# Causality Enhanced Societal Event Forecasting With Heterogeneous Graph Learning

Songgaojun Deng
*Department of Computer Science*
*Stevens Institute of Technology*
Hoboken, New Jersey, USA
sdeng4@stevens.edu

Huzefa Rangwala
*Department of Computer Science*
*George Mason University*
Fairfax, Virginia, USA
rangwala@cs.gmu.edu

Yue Ning
*Department of Computer Science*
*Stevens Institute of Technology*
Hoboken, New Jersey, USA
yue.ning@stevens.edu

***Abstract*—Using observational event data to forecast societal events has been extensively studied in data-driven models. Prior work focuses on correlational analysis and ignores the importance of causal relationships behind events. Understanding the causality of events helps one infer future events by pinpointing potential triggers. In light of complex and dynamic social environments, it is difficult to comprehensively analyze the causes of societal events. In this work, we study the causal relationship between topics and events where topics are extracted from event-related documents. These topics represent probability distributions of words. We introduce a method to discover topics that have a causal effect on future events of interest. Next, we propose a causality-enhanced dynamic heterogeneous graph learning framework where topics, documents, and words are represented as nodes with changing edges. To handle the temporal dependencies of dynamic graphs, we introduce a temporal information learning module that updates node representations based on their evolving context and heterogeneous semantics. We conduct extensive experiments on four real-world datasets and demonstrate the effectiveness of our method in societal event prediction.**

***Index Terms*—Event Forecasting, Causality, Heterogeneous Graph Learning**

## I. Introduction

Societal events such as protests largely affect our daily lives. Understanding hidden social patterns and anticipating such events is important to many stakeholders, including investors, suppliers, and governments. Many efforts have emerged to achieve high accurate societal event prediction. However, previous approaches mainly focused on correlation-based studies, i.e., measuring associations between feature and target variables. Bringing causality into predictive research is promising because knowing underlying causes of events helps humans reason about future events. Involving causal information in predictive methods may also help improve forecast accuracy.

There are several attempts to incorporate causal information into predictive models for improving prediction accuracy. Some studies introduced causal objectives such as maximizing causal effects [1] in learning objectives to improve the predictive ability of recommender systems. Other work introduced pre-learned causal effects as prior knowledge to guide the model training, in disease diagnosis prediction [2], and computer vision tasks [3]. This suggests the potential benefits of leveraging causal information to enhance event prediction. Yet, few studies have investigated causal information in societal event forecasting, and the task presents many challenges:

- The causal factors that lead to societal event occurrence are very complex. In a dynamic social environment, events are generally triggered by a combination of factors, and different events are causally affected by various factors.
- Exploiting causal information about events has the potential to help predict future events, as past causal factors are very likely to trigger future events. However, how to effectively harness causal information in data-driven models is a difficult and under-explored task.
- Modeling temporal information has shown advantages in accurately predicting societal events in dynamic social environments [4], [5]. Thus, it is important to develop models that learn temporal dependencies and capture evolving context information.

To address the above challenges, we propose a causality-enhanced dynamic heterogeneous graph learning model for predicting societal events. Recently, graph representation learning has been successful in various fields. In societal event forecasting, researchers have proposed to model graph-based data, such as word graphs [5] and knowledge graphs [6], [7]. Inspired by such achievements, we propose to incorporate node causality in graph learning, i.e., to highlight nodes that are potential causal factors for the occurrence of future events of interest. During the graph learning, causal message passing enables nodes to learn richer contextual features than traditional correlation-based models. To better distinguish between causal nodes and others, we study heterogeneous graphs that include different types of nodes where some nodes preserve causal information and some do not. Compared with homogeneous graphs, heterogeneous graphs present richer relational characteristics. We develop a dynamic heterogeneous graph-based model to learn evolving information over historical timestamps. We also exploit causal information in graph learning (i.e., causal nodes) to enhance representational learning. Our contributions are summarized as follows:

- We propose to study causal factors of societal events in the form of topics. We consider a topic as a learned probability distribution of words, which contains richer semantic information than a single word or a single event. We introduce a causal inference pipeline for discovering causal topics of future events of interest based on observational event data.

- We design a dynamic heterogeneous graph learning framework where historical topics, documents, and words are represented as nodes with evolving edges. It differs from existing heterogeneous graph methods that learn node embeddings in static graphs or dynamic graphs with only evolving nodes.
- We introduce a causality-aware message passing module and a correlation-based message passing module in the proposed framework to incorporate discovered causal topics and heterogeneous semantic information.

We evaluate our method on real-world event datasets compared with several state-of-the-art models. We demonstrate the strengths of our approach in event prediction and also discuss the potential impacts and limitations of this work.

## II. Related Work

### A. Event Forecasting

Event forecasting focuses on predicting future events based on past social indicators, such as published news articles or social media data. Event forecasting has been studied in different fields such as stock markets [8], epidemics [9], crime analysis [10], and civil unrest movements [4]. The methods studied include traditional statistical models such as Hidden Markov Models [11], [12], machine learning approaches such as logistic regression [4], [13], and deep learning models [5]–[7], [14], [15]. Deep learning methods have achieved great success in predicting societal events. An attention-based spatio-temporal learning framework was proposed to model dynamic patterns of citywide abnormal events [15]. Research on graph neural networks for event prediction has also made remarkable progress. A dynamic graph based approach was introduced to forecast events and identify event-related context graphs [5]. Some researchers investigated relational graph data as a rich source of context in event prediction [6], [7]. However, few studies have explored causality in event prediction. In this paper, we attempt to include causality in graph learning which involves discovering causal information and developing a predictive model based on causal information.

### B. Heterogeneous Graph Learning

Heterogeneous graphs, which contain multiple types of nodes or multiple types of edges, have become ubiquitous in real-world scenarios, such as recommender systems and bibliographic networks. Graph neural networks (GNNs) have shown great success in processing graph-structured data for prediction tasks. GNNs learn embeddings/hidden features for each graph node by using topological information and passing messages from neighboring nodes to the target node. Some studies have attempted to extend GNNs to model heterogeneous graphs. The relational graph convolutional network (RGCN) was introduced to model knowledge graphs by learning different weight matrices for each edge type [16]. A composition-based multi-relational graph convolutional network (CompGCN) [17] was proposed to embed both nodes and relations/edges in a relational graph. Some researchers proposed a heterogeneous graph neural network, which uses random walk strategies to sample heterogeneous neighbors of each node and use node type-specific recurrent networks to integrate multimodal features [18]. The graph attention network (GAT) was extended to learn different weights for different edge types [19]. More recently, a transformer-like heterogeneous graph model, heterogeneous graph transformer (HGT), was proposed [20]. It learns node- and edge-type dependent parameters to characterize the heterogeneous attention over each edge. Such models focus on static heterogeneous graphs or dynamic heterogeneous graphs with evolving nodes. In this work, we introduce a novel heterogeneous graph learning framework that handles dynamic graphs with evolving edges. In addition, the proposed framework is able to leverage equipped causal information for graph level predictions.

## III. Preliminaries and Problem Formulation

In this section, we first formally define some important concepts. Then we present preliminary knowledge about causal inference and formulate our problem.

**Definition 1.** ***Heterogeneous Graph.*** *A heterogeneous graph is defined as a directed graph $G \subseteq (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R})$ with multiple types of nodes and edges. Particularly, each node $v \in \mathcal{V}$ is associated with a node type $\tau(v)$: $\mathcal{V} \rightarrow \mathcal{A}$ and each edge $e \in \mathcal{E}$ is associated with an edge type $\phi(e)$: $\mathcal{E} \rightarrow \mathcal{R}$. For example, we have three different types of nodes: word, document, or topic. We denote an edge as $e = (u, v)$, indicating that edge $e$ connects node $u$ and node $v$.*

**Definition 2.** ***Dynamic Heterogeneous Context Graph.*** *A heterogeneous context graph is a type of heterogeneous graph where edges have timestamps. A timestamped edge $(e, t) = ((u, v), t)$ denotes the connection of two nodes $u$ and $v$ at time $t$. A heterogeneous context graph consists of all timestamped edges with the same timestamp. We use $G[t] \subseteq (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R}, \mathcal{T})$ to denote the heterogeneous context graph at $t \in \mathcal{T}$. A dynamic heterogeneous context graph is built upon a sequence of heterogeneous context graphs in ascending time order representing dynamic contexts over time, e.g., $\{G[1], ..., G[T]\}$. Note that a dynamic heterogeneous context graph can have static edges (and nodes), and such edges appear at all timestamps.*

**Definition 3.** ***Event Forecasting.*** *Given historical data $\mathcal{X}$ as input features (e.g., news articles), learn a classifier $f(\mathcal{X}) \rightarrow \mathcal{Y}$ that maps the input to a binary target variable $\mathcal{Y} \in \{0, 1\}$. The target variable indicates whether a type of event (e.g., protest) will occur at a target location at a future time.*

We construct a dynamic heterogeneous context graph from historical news articles as input features to address the above event forecasting problem. Given a city and a historical window, we collect news articles reported during the historical window time in this city to construct a graph. An example of this graph is shown in Fig. 2a. *We define three types of nodes: word, topic, and document.* The word nodes are keywords extracted from the documents, and the topics are obtained from a pre-trained topic model [21]. Different edge types are defined for different node type pairs. We construct timestamped edges based on the news articles reported in the given timestamp. In detail, the edge weight between two word nodes is determined by their positive pointwise mutual information (PMI) scores [5], [22], and the edge weight between a document node and a word node is defined by TF-IDF [22]. The edge weight between a topic and a word node denotes the probability of the word relating to that topic. A similar edge weight relation applies to topic and document nodes. The edge weight between two topic nodes denotes the cosine similarity of these two topics, which are static and do not change over time. Given the different semantic contexts of each timestamp, the edges connecting words to words and/or topics show recurring patterns. For example, a student rally for educational reform occurs on day $t$ and happens again on day $t + 5$. Note that we assume each news article is unique (i.e., the same article does not appear for multiple days), so the edges between words and documents and the edges between topics and documents are unique over time.

Next, we present some preliminaries on causal inference and formulate our problem based on the aforementioned concepts.

**Definition 4.** ***Treatment Effect.*** *Treatment effect refers to the causal effect of a given treatment or intervention (e.g., the administering of a drug) on an outcome variable of interest (e.g., the health of the patient). In the Rubin causal model [23], the treatment effect measures the difference in outcomes between a unit assigned to the treatment group and a unit assigned to the control group. The average treatment effect (ATE) measures the difference in the mean (average) outcome between the treatment and control groups.*

**Definition 5.** ***Propensity Score Matching (PSM).*** *PSM is a statistical matching technique that estimates the ATE of a treatment by accounting for the covariates/confounding variables (i.e., variables other than treatment that may affect outcome) that predict receiving the treatment [24]. PSM attempts to reduce the bias due to confounding variables that could be found in an estimate of the treatment effect.*

**Problem Statement** Based on the aforementioned definitions, we aim to build an event predictor $F : \mathcal{X} \rightarrow \mathcal{Y}$ that takes a dynamic heterogeneous context graph $\{G[t-k+1], ..., G[t]\} \in \mathcal{X}$ as input and estimates the occurrence probabilities of a type of events (e.g., protest) in the future $Y^{t+1:t+1+\Delta} \in \{0, 1\} \in \mathcal{Y}$ for a given location. A causal node set $\pi$ (i.e., causal topics) that has causal effects on the occurrence of future events is discovered and identified in the input graph. $k \geq 1$ is the size of the historical window, indicating that the dynamic heterogeneous context graph is built based on data during a $k$-size time frame. The prediction window $(t + 1 : t + 1 + \Delta)$ where $\Delta \geq 1$ indicates the time window that model forecasts the occurrence of events.

## IV. Methodology

In this section, we present the technical details of our proposed method. There are two major components in this method: causal analysis of topics and the incorporation of causal information into a dynamic heterogeneous graph model for event prediction. Next, we elaborate on these two parts.

### A. Causal Analysis

In this paper, we examine topics that have the potential to trigger or hinder some future events (e.g., protests) in a given location. We refer to these topics as causal topics. Answering this question can help people understand cause-and-effect relationships behind societal events. Ideal causal identification is difficult to achieve because counterfactual outcomes are not observable in the real world. In this study, we employ techniques from causal inference literature that can bring us closer to interpretable event analysis.

*1)* ***Discovering Causal Topics****:* We begin with a corpus of news articles, including texts and timestamps reported in different cities. We first obtain a collection of timestamps for each city with sufficient news coverage during these times. We create time windows for each timestamp and each city, considering $w$ days (historical window) before each timestamp and $m$ days after each timestamp (prediction window).

**Treatment assignment.** We then group these windows into treatment and control groups based on whether the news article collection at the given timestamp and city includes a target topic $j$ or not, denoted as $Z^j \in \{0, 1\}$. The target topic is defined based on a pre-trained topic model given the corpus. We denote the total number of topics as $J$. We define covariates/confounders $X$ as the frequency of unigrams in the historical window. The outcome as a binary value $Y \in \{0, 1\}$ indicates whether an event of interest (e.g., protest) occurs in the prediction window or not.

**Propensity Score Matching.** We use Propensity Score Matching to obtain treated-controlled instance pairs. A propensity score is the probability of an instance being assigned to a particular treatment given observed covariates, i.e., $P(Z^j = 1|X)$. In the implementation, we learn a propensity score estimator using logistic regression. We match each treated instance with a controlled instance of similar propensity score values. The matching operation is non-replacement, i.e., no controlled instance is used more than once. The matching procedure produces a treatment group and a control group of the same size. We estimate the ATE based on the outcomes of each matched instance pair. Formally, the ATE can be written as $\mathbb{E}[Y_{Z^j=1} - Y_{Z^j=0}]$, where $Y_{Z^j=1}, Y_{Z^j=0}$ are the outcomes of a pair of treated and controlled instances, respectively.

**Statistical test.** Given the ATEs for all topics, we select the causes by measuring their significance using two-tailed z-score tests with a significance level of 99%. If the z-score of a topic falls into the right tail ($> 0$), we consider it has a positive causal effect. If it is in the left tail ($< 0$), we consider it has a negative causal effect. We illustrate the causal topic discovery procedure in Fig. 1.

*2)* ***Discovering Evolving and Multi-view Causal Topics****:* Motivated by ever-changing social environments, we propose to explore evolving causal topics that can reflect time-sensitive causes. Following the procedures introduced in the previous

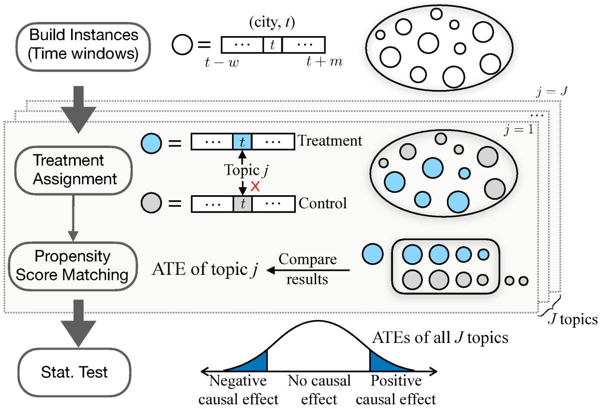


Fig. 1: Illustration of the causal topic discovery process. Each circle represents an instance. Each instance refers to a window of time in a city. Blue circles indicate treated instances and gray circles represent controlled instances. Topic $j$ appears in the news data at $t$-th timestamp in the treated instance but not in the control instance.

section, we update the causal topics for each season (i.e., 3 months) in chronological order using time windows in the past. In addition, topics can have long-term or short-term causal effects. Based on this intuition, we vary the size of the prediction window $m$ (e.g., 3, 7, and 14) to obtain topics that have causal effects on protests in different lengths of the future. The size of a historical window $w$ is set to 14 days. We use $\pi$ to represent the discovered multi-view causal topics.

### B. Event Prediction Model

We propose a dynamic **h**eterogeneous **g**raph model with **c**ausality enhanced node representations, **HGC**, to forecast societal events. Inspired by a self-attention-based model Heterogeneous Graph Transformer (HGT) [20], our model is developed to capture the temporal information underlying dynamic heterogeneous context graphs with evolving edges. Meanwhile, it can effectively model causal information to improve model prediction. Figure 2b shows the overall architecture of the proposed model. The goal of the proposed model is to apply message passing on the dynamic heterogeneous context graph and reason about the occurrence of future events of interest. The model can be decomposed into two main parts: (1) **Causality Enhanced Dynamic Message Passing** which computes a causally and temporally contextualized representation for each node by aggregating messages from heterogeneous source nodes; (2) **Overall Aggregation** which generates a global embedding for final prediction.

*1) **The HGT framework**:* The HGT model is an attention-based framework that learns node representations in heterogeneous graphs for node-level prediction and link prediction tasks. The framework can be written as follows:

$$\mathbf{h}_v^l \leftarrow \underset{\forall u \in N(v), \forall e \in E(u,v)}{\mathbf{AGG}} \big(\mathbf{ATT}(u,e,v) \cdot \mathbf{MSG}(u,e,v)\big), \quad (1)$$

where $N(v)$ denotes all the source nodes of node $v$ and $E(u,v)$ denotes all the edges from node $u$ to $v$. There are three basic operators in the framework: **ATT** denotes the attention module, which calculates the importance of each source node; **MSG** means the message module, which obtains the message for the source node; and **AGG** is an aggregation module, which aggregates the neighborhood attention message of source nodes via some operators, e.g., mean or sum.

The HGT model handles dynamic heterogeneous graphs, provided that the graph edges are fixed and the graph nodes have different timestamps. The approach incorporates a learnable relative time embedding into the source node embedding, where the relative time is the time gap between the source and target nodes. *However, the method cannot handle edges that change over time, which is a critical feature of event-related context graphs.* For example, the connection of two major entities recurring within a historical period can provide important clues for future events.

*2) **Causality Enhanced Dynamic Message Passing**:* To better capture the temporal dependencies in dynamic heterogeneous context graphs, we introduce a framework to model the temporal information of each graph node in the context of evolving edges. Furthermore, we propose causality-aware message passing, which exploits the predetermined causal node information from the causal analysis part. We first design a temporal information learning module (**TEM**) that incorporates embeddings of nodes learned in the past graph:

$$\mathbf{h}_v^l[t] \leftarrow \mathbf{TEM}(\tilde{\mathbf{h}}_v^l[t],\ \mathbf{h}_v^l[<t]), \quad (2)$$

where $\mathbf{h}_v^l[<t] \in \mathbb{R}^d$ is the past states of node $v$, and $\tilde{\mathbf{h}}_v^l[t] \in \mathbb{R}^d$ is learned from a combination module (**COM**):

$$\tilde{\mathbf{h}}_v^l[t] \leftarrow \mathbf{COM}(\mathbf{o}_v^l[t],\ \mathbf{c}_v^l[t]). \quad (3)$$

The **COM** module combines two types of context information obtained from graph message passing.

- *Correlation-based message passing*:

$$\mathbf{o}_v^l[t] \leftarrow \underset{\forall u \in N^t(v), \forall e \in E^t(u,v)}{\mathbf{AGG}} \big(\mathbf{ATT}(u,e,v) \cdot \mathbf{MSG}(u,e)\big), \quad (4)$$

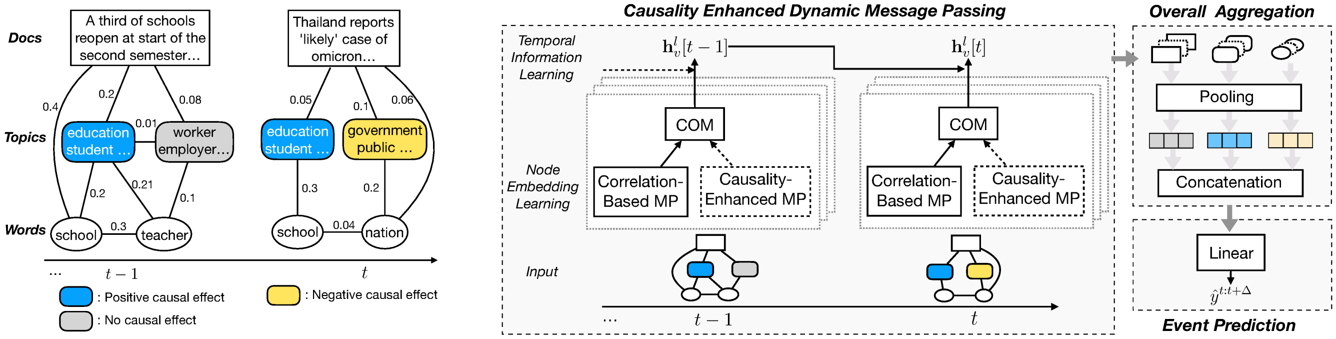
where $N^t(v)$ denotes all the source nodes of node $v$ at the $t$-th heterogeneous context graph. $E^t(u,v)$ denotes all the edges from node $u$ to $v$ at the $t$-th heterogeneous context graph $G[t]$. The **ATT**, **MSG**, and **AGG** have similar functions as in the HGT framework.

- *Causality-aware message passing*:

$$\mathbf{c}_v^l[t] \leftarrow \underset{\forall u \in N^t(v), \forall e \in E^t(u,v)}{\mathbf{AGG}} \big(\mathbf{C\text{-}ATT}(u,v) \cdot \mathbf{C\text{-}MSG}(u,e)\big). \quad (5)$$

The **C-ATT** estimates another importance score of each source node given its causal identifier (i.e., causal or not). The **C-MSG** computes another message vector for each source node.

*a) **Correlation-based Message Passing**:* In the heterogeneous context graph at time $t$, given a target node $v$, its neighbors can be of different types, leading to different edge types. We map target node $v$ into a Query vector, and source node $u$ into a Key vector, and calculate their scaled dot product to obtain the attention score. In our dynamic heterogeneous context graph, the edge weight between two nodes is defined by their associations, e.g., TF-IDF score between a word and a document node or PMI score between two word nodes. This information can be important when passing information from the source node to the target node, especially in dynamic graphs with evolving edges. For example, if a word-word



(a) Dynamic heterogeneous context graph.

(b) The overall architecture of the proposed framework.

Fig. 2: (a) An example of the dynamic heterogeneous context graph with three types of nodes: word, topic, and document. Topic nodes have either positive, negative, or no causal effects. (b) The overall architecture of the proposed event prediction framework. The proposed framework takes dynamic heterogeneous context graphs as input and learns node embeddings enhanced by causal topics. Then, we aggregate the information of each type of node for event prediction via an output layer.

edge has a higher PMI score at time $t-1$ and a lower PMI score at time $t$, the learned message should be different at the two timestamps. Thus, we propose a novel method to calculate attention scores in weighted heterogeneous graphs. The attention scores are achieved by combining content-based and prior-based associations. The content-based association is evaluated based on the embedding of the source and target nodes. The prior-based association depends on the edge attributes and is calculated using the edge weight and an edge type-specific weight vector. We adopt multi-head attention to capture multiple hidden relationships for an edge. The $h$-head attention for an edge $(e,t)=((u,v),t)$ can be written as:

$$\mathbf{ATT}(u,e,v) = \text{Softmax}\Big(\overset{h}{\underset{i=1}{\|}} \text{head}^i_{\text{ATT}}(u,e,v)\Big) \tag{6}$$

$$\begin{aligned}\text{head}^i_{\text{ATT}}(u,e,v) &= \Big(\big(\mathbf{W}^i_{\text{Q}}\mathbf{h}^{l-1}_v + \omega_e\mathbf{w}_{\phi(e)}\big)^\top(\mathbf{W}^i_{\text{K}}\mathbf{h}^{l-1}_u)\Big)/\sqrt{d}\\ &= \Big(\underbrace{(\mathbf{W}^i_{\text{Q}}\mathbf{h}^{l-1}_v)^\top(\mathbf{W}^i_{\text{K}}\mathbf{h}^{l-1}_u)}_{\text{content-based}}\\ &\quad+\underbrace{(\omega_e\mathbf{w}_{\phi(e)})^\top(\mathbf{W}^i_{\text{K}}\mathbf{h}^{l-1}_u)}_{\text{prior-based}}\Big)/\sqrt{d}\end{aligned} \tag{7}$$

where $^\top$ is transposition and $\|$ means concatenation. $\mathbf{h}^{l-1}_u, \mathbf{h}^{l-1}_v \in \mathbb{R}^d$ are the embeddings of node $u$ and $v$ from the $l-1$-th layer, respectively. The initial embedding of a node (e.g. $\mathbf{h}^0_u$) is a pre-trained word embedding for a word node and a randomly initialized embedding for topic and document nodes. We define a generalized embedding for all document nodes, since the number of documents can be infinite. $\omega_e \in \mathbb{R}$ is the weight of the edge $e$ and $\mathbf{w}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h}}$ is the distinct weight vector for the edge type $\phi(e)$. $\mathbf{W}^i_K, \mathbf{W}^i_Q \in \mathbb{R}^{\frac{d}{h}\times d}$ are weight matrices for the $i$-th head that project the embedding of the source node $u$ and target node $v$ into the $i$-th Key and Query vector, respectively. $d$ denotes the feature size. Unlike the HGT framework, we ignore the unique edge-based matrix used to handle possible different edge types. In a heterogeneous context graph, the type of edge is defined by the type of the two end nodes and is unique. For example, there is only one type of edge between a word node and a topic node. The edge weight between a topic node and a word node indicates the probability of this word being associated with this topic. Eq. 7 captures different semantic relationships of different node pairs. Then, we concatenate $h$ attention heads together to get the attention vector for each node pair. We apply a Softmax to get an attention score for each head. For each node pair $(u,v)$, the attention scores form a $h$-dimensional vector, i.e., $\mathbf{ATT}(u,v) \in \mathbb{R}^h$.

To calculate the message, we first map the source node $v$ into a Value vector and then apply a linear transformation on the Value vector. The multi-head message for each edge with $h$ heads is as follows:

$$\mathbf{MSG}(u,e) = \overset{h}{\underset{i=1}{\|}}\Big(\mathbf{W}^i_{\text{V}}\mathbf{h}^{l-1}_u\mathbf{W}^{\text{MSG}}_{\phi(e)}\Big), \tag{8}$$

where $\mathbf{W}^i_{\text{V}} \in \mathbb{R}^{\frac{d}{h}\times d}$ is the weight matrix that project the embedding of the source node $u$ into the $i$-th Value vector. The matrix $\mathbf{W}^{\text{MSG}}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h}\times\frac{d}{h}}$ is edge type specific to incorporate edge dependencies.

Then, in the **AGG** function, we use a mean operator to aggregate the information from all neighbors:

$$\mathbf{o}^l_v[t] = \underset{\forall u\in N^t(v),\forall e\in E^t(u,v)}{\text{Mean}}\big(\mathbf{ATT}(u,e,v)\cdot\mathbf{MSG}(u,e)\big), \tag{9}$$

where the $[t]$ denotes the node embedding obtained by learning the $t$-th heterogeneous context graph.

*b) **Causality-aware Message Passing:*** We discover causal topics from observational data and propose to incorporate such causal topics in heterogeneous graph learning. Specifically, the causal information of topic nodes is propagated over the graph through message passing. We also use a self-attention approach to learn context-based causal information, since causal topics generally have different effects in different contexts. Suppose an edge $(e,t) = ((u,v),t)$ whose source node $u$ is a topic. We use $\varsigma(u) \in \{1,-1,0\}$ to denote the causal identifier of the source topic node $u$, i.e., positive causal effect ($\varsigma(u)=1$), negative causal effect ($\varsigma(u)=-1$), or no causal effect ($\varsigma(u)=0$). The causal identifier of the topic node is determined by the multi-view causal topics obtained in Sec. IV-A2. Here, we define a weight matrix $\Gamma^i_{\varsigma(u)} \in \mathbb{R}^{\frac{d}{h}\times\frac{d}{h}}$ for learning causal information given the causal identifier of node $u$. The multi-head attention in the causality-aware message passing is formally defined as below:

$$\textbf{C-ATT}(u,v) = \text{Softmax}\Big(\mathop{\Big\|}_{i=1}^{h} \text{head}^i_{\text{C-ATT}}(u,v)\Big) \quad (10)$$

$$\text{head}^i_{\text{C-ATT}}(u,v) = \frac{(\mathbf{W}^i_{\text{C-Q}}\mathbf{h}^{l-1}_v)^\top \Gamma^i_{\varsigma(u)}(\mathbf{W}^i_{\text{C-K}}\mathbf{h}^{l-1}_u)}{\sqrt{d}} \quad (11)$$

where $\mathbf{W}^i_{\text{C-K}}, \mathbf{W}^i_{\text{C-Q}} \in \mathbb{R}^{\frac{d}{h}\times d}$ are weight matrices for the $i$-th attention head. The use of the weight matrix $\Gamma^i_{\varsigma(u)}$ aims to propagate different causal information between nodes.

With the obtained attention, we propose a time-aware message that incorporates a time embedding into the source node embedding. It is motivated by the fact that causal topics might have long-term or short-term effects, and including time information can better capture time-aware causal information in dynamic graphs. Inspired by the positional encoding in Transformer [25], we define a fixed time embedding $TE[t] \in \mathbb{R}^d$as follows:

$$TE[t]_{(2s)} = sin\big(pos(t)/10000^{2s/d}\big) \quad (12)$$

$$TE[t]_{(2s+1)} = cos\big(pos(t)/10000^{2s+1/d}\big) \quad (13)$$

where $s$ is the dimension index and $d$ is the embedding size. Each dimension of the time embedding corresponds to a sinusoid. We use $pos(\cdot)$ to denote the position of the timestamp $t$ in the historical window, e.g., $pos(t) = k$ given all historical timestamps $(t-k+1,...,t)$. With the time embedding, we formalize the calculation of the message as follows:

$$\textbf{C-MSG}(u,e) = \mathop{\Big\|}_{i=1}^{h} \Big((\mathbf{W}^i_{\text{C-V}}\mathbf{h}^{l-1}_u + TE[t])\mathbf{W}^{\text{C-MSG}}_{\phi(e)}\Big) \quad (14)$$

where $\mathbf{W}^i_V \in \mathbb{R}^{\frac{d}{h}\times d}$ is the weight matrix for projecting the embedding of the source node $u$ into the $i$-th Value vector for the causality-aware message passing. $\mathbf{W}^{\text{C-MSG}}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h}\times\frac{d}{h}}$ is the edge type based learnable matrix. In the **AGG** function, we aggregate the information from all neighbors using the mean operator:

$$\mathbf{c}^l_v[t] = \mathop{\text{Mean}}_{\forall u\in N^t(v), \forall e\in E^t(u,v)} \big(\textbf{C-ATT}(u,v)\cdot\textbf{C-MSG}(u,e)\big). \quad (15)$$

For the combination module, we use a simple summation of the two types of messages followed by a ReLU function:

$$\tilde{\mathbf{h}}^l_v[t] = \text{ReLU}(\mathbf{o}^l_v[t] + \mathbf{c}^l_v[t]). \quad (16)$$

*c) **Temporal Information Learning**:* Based on the message passing modules introduced above, we obtain the learned graph embedding of each node in the $t$-th heterogeneous context graph. In the dynamic heterogeneous context graph, the same node may appear in multiple heterogeneous context graphs with different neighboring nodes. For example, an entity name (i.e., a word node) may be mentioned in news on several historical days. Such information may be critical in context and helpful for accurate event prediction. Thus, we propose to involve a historical node embedding when updating the embedding of nodes for the current time. Formally, we introduce a node-type specific parameter that weights the node embedding at time $t$ and its past state $\mathbf{h}^l_v[< t]$. We define the past state of a node $v$ as the embedding of the node obtained before the current timestamp $t$. For instance, if node $v$ appears in the $t-3$-th and $t$-th graph, the past state for time $t$ is the node embedding learned form the $t-3$-th graph, i.e., $\mathbf{h}^l_v[< t] = \mathbf{h}^l_v[t-3]$. For nodes without past states, a zero vector is defined, i.e., $\mathbf{h}^l_v[< t] = \mathbf{0}$. The formal calculation can be written as follows:

$$\mathbf{h}^l_v[t] = \alpha_{\tau(v)}\cdot\tilde{\mathbf{h}}^l_v[t] + (1-\alpha_{\tau(v)})\cdot\mathbf{h}^l_v[< t] \quad (17)$$

where $\tilde{\mathbf{h}}^l_v[t]$ is the node feature obtained from the message passing modules (Eq. 16). $\alpha_{\tau(v)} \in \mathbb{R}$ is the learnable parameter for node type $\tau(v)$.

*3) Overall Aggregation and Event Prediction:* We include an averaging pooling layer on the latest node embeddings for all types of nodes. For this purpose, we obtain the global embedding of each type of node. We further concatenate these global embeddings and feed them into a linear output layer for event prediction. The computation can be written as:

$$\hat{y}^{t:t+\Delta} = \sigma\Bigg(\Big(\mathop{\|}_{a\in\mathcal{A}} \mathop{\text{Mean}}_{v\in\mathcal{V},\tau(v)=a}\big(\mathbf{h}^l_v[t]\big)\Big)\mathbf{w}_O + b_O\Bigg), \quad (18)$$

where $a \in \mathcal{A}$ denotes a node type, $\mathbf{w}_O \in \mathbb{R}^{d*|\mathcal{A}|\times d}, b_O \in \mathbb{R}$ are parameters of the output layer, and $\sigma$ is the sigmoid function. We minimize the binary cross entropy loss to optimize the model parameters.

## V. Experimental Evaluation

To evaluate our model for societal event forecasting, we aim to answer the following research questions: **RQ1**: How well does our causality enhanced model predict future events compared to other approaches? **RQ2**: How do causal topics affect the performance of event prediction in our proposed model? **RQ3**: How sensitive is our model to hyperparameters?

We adopt the F1 score (F1) and the balanced accuracy (BACC) to evaluate the prediction performance.

### A. Datasets

The experiments are conducted on four event datasets collected from Integrated Conflict Early Warning System (ICEWS) [26]. These events are encoded into 20 main categories (e.g., protest, demand, appeal) using Conflict and Mediation Event Observations (CAMEO) event codes. Each event has attributes such as geolocation, date, category, etc. In this work, we focus on predicting one category of events with significant social impact: *Protest*. The event prediction task essentially becomes a binary classification problem. We built event datasets for four countries, including Thailand (THA), Afghanistan (AFG), Egypt (EGY), and Russia (RUS), covering the period from 2014 to 2017. For each country, we collect historical news articles and protest events for different timestamps and city pairs to create training and testing samples. We ignore samples with limited context information (e.g., fewer than 7 news articles reported for a city in the historical window) and samples with limited temporal information (e.g., less than 3 days of news articles reported). Table I lists the main statistics for the four datasets. Note that we choose the number of topics based on the coherence of each topic.

### B. Comparison Methods

We compare our approach with several state-of-the-art baselines that learn homogeneous, heterogeneous, static and dynamic graphs.

TABLE I: Dataset statistics. %Positive indicates the rate of positive samples when the prediction window is 5. #News indicates the average number of historical news articles for each sample, along with the standard deviation.

| Dataset | #Samples | %Positive | #Cities | #News | #Topics ($J$) |
|---|---|---|---|---|---|
| THA | 1,151 | 44.22% | 23 | 40±35 | 50 |
| AFG | 1,318 | 38.01% | 18 | 48±33 | 60 |
| EGY | 1,371 | 59.96% | 24 | 61±43 | 60 |
| RUS | 2,323 | 40.81% | 26 | 134±127 | 60 |

- GAT [19], which applies multi-head attention on neighbors' embeddings. We build a static homogeneous graph containing only word nodes and use it as input.
- EvolveGCN [27], which adapts the GCN [28] model along the temporal dimension. It captures the dynamism of the graph sequence through using an RNN to evolve the GCN parameters. The input is a sequence of dynamic homogeneous graphs containing only word nodes.
- RGCN [16], which learns a unique weight matrix for each edge type. We use static heterogeneous graphs as input.
- HGT [20], which includes node- and edge-type dependent parameters to characterize the heterogeneous attention over each edge. It introduces the relative temporal encoding technique to handle dynamic heterogeneous graphs in which the timestamp of nodes can be different. It takes dynamic heterogeneous context graphs as input. Considering the ever-changing edges in our data, we adapt this model by applying temporal encoding to edges.

To analyze the effectiveness of our model components, we test two variant models: **HGC**$_{-\text{prior}}$ removes the prior-based association in correlation-based message passing. **HGC**$_{-\text{causal}}$ eliminates the causality-aware message passing module.

### C. Implementation Details

*1) Graph Data Construction:* To construct a sample, i.e., a dynamic heterogeneous context graph, we set the historical window size to 7, and the prediction window size to 5. We use the Latent Dirichlet Allocation (LDA) [21] model to train a topic model for each country. We pre-train a 300-dimensional word2vec embedding [29] for each word using all text data in each country. Pre-processing of the text data is performed, including cleaning, tokenizing words, and removing stop words.

For each heterogeneous context graph, the word-word edge weight is the positive PMI score, the word-document edge weight is the TF-IDF value, the topic-topic edge weight is the cosine similarity, and the topic-document edge weight is determined by the probability of the document being related to the topic. For word-topic edges, we also use the probability that the word is related to the topic, but we only consider the top 30 words related to the topic. For topic-topic and topic-document edges, edges below a threshold of 0.2 and 0.01 have a weight of 0, respectively.

We randomly split the data samples into training, validation, and test sets at a ratio of 60%-20%-20% for each dataset. This partition is used for most experiments unless otherwise stated.

*2) Training Details:* We search for the size of the hidden states from $\{32,48,64,80\}$ in neural networks of all methods. Note that we keep the same hidden state dimension for different hidden layers. The number of graph learning layers is 2 for GAT and RGCN models, and 1 for EvolveGCN, HGT and our model. For multi-head attention-based approaches, we set the number of heads to 4. All parameters are initialized with Glorot initialization [30] and trained using the Adam [31] optimizer with learning rate 1e-3, weight decay 5e-4, and dropout rate 0.5. We set the batch size to 32 in all settings. For all methods, the best-trained model is selected by early stopping when the validation loss does not decrease for 20 consecutive epochs. All experimental results are the average of 5 randomized trials. All code is implemented using Python 3.7.9 and Pytorch 1.7.0 with CUDA 9.2. All graph neural networks are implemented using the Deep Graph Library 0.5.2 [32]. The implementation code for the HGT model is adapted from https://github.com/acbull/HGT-DGL. Code of the proposed model is available at https://github.com/yuening-lab/HGC.

## VI. Experimental Results

### A. Event Prediction Performance (RQ1)

We report the event prediction results in terms of F1 and BACC for the proposed model and the baselines on the four datasets, as shown in Table II. We conduct experiments on two data settings that vary the ratio of the training and test sets while fixing the size of the validation set to 20% of the total data. We aim to examine the predictive power of different models when using limited training data. The results show that for both metrics, the proposed model outperforms all baselines in both data settings for all datasets. The proposed model achieved relative performance improvements of 1.8%-8.6% and 1.8%-5.6% over the baselines for F1 and BACC, respectively. For homogeneous graph-based models, the dynamic model EvolveGCN outperforms GAT in most cases. When the training ratio is 40%, GAT achieves better results in both F1 and BACC on THA and AFG datasets. As can be seen from Table I, THA and AFG contain fewer samples and less news than other datasets. It suggests that when homogeneous graphs are relatively sparse or training samples are limited, dynamic homogeneous models might be less helpful in capturing important information for predicting events. For heterogeneous graph based models, the HGT model achieves better performance than the RGCN model and beats all other homogeneous graph based methods. It demonstrates the effectiveness of self-attention in learning hidden features in heterogeneous graphs.

We conduct an ablation study to analyze the effect of two parts of our framework, i.e., the causality-aware message passing module and the prior-based association term in the correlation-based message passing module. We notice that the results for both variants of our base model show some performance degradation, most notably in the THA dataset.

TABLE II: Event prediction performance comparison of different approaches.

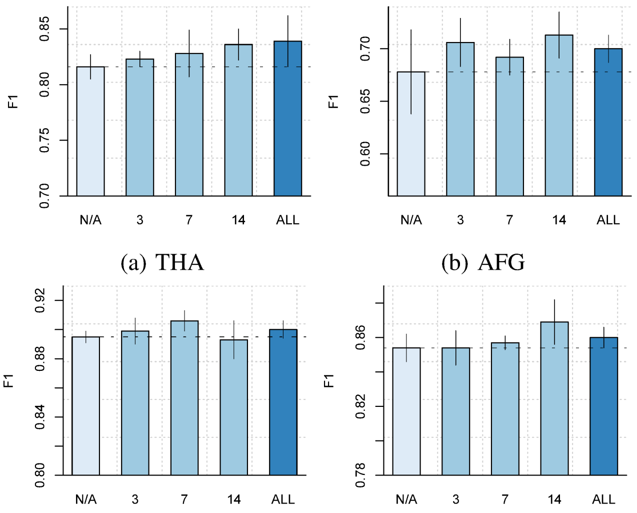
| | Training ratio | Metric | GAT | EvolveGCN | RGCN | HGT | **HGC$_{-\text{causal}}$** | **HGC$_{-\text{prior}}$** | **HGC** |
|---|---|---|---|---|---|---|---|---|---|
| THA | 60% | F1 | 0.713±0.038 | 0.717±0.019 | 0.754±0.014 | 0.803±0.03 | 0.816± 0.011 | 0.824±0.023 | **0.839±0.023** |
| | | BACC | 0.767±0.029 | 0.765±0.015 | 0.795±0.012 | 0.838±0.024 | 0.849± 0.009 | 0.854±0.019 | **0.867±0.019** |
| | 40% | F1 | 0.662±0.028 | 0.628±0.060 | 0.719±0.013 | 0.765±0.025 | 0.770±0.022 | 0.792±0.015 | **0.796±0.019** |
| | | BACC | 0.711±0.019 | 0.698±0.033 | 0.759±0.007 | 0.800±0.021 | 0.806±0.017 | 0.823±0.012 | **0.826±0.015** |
| AFG | 60% | F1 | 0.512±0.056 | 0.576±0.043 | 0.599±0.008 | 0.650±0.029 | 0.678±0.04 | 0.689±0.028 | **0.700±0.013** |
| | | BACC | 0.641±0.015 | 0.673±0.025 | 0.684±0.011 | 0.721±0.021 | 0.745±0.027 | 0.751±0.021 | **0.758±0.010** |
| | 40% | F1 | 0.544±0.071 | 0.541±0.029 | 0.610±0.023 | 0.629±0.028 | 0.670±0.025 | 0.663±0.016 | **0.683±0.011** |
| | | BACC | 0.657±0.036 | 0.635±0.012 | 0.686±0.023 | 0.711±0.019 | 0.741±0.019 | 0.735±0.012 | **0.751±0.009** |
| EGY | 60% | F1 | 0.851±0.009 | 0.871±0.012 | 0.877±0.007 | 0.882±0.007 | 0.895±0.004 | **0.899±0.006** | **0.899±0.006** |
| | | BACC | 0.773±0.016 | 0.816±0.015 | 0.842±0.010 | 0.837±0.012 | 0.855±0.003 | 0.858±0.017 | **0.859±0.006** |
| | 40% | F1 | 0.838±0.020 | 0.841±0.006 | 0.866±0.007 | 0.866±0.010 | 0.877±0.009 | 0.878±0.009 | **0.882±0.002** |
| | | BACC | 0.766±0.029 | 0.783±0.015 | 0.839±0.008 | 0.831±0.006 | 0.847±0.012 | 0.851±0.011 | **0.850±0.009** |
| RUS | 60% | F1 | 0.771±0.018 | 0.809±0.019 | 0.823±0.007 | 0.842±0.015 | 0.854±0.008 | 0.851±0.005 | **0.860±0.006** |
| | | BACC | 0.798±0.022 | 0.838±0.016 | 0.850±0.006 | 0.867±0.014 | 0.877±0.007 | 0.875±0.005 | **0.883±0.006** |
| | 40% | F1 | 0.743±0.020 | 0.782±0.006 | 0.784±0.012 | 0.808±0.010 | 0.823±0.010 | 0.816±0.003 | **0.827±0.007** |
| | | BACC | 0.780±0.011 | 0.814±0.006 | 0.817±0.009 | 0.838±0.009 | 0.851±0.009 | 0.846±0.003 | **0.855±0.006** |

## *B. Causal Topic Analysis (RQ2)*

In this work, we first discover causal topics based on causal inference methods and then use them for heterogeneous graph learning. To examine the effect of causal topics in the proposed model, we conduct two experiments varying the causal topics involved in the proposed model.

*a) Multi-view and Single-view Causal Topics:* We propose to utilize multi-view causal topics in our model. Here, we examine the effect of single-view causal topics in the model training. We define single-view causal topics as causal topics discovered in one setting, e.g., when the size of prediction window $m$ is 3. The prediction results on the four datasets in terms of F1 score are shown in Fig. 3. The bars of "N/A" show the results of our model variant, which removes the causality-aware message passing module. "ALL" means our model with multi-view causal topics (the base model). "3,7,14" represent our model with single-view causal topics when the size of prediction window $m$ is 3, 7, or 14, respectively. We observe that the results are sometimes better than "ALL" when $m$ is 7 or 14. This can be explained by the fact that short-term causal topics may have less impact when we predict more distant future events (e.g., predicting events within the next 5 days). In this case, noisy information may be included in "ALL". We also notice that "7" or "14" achieves the best F1 score in the different datasets. It may be due to the different causal effects of causal topics. For example, the ATEs of positive causal topics obtained when $m = 7$ might be larger than those obtained when $m = 14$. Thus, causal topics with $m = 7$ show greater help for prediction. We will explore the impact of causal topics on event prediction more in future work.

*b) Qualities of Causal Topics:* One limitation of our model is its dependence on the causal topics discovered in advance. It is difficult even for human experts to pinpoint the causal topics, i.e., the ground truth. To test how our model is affected by the quality of the causal topics, we varied the significance level in the causal analysis to generate different numbers of causal topics. In Table  III, we report the prediction results for the THA and AFG datasets, accounting for the apparent change in the number of causal topics when varying the significance level. We fix the hyperparameters of the model when we run different experiments. The results show that, in general, the model produces better prediction results when the significance level was relatively high, such as 99% or 95%, compared to 80%. It implies that involving more causal topics in which we have less confidence will deteriorate performance. In the THA dataset, the best results are obtained using causal topics with an importance level of 95%, probably because causal topics that are not within the 99% importance level are also important for model training. Thus, there is a trade-off between involving fewer causal topics with high confidence or involving more causal topics that may sacrifice confidence.



(a) THA

(b) AFG

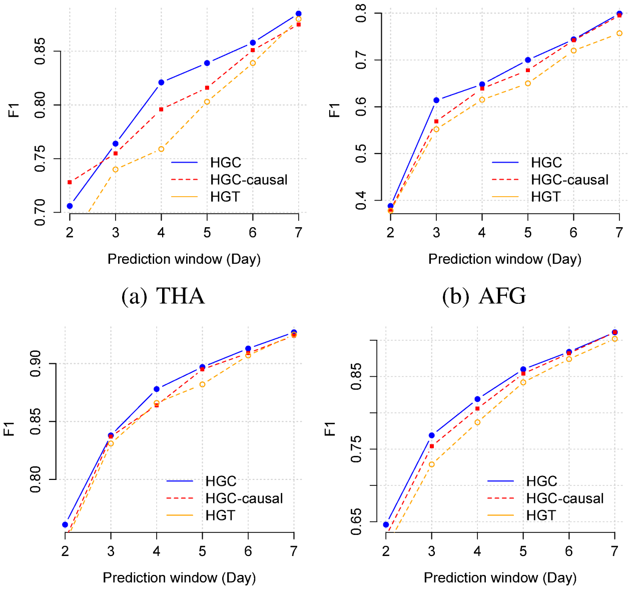
(c) EGY

(d) RUS

Fig. 3: Sensitivity analysis on multi-view and single-view causal topics. X-axis indicates causal topics used in our model.

TABLE III: Event prediction results when selecting causal topics with different confidence levels. #Pos/#Neg indicates that the average number of topics per sample that has a positive/negative causal effect on future protests.

| | 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 |



(a) THA (b) AFG

(c) EGY (d) RUS

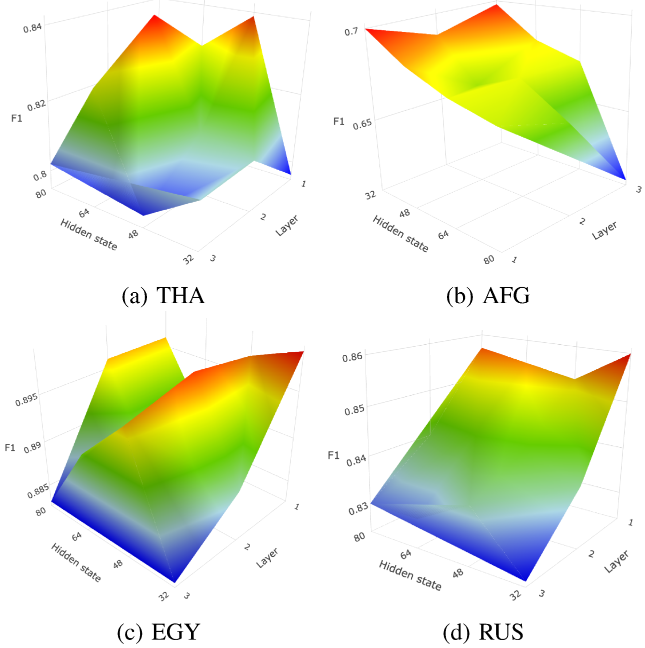
Fig. 4: Sensitivity analysis on prediction window size.

## C. Sensitivity Analysis (RQ3)

We evaluate the sensitivity of our model in the following two experiments.

*a) Prediction Window Size:* We investigate the performance of our model for different prediction window sizes from 2 to 7. The F1 score results on the four datasets are reported in Fig. 4. We report the results of our model and the best baseline model: HGT. From the results, we observe that our proposed model consistently outperforms the HGT model. When the prediction window is 2, our model without a causal component beats the others on the THA dataset. However, it is worth mentioning that for all other settings, our base model achieves the best prediction results. The results demonstrate the strength of our model in predicting future events for both short and long time windows.

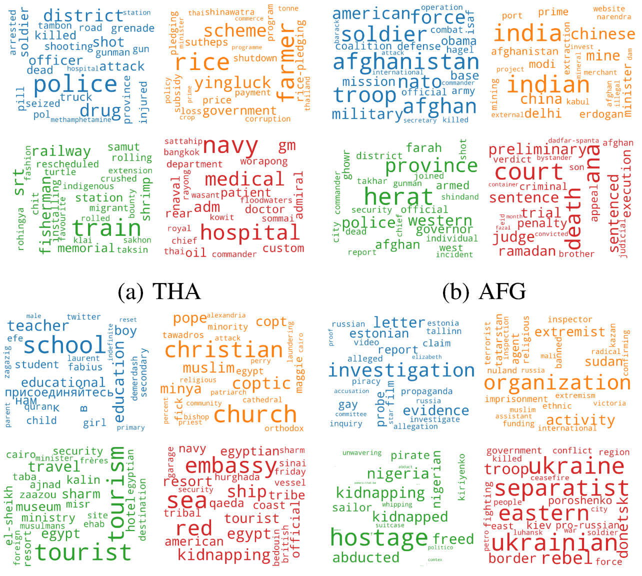
*b) Model Hyperparameters:* We study the effect of two hyperparameters, including the dimension of the hidden states and the number of graph layers. We show the results of our model on the four datasets in Fig. 5. We can observe that increasing the hidden states dimension and the number of graph layers simultaneously will degrade the performance to a great extent. It is because the large size of the model makes it easier to over-fitting when the sample size is relatively small (around 1000 to 3000 samples for all datasets). Setting the number of layers to 1 and tuning the dimensionality of the hidden states, the model produces good prediction results.



(a) THA (b) AFG

(c) EGY (d) RUS

Fig. 5: Sensitivity analysis on model hyperparameters.



(a) THA (b) AFG

(c) EGY (d) RUS

Fig. 6: Examples of causal topics discovered for each dataset displayed using Word Cloud [33].

### D. Case Study and Discussion

We summarize the causal topics discovered from the causal inference method (Sec. IV-A2). We show four topics that have positive causal effects for four datasets in Fig. 6. We observe that given the different social contexts, the causal topics differ for each country. The identified causal information is not determinative for the occurrence of future events, given complex and changing social environments. Nevertheless, this work takes the first step to explore the possibility of incorporating causal information into societal event prediction models. Through our work, we hope to expand the discussion of potential research directions for societal event prediction and combine quantitative and qualitative analysis to better understand societal events.

There are some limitations of this work. The first one is that our model relies on pre-detected topics that may have a causal impact on future events. The discovery process uses causal inference algorithms and observational data, i.e., news and events. When observational data are limited, we may not be able to obtain causal topics and may need to perform manual analysis. Secondly, the proposed model is limited in terms of its generalizability. In this study, we use country-specific data to detect causal topics and train a model for each country. It restricts its ability to handle more complex situations, e.g., cross-country prediction.

## VII. Conclusion and Future Work

Predicting societal events is beneficial for decision-making and resource allocation, and modeling the causality of events can help people understand more about the underlying mechanisms. In this paper, we propose a new approach that discovers possible causal topics for future events and incorporates this causal information into a heterogeneous graph learning framework by considering these topics as key nodes in the graph. We demonstrate the effectiveness of the proposed model on real-world event datasets. We analyze the impact of causal topics in our model from two aspects: (1) multi-view and single-view topics, and (2) causal topics with higher or lower confidence. We also provide case studies that summarize possible causal topics in different national contexts and discuss the goals of this work for expanding the potential study of societal events.

## Acknowledgments

This work is supported in part by the US National Science Foundation under grants 1948432 and 2047843.

## References

[1] S. Bonner and F. Vasile, "Causal embeddings for recommendation," in *RecSys*, 2018, pp. 104–112.

[2] J. Li, X. Jia, H. Yang, V. Kumar, M. Steinbach, and G. Simon, "Teaching deep learning causal effects improves predictive performance," *arXiv:2011.05466*, 2020.

[3] J. Chen, X. Wu, Y. Hu, and J. Luo, "Spatial-temporal causal inference for partial image-to-video adaptation," ser. AAAI, vol. 35, no. 2, 2021, pp. 1027–1035.

[4] Y. Ning, S. Muthiah, H. Rangwala, and N. Ramakrishnan, "Modeling precursors for event forecasting via nested multi-instance learning," in *KDD*. ACM, 2016, pp. 1095–1104.

[5] S. Deng, H. Rangwala, and Y. Ning, "Learning dynamic context graphs for predicting social events," ser. KDD, 2019, pp. 1007–1016.

[6] ——, "Dynamic knowledge graph based multi-event forecasting," ser. KDD, 2020, pp. 1585–1595.

[7] ——, "Understanding event predictions via contextualized multilevel feature learning," ser. CIKM, 2021, pp. 342–351.

[8] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of computational science*, vol. 2, no. 1, pp. 1–8, 2011.

[9] H. Achrekar, A. Gandhe, R. Lazarus, S.-H. Yu, and B. Liu, "Predicting flu trends using twitter data," in *IEEE Conference on Computer Communications Workshops*. IEEE, 2011, pp. 702–707.

[10] X. Wang, M. S. Gerber, and D. E. Brown, "Automatic crime prediction using events extracted from twitter posts," ser. SBP-BRiMS. Springer, 2012, pp. 231–238.

[11] F. Qiao, P. Li, X. Zhang, Z. Ding, J. Cheng, and H. Wang, "Predicting social unrest events with hidden markov models using gdelt," *Discrete Dynamics in Nature and Society*, vol. 2017, 2017.

[12] F. Qiao, X. Zhang, and J. Deng, "Learning evolutionary stages with hidden semi-markov model for predicting social unrest events," *Discrete Dynamics in Nature and Society*, vol. 2020, 2020.

[13] L. Zhao, J. Ye, F. Chen, C.-T. Lu, and N. Ramakrishnan, "Hierarchical incomplete multi-source feature learning for spatiotemporal event forecasting," ser. KDD, 2016, pp. 2085–2094.

[14] A. M. Ertugrul, Y.-R. Lin, W.-T. Chung, M. Yan, and A. Li, "Activism via attention: interpretable spatiotemporal learning to forecast protest activities," *EPJ Data Science*, vol. 8, no. 1, pp. 1–26, 2019.

[15] C. Huang, C. Zhang, J. Zhao, X. Wu, D. Yin, and N. Chawla, "Mist: A multiview and multimodal spatial-temporal learning framework for citywide abnormal event forecasting," ser. WWW, 2019, pp. 717–728.

[16] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," ser. ESWC. Springer, 2018, pp. 593–607.

[17] S. Vashishth, S. Sanyal, V. Nitin, and P. Talukdar, "Composition-based multi-relational graph convolutional networks," *arXiv:1911.03082*, 2019.

[18] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," ser. KDD, 2019, pp. 793–803.

[19] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv:1710.10903*, 2017.

[20] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," ser. TheWebConf, 2020, pp. 2704–2710.

[21] M. Hoffman, F. Bach, and D. Blei, "Online learning for latent dirichlet allocation," vol. 23, 2010.

[22] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," ser. AAAI, vol. 33, no. 01, 2019, pp. 7370–7377.

[23] J. S. Sekhon, "The neyman-rubin model of causal inference and estimation via matching methods," *The Oxford handbook of political methodology*, vol. 2, pp. 1–32, 2008.

[24] M. Caliendo and S. Kopeinig, "Some practical guidance for the implementation of propensity score matching," *Journal of economic surveys*, vol. 22, no. 1, pp. 31–72, 2008.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," ser. NIPS, 2017, pp. 5998–6008.

[26] E. Boschee, J. Lautenschlager, S. O'Brien, S. Shellman, J. Starz, and M. Ward, "Icews coded event data," 2015.

[27] A. Pareja, G. Domeniconi, J. Chen, T. Ma, T. Suzumura, H. Kanezashi, T. Kaler, T. Schardl, and C. Leiserson, "Evolvegcn: Evolving graph convolutional networks for dynamic graphs," in *AAAI*, vol. 34, no. 04, 2020, pp. 5363–5370.

[28] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017.

[29] P. Gupta, M. Pagliardini, and M. Jaggi, "Better word embeddings by disentangling contextual n-gram information," *arXiv:1904.05033*, 2019.

[30] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, 2010, pp. 249–256.

[31] D. Kinga and J. B. Adam, "A method for stochastic optimization," ser. ICLR, vol. 5, 2015.

[32] M. Wang, D. Zheng, Z. Ye, Q. Gan, M. Li, X. Song, J. Zhou, C. Ma, L. Yu, Y. Gai, T. Xiao, T. He, G. Karypis, J. Li, and Z. Zhang, "Deep graph library: A graph-centric, highly-performant package for graph neural networks," *arXiv:1909.01315*, 2019.

[33] F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, "Word cloud explorer: Text analytics based on word clouds," ser. HICSS. IEEE, 2014, pp. 1833–1842.