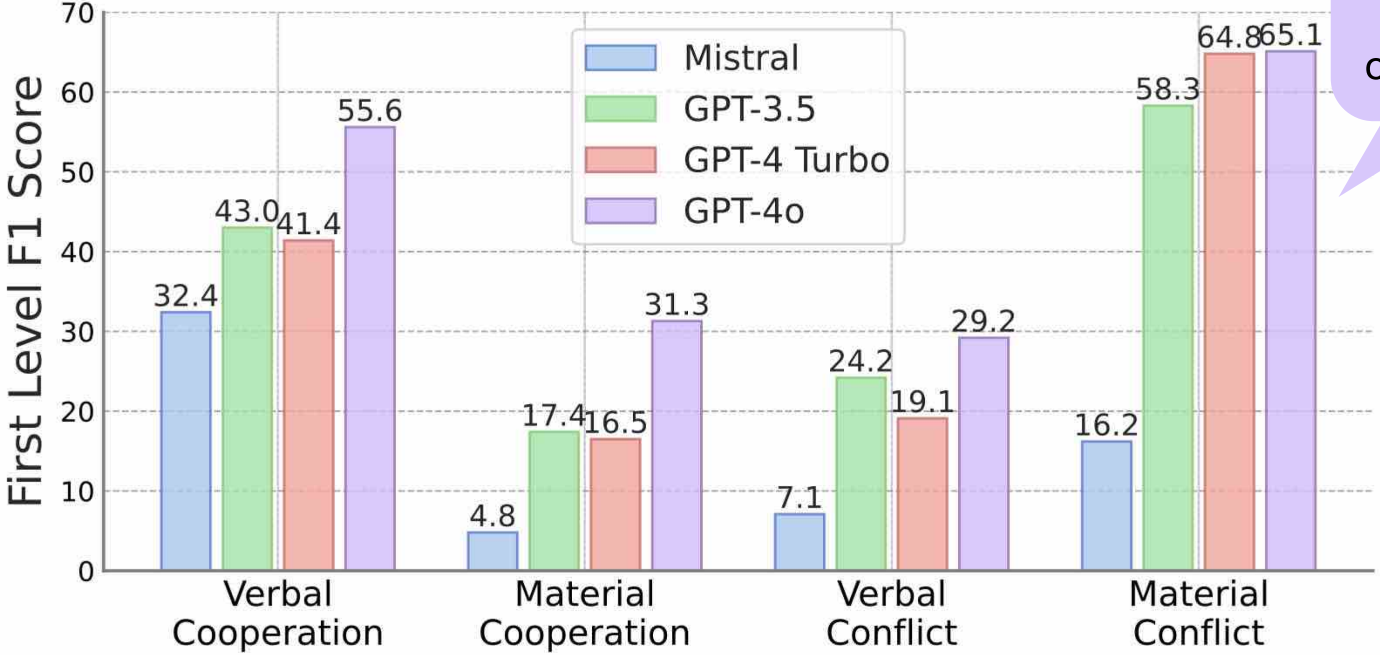
Direct IO: standard LLM chatbot approach

Zero-Shot Chain-of-Thought (ZS-CoT): prompt to the LLM to encourage explicit step-by-step thinking before making the final prediction.

ReAct: follow **ReAct** to interact with the provided environments through an iterative process of thinking, acting, and observing.

Agent	Tool-Use		Binary KL (⇓)	Quad KL (⇓)	First-level Relation (%)			Second-level Relation (%)		
	Action Type	API			Pre. (⇑)	Rec. (⇑)	F1 (⇑)	Pre. (⇑)	Rec. (⇑)	F1 (⇑)
Direct IO	—	—	$6.5_{\pm1.7}$	$15.9_{\pm1.5}$	$27.6_{\pm8.1}$	$19.7_{\pm5.9}$	$18.8_{\pm6.9}$	$6.6_{\pm1.5}$	$5.1_{\pm0.4}$	$3.5_{\pm0.8}$
ZS-COT	—	—	$6.9_{\pm0.8}$	$10.1_{\pm0.8}$	$27.6_{\pm4.0}$	$36.0_{\pm4.5}$	$26.7_{\pm4.1}$	$10.2_{\pm1.4}$	$17.4_{\pm1.1}$	$10.5_{\pm0.7}$
ReAct	Single Function	*Event-Only*	$33.5_{\pm0.7}$	$\underline{6.7}_{\pm0.7}$	$\underline{44.3}_{\pm3.9}$	$\underline{54.2}_{\pm3.9}$	$\underline{41.4}_{\pm1.7}$	$\underline{25.3}_{\pm2.6}$	$\underline{47.4}_{\pm2.4}$	$\underline{26.9}_{\pm1.9}$
	Single Function	*News-Only*	$6.1_{\pm1.0}$	$12.8_{\pm0.6}$	$27.8_{\pm3.1}$	$25.9_{\pm2.9}$	$21.8_{\pm2.3}$	$6.3_{\pm2.2}$	$9.0_{\pm2.0}$	$5.4_{\pm1.3}$
ReAct	Single Function	All	$\mathbf{3.1}_{\pm0.5}$	$\mathbf{5.9}_{\pm1.0}$	$\mathbf{47.6}_{\pm5.8}$	$\mathbf{58.3}_{\pm2.6}$	$\mathbf{44.2}_{\pm4.0}$	$\mathbf{28.7}_{\pm3.9}$	$\mathbf{51.0}_{\pm4.0}$	$\mathbf{29.6}_{\pm3.7}$
	Code Block	All	$\underline{5.1}_{\pm0.9}$	$8.9_{\pm0.5}$	$27.1_{\pm4.0}$	$38.6_{\pm2.5}$	$25.9_{\pm2.2}$	$11.6_{\pm2.4}$	$26.3_{\pm2.0}$	$12.6_{\pm1.7}$

Different LLM agents on event type prediction



GPT-4o consistently outperforms other models.

Summary and future work

Summary

- Introduction and motivation for event predictions
  - Definitions and challenges
  - A brief summary of early approaches

- Graph neural networks (GNNs)-based methods
  - Valina graph learning, graph learning with contextual information, and with causal reasoning

- Large language models (LLMs)-based methods
  - LLMs for event prediction, data construction and a benchmark

Future directions

- Combining the strengths of GNNs and LLMs
  - Utilize the structural understanding of GNNs with the contextual and linguistic capabilities of LLMs.

- Leveraging LLMs for event prediction with causality
  - Incorporate causal relationships into LLM-based predictions.

- Addressing biases and ensuring fairness
  - Identify and mitigate biases in both data and models.

…

Tutorial: Advances in Human Event Modeling: From Graph Neural Networks to Language Models

Thank you

If you have any questions, feel free to contact:
Songgaojun Deng: s.deng@uva.nl
Yue Ning: yue.ning@stevens.edu

Tutorial website

(Slides uploaded)