CDEP: Cross-domain Event Prediction Based on Continuous Learning

Yuxin Zhang
Southeast University
NanJing, China
zzyx_cs@seu.edu.cn

Deqiang Yin
Southeast University
NanJing, China
gmrqiang@gmail.com

Zhenqian Wu
Jiangsu Automation Research Institute
Lianyun Gang, China
254088544@qq.com

Yike Wang
Jiangsu Automation Research Institute
Lianyun Gang, China
upcwyk@163.com

Shengning Chen
Jiangsu Automation Research Institute
NanJing, China
352284858@qq.com

Guilin Qi*
gqi@seu.edu.cn
School of Computer Science and Engineering, Southeast University
Nanjing, China

ABSTRACT

Event prediction aims to extrapolate potential future events by analyzing patterns and trends inherent in historical events. The internet is replete with a plethora of event information, some of which pertain to events with profound ramifications on both the natural environment and society at large, such as natural calamities, acts of terrorism, the proliferation of infectious diseases, and criminal incidents. Proactively predicting the emergence of these events, or forecasting their subsequent trajectories, can serve as an invaluable strategy in attenuating their detrimental impacts on human civilization. Existing event prediction methods are inherently crafted to cater to specific application realms, rendering models adept in one domain ineffective in forecasting events across disparate domains. Nevertheless, irrespective of the semantic or structural divergences characterizing events across these domains, there exists an underlying commonality in event development trends. To bridge this research gap, this article presents a novel event prediction model, dubbed CDEP that amplifies the precision of event forecasting by identifying parameters quintessential to domain-specific event prediction, and restricting the update of parameters related to event prediction commonalities, facilitating their adaptation or seamless transition across previously uncharted domains. Experimental results across two public benchmark datasets unequivocally highlight the substantial enhancement brought about by continual learning, CDEP could outperform the state-of-the-art model by a considerable margin.

CCS CONCEPTS

• **Computing methodologies → Lifelong machine learning**.

KEYWORDS

event prediction, continual learning, graph neural networks

ACM Reference Format:
Yuxin Zhang, Deqiang Yin, Zhenqian Wu, Yike Wang, Shengning Chen, and Guilin Qi. 2018. CDEP: Cross-domain Event Prediction Based on Continuous Learning. In *Proceedings of the International Joint Conference on Knowledge Graphs (IJCKG'23).* ACM, New York, NY, USA, 9 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Event prediction aims to extrapolate potential future events by analyzing patterns and trends inherent in historical events. The internet is replete with a plethora of event information, some of which pertain to events with profound ramifications on both the natural environment and society at large, such as natural calamities, acts of terrorism, the proliferation of infectious diseases, and criminal incidents. Proactively predicting the emergence of these events, or forecasting their subsequent trajectories, can serve as an invaluable strategy in attenuating their detrimental impacts on human civilization.

The significance of event prediction has led to the proliferation of various prediction models across different domains. For instance, meteorologists employ specialized models to forecast weather patterns, while law enforcement agencies utilize distinct algorithms to anticipate criminal activities. However, a glaring challenge faced by the current landscape of event prediction is the confinement of these models within their specific realms of application. A model adept at predicting earthquakes might falter when it comes to forecasting a pandemic's spread. This limitation stems from the fact that these models, although designed for specific applications, do not capitalize on the fundamental similarities underlying event development trends across diverse domains. Despite the evident semantic differences in events across varied domains, the underlying patterns governing their emergence and progression bear certain commonalities. Leveraging these commonalities can pave the way for a more universal, cross-domain event prediction model that can adapt and perform proficiently across different domain scenarios.

In recent years, there has been notable progress in continual learning methods, particularly in addressing the challenging issue of catastrophic forgetting in neural network models. Catastrophic

*Corresponding author.

IJCKG'23, December 8 to 9, 2023, Miraikan, Tokyo, Japan

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

forgetting occurs when neural networks tend to "forget" previously acquired knowledge when learning new tasks or domains. The integration of continual learning techniques into event prediction models has emerged as a promising approach to mitigate this problem. By incorporating continual learning strategies, event prediction models can retain and leverage common knowledge across various domains, enhancing their adaptability to cross-domain event prediction tasks. In our observations, textual descriptions of the same event often exhibit redundancy and noise, which can inadvertently introduce misleading into predictive models. For a comprehensive understanding of events, it's paramount that models adeptly discern the sophisticated structural nuances of events as well as their intrinsic evolutionary trend patterns. To address this challenge, we advocate for the transformation of event-related textual data into a structured graphical representation. Nevertheless, continual learning methods based on graph-structured data are still immature. Existing graph-based continual learning methods can be categorized as regularization methods[1, 3, 11, 20], which restrict the update of key parameters; parameter isolation methods[28], which assign different parameters to different tasks; and memory replay methods[29], which store representative nodes. These aforementioned methods are still defective in practical applications and cannot be directly applied to downstream tasks.

In light of the challenges previously highlighted, we present an innovative Cross-Domain Event Prediction (CDEP) method anchored in graph continual learning. CDEP takes gnn as the backbone of event prediction models and improves GNN to model complex entity and event heterogeneous knowledge in event graphs in order to facilitate the extraction of embedded node representations. Building upon this foundation, we integrate a graph continual learning methodology into the event prediction framework. This integration empowers the model to seamlessly encode shared knowledge spanning multiple domains[4, 10], thereby enhancing its adaptability for cross-domain event prediction tasks. In addition, design a memory playback sample sampling strategy for cross-domain event prediction. Experimental results across two public benchmark datasets unequivocally highlight the substantial enhancement brought about by continual learning, CDEP could outperform the state-of-the-art model by a considerable margin.

2 RELATED WORK

2.1 Event Prediction

Event prediction methods can be broadly classified into two primary categories based on their utilization of historical event data and their approach to forecasting future events: rule-based and sequence-based methods.

Rule-based methods typically mine associations between events and their precursors using methods based on frequent set [9]. Specifically, for a new input x', the typical strategy is to compute the set of events triggered by any association rule starting from x' as proposed by Han et al.[9], and then rank them according to the confidence level, retaining the most important ones.

Sequence-based methods are mainly categorized into complete sequence classification and event generation. Complete sequence classification methods treat event prediction as a multi-classification problem, where each class represents a semantic class of possible next events[7]. Typically, as proposed by Tama et al.[21], this is done by comparing the input sequence with the features of each category thereby identifying the category with the higher probability value as the next event.

Event generation methods usually fall into two categories, attribute-based methods and description-based methods. The attribute-based method generates a vector of events at the $t+1$ moment by encoding attributes such as location, subject, and semantic category, which is then decoded by a decoder to finally obtain various attributes for the next event [6]. Su et al.[19] state that the description-based methods typically encode a textual sequence of events into a vector representation of events using a recurrent neural network. The vector representation of each event of the sequence of events is then fed into a higher-level recurrent neural network to generate predicted event vectors[25].

2.2 Continual Learning

Unlike traditional deep learning models that can only acquire static knowledge, continuous learning is able to continuously acquire, update, memorize, and utilize knowledge in an ever-changing environment. Two approaches to continuous learning are introduced: weight regularization and replay-based methods.

Weight regularization methods aim to impose constraints on the update range of the weight parameters in a neural network. Among them, the EWC(Elastic Weight Consolidation)[13] method is a typical weight regularization method that preserves the knowledge of the old task by restricting the difference between the old weights and the new ones.

The main idea of memory replay is to store old training samples by storing a smaller subset of the old task that can be used to train together with new data or to constrain the generative network. The biggest difficulty of memory replay lies in the need to repeatedly sample, augment, and replace the stored training samples during construction. Early sampling methods mostly used random sampling such as the reservoir sampling algorithm [17] and the ring buffer algorithm [16]. The drawback of the memory replay method is that there is a risk of overfitting when dealing with old training samples. This is due to the fact that only a small portion of the old training samples are stored in the memory buffer, which affects the generalization ability of the model[23].

3 PRELIMINARIES

3.1 Problem Statement

In practical scenarios, real-world events often comprise multiple atomic sub-events intricately interwoven through relations, such as causal and temporal relations. Such aggregations of atomic events and their interrelations are termed complex events. The event prediction task defined in this paper is a narrow semantic-based event prediction. Given a set of documents, denoted as $\mathcal{D}$, that portrays a singular complex event, the model predicts the semantics of possible future events, i.e., event types. This is formalized by the expression: $\hat{C} = \text{argmax}_{C_i \in C} P\left(C_i | s_1, s_2, \cdots, s_T\right)$, where s_T denotes the event type of the atomic event e_T in the complex event, and $\hat{C}$ is the optimal prediction in the candidate set of event types $C = \{C_1, \cdots, C_N\}$, N is the number of atomic event types included in the complex event. To accurately comprehend complex events, it is imperative for models to not only internalize representations of atomic events but also to discern the intricate structural characteristics of complex events and their inherent evolutionary trajectories. Consequently, a pragmatic approach involves converting these complex events into a graph-structured format.

For a given document set $\mathcal{D}$, that delineates a specific complex event, we employ advanced information extraction methodologies to construct an instance graph $\mathcal{G}$. This graph encompasses a set of event nodes E, and a distinct set of entity nodes V. Interconnecting these nodes, we introduce a set of relational edges $\mathcal{R}$, which articulate the intricate relationships and interactions within the event structure. In the context of continual learning, the model undergoes sequential training across a succession of event prediction tasks $\mathcal{T} = \mathcal{T}_1, \cdots, \mathcal{T}_K$ that are distinctly domain-disjointed. Every task $\mathcal{T}_k$ encompasses a set of complex event instance graphs designated for both training G_k^{tr} and evaluation G_k^{te}, accompanied by their respective event prediction labels $\mathcal{Y}_k = y_1, \cdots, y_{k_N}$. Here, k_N represents the number of atomic event types present within the task $\mathcal{T}_k$. This article aims to develop an event prediction model tailored for a sequence of distinct domain-specific tasks. The overarching objective is to ensure the model's proficiency in forecasting events within novel domains while preserving its predictive accuracy for previously encountered domains.

3.2 Event Graph Construction

To effectively capture the intricate relationships among events and the inherent structural attributes of complex events, this section advocates for converting a document collection from a corpus, which delineates the same complex event, into a corresponding graph-structured representation. We employ the state-of-the-art information extraction system OneIE[15] to extract entity, relation, and event information. Subsequently, for a collection of documents that describe a singular complex event, co-reference disambiguation is executed across the extracted entities and events. Upon acquiring the discrete event nodes, we ascertain both temporal and causal interrelations among events utilizing the derivative prompt joint learning event relations extraction method. The instance graph of the constructed complex event is denoted as $\mathcal{G}$. This graph $\mathcal{G}$, encompasses a set of event nodes E, along with a set of entity nodes V. Further, it incorporates a set of relational edges $\mathcal{R}$ linking these nodes. The relation edges $\mathcal{R}$ can be classified into three distinct categories: (1) event-event edges $\mathcal{R}_{eve-eve}$ connecting the directed temporal or causal edges between two events; (2) event-entity edges $\mathcal{R}_{eve-ent}$ connecting attribute edges between events and entities; and (3) entity-entity edges $\mathcal{R}_{ent-ent}$ connecting relationship edges between entities.

4 METHODOLOGY

The proposed CDEP model is articulated around three core modules: an event prediction backbone, a topology-aware continual learning mechanism, and a memory replay strategy tailored for graph-structured data. Given an instance graph of complex events, the backbone module forecasts potential future event types. To maintain prediction efficacy across historic domains, CDEP adopts a nuanced continual learning mechanism. CDEP utilizes a topology-aware continual learning-based approach to slow down the update

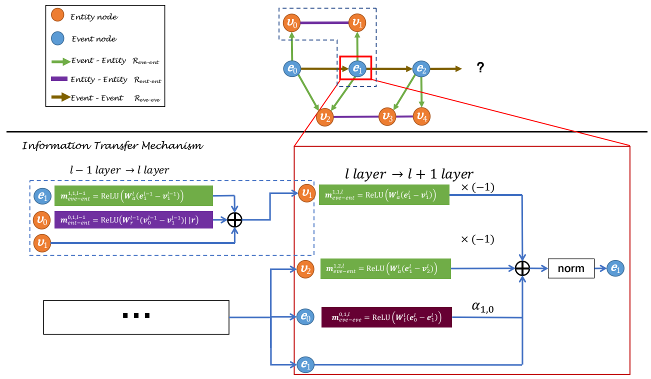

Figure 1: Event Prediction message passing mechanism

rate of key parameters affecting the prediction performance, thus preserving the predictive capability of the event prediction model on historic domains. Complementing this, our uniquely crafted memory replay strategy, optimized for graph-structured data, provides an additional module to counteract the model's catastrophic forgetting tendencies.

4.1 Event Prediction Model

The CDEP model integrates a relational graph convolutional network as its foundational architecture for event prediction. To adeptly unearth underlying patterns in potential event evolution, a bespoke message-passing mechanism is designed for each distinct relationship type, as visually depicted in Figure 1. CDEP initializes the semantic representation of entities in the complex event instance graph by pre-training the BERT model and employs a linear tensor operation to aggregate the argument information of the event nodes, the initialized semantic encoding of the atomic event e_i is:

$$\boldsymbol{e}_i^0 = \mathrm{MLP}(\sum_{j \in \mathcal{N}_{ent}^i} \mathrm{BERT}(w_j)), \tag{1}$$

where $\mathcal{N}_{ent}^i$ denotes the set of argument nodes neighboring the atomic event e_i and w_j denotes the semantic text of the argument entity. Conventional relational graph convolutional networks adopt unique edge representations for each type of edge. However, this approach can cause an exponential surge in model parameters as the edge types increase, rendering it unsuitable for cross-domain contexts. Addressing this, CDEP conceptualizes edges within the complex event instance graphs into three distinct categories: $\mathcal{R}_{eve-ent}$, $\mathcal{R}_{eve-eve}$, and $\mathcal{R}_{ent-ent}$. Specifically, $\mathcal{R}_{eve-ent}$ represents a unidirectional edge from an event node to its corresponding argument node; $\mathcal{R}_{eve-eve}$ signifies a unidirectional edge from one event node to another, and $\mathcal{R}_{ent-ent}$ corresponds to the undirected edge interconnecting entities. In tandem with this, CDEP introduces specialized messaging mechanisms for each edge type, ensuring a more refined aggregation of the semantic and structural attributes inherent to atomic events. The passing information of the three edges in the l layer of the graph neural network is defined as follows:

$$\boldsymbol{m}_{eve-ent}^{i,j,l} = \mathrm{ReLU}\left(\boldsymbol{W}_a^l\left(\boldsymbol{e}_i^l - \boldsymbol{v}_j^l\right)\right) \tag{2}$$

$$\boldsymbol{m}_{ent-ent}^{i,j,l} = \mathrm{ReLU}\left(\boldsymbol{W}_r^l\left(\boldsymbol{v}_i^l - \boldsymbol{v}_j^l\right)||\boldsymbol{r}\right) \tag{3}$$

$$\boldsymbol{m}_{eve-eve}^{i,j,l} = \mathrm{ReLU}\left(\boldsymbol{W}_t^l\left(\boldsymbol{e}_i^l - \boldsymbol{e}_j^l\right)\right) \tag{4}$$

where $\boldsymbol{W}_a, \boldsymbol{W}_r, \boldsymbol{W}_t$ denote the linear variation matrices of the edges $\mathcal{R}_{eve-ent}$, $\mathcal{R}_{ent-ent}$, and $\mathcal{R}_{eve-eve}$, respectively, and $||$ denotes vector concatenation operations; In addition, CDEP learns a specific vector representation $\boldsymbol{r}$ for $\mathcal{R}_{ent-ent}$. Entities directly aggregate neighborhood information to update node representations. Atomic events need to consider not only neighboring entities and events but also the degree of association of neighboring events. Therefore, this section uses edge-aware attention to aggregate information between events and events or entities:

$$\alpha_{i,j} = \sigma\left(\mathrm{MLP}\left(\boldsymbol{e}_i - \boldsymbol{e}_j\right)\right) \tag{5}$$

$$\boldsymbol{e}_i^{l+1} = norm\left(\boldsymbol{e}_i^l + \sum_{j\in\mathcal{N}_{eve}^i}\alpha_{i,j}\boldsymbol{m}_{eve-eve}^{i,j,l} + \sum_{k\in\mathcal{N}_{ent}^i}\boldsymbol{m}_{eve-ent}^{i,k,l}\right) \tag{6}$$

$$\boldsymbol{v}_i^{l+1} = \boldsymbol{v}_i^l + \sum_{j\in\mathcal{E}_{eve}^i}\boldsymbol{m}_{\boldsymbol{eve-ent}}^{j,i,l} \tag{7}$$

$\mathcal{N}_{ent}^i$ denotes the atomic event e_i neighboring argument nodes, $\mathcal{N}_{eve}^i$ denotes the set of neighboring event nodes for which there exist directed edges pointing to e_i, and $\mathcal{E}_{eve}^i$ denotes the entity node v_i of the set of neighboring event nodes $norm(\bullet)$ is the regularization function. CDEP pools all atomic event representations in the complex event graph to obtain the graph representation g, by which g predicts the event type c of future event nodes:

$$\boldsymbol{c} = \mathrm{Softmax}\left(\mathrm{MLP}\left(\mathrm{Pooling}\left(\left\{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_{|E|}\right\}\right)\right)\right) \tag{8}$$

Pooling operations can be alternatively average pooling, maximum pooling, etc., after which future event types are predicted by a multilayer perceptron.

4.2 Continual Learning based on Topology Awareness

Upon completing the training process for domain task $\mathcal{T}_k$, the model acquires a set of optimized parameters denoted as θ_k^*. These parameters are fine-tuned to minimize the event prediction loss specific to the domain. However, it's important to recognize that not all parameters within the GNN contribute equally to the model's performance. Inspired by the TWP method [11], CDEP models the attention coefficients between the central node and its first-order neighboring nodes as topological features of the center node. Thus, according to Eqs. 4, 5, and 6, the attention coefficients of the $\mathcal{R}_{eve-eve}$ edges can be rewritten as:

$$atte_{eve-eve}^{i,j,l} = f_{eve-eve}\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{e}_j^{l-1}; \theta_{eve-eve}^l\right) \tag{9}$$

where $f(\bullet)$ denotes the neural network projection of the attention coefficients obtained from the node representations; $\theta_{eve-eve}^l$ denotes the parameter associated with the computation of the attention coefficients of the $\mathcal{R}_{eve-eve}$ edges; $\boldsymbol{e}_i^{l-1}$, $\boldsymbol{e}_i^{l-1}$, $\boldsymbol{e}_j^{l-1}$ then denote the feature embedding of nodes e_i and e_j in layer l, respectively. Unlike $\mathcal{R}_{eve-eve}$, $\mathcal{R}_{eve-ent}$ edges do not have an attention mechanism. Therefore CDEP uses the distance between nodes as the attention coefficient, which is calculated as follows:

$$\begin{aligned} atte_{eve-ent}^{i,j,l} &= (\boldsymbol{e}_i^{l-1}W_a^l)^T \tanh(\boldsymbol{v}_j^{l-1}W_a^l) \\ &= f_{eve-ent}(\boldsymbol{e}_i^{l-1}; \boldsymbol{v}_j^{l-1}; \theta_{eve-ent}^l) \end{aligned} \tag{10}$$

where $\theta_{eve-ent}^l = W_a^l$ and the attention coefficient $atte_{eve-ent,i,j}^l$ is represented by the distance between e_i and v_j. During continual learning, the model only focuses on the topological information of the event nodes, so only the parameters related to the $\mathcal{R}_{eve-eve}$, $\mathcal{R}_{eve-ent}$ edges are considered. The model loss for the domain task $\mathcal{T}_k$ is denoted as $\mathcal{L}(G_k^{tr};\theta), \theta = \{\theta_{eve-eve}, \theta_{eve-ent}\} = \{\mathcal{W}_{m+n}\}$ contains all the parameters of the neural network. When a small change occurs in one of the parameters of the neural network, the change in the structural characteristics of the model can be approximated as:

$$\begin{aligned} &f\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{e}_j^{l-1}; \theta_{eve-eve}^l + \Delta\mathcal{W}_m^l\right) - f\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{e}_j^{l-1}; \theta_{eve-eve}^l\right) \\ &\qquad\approx \sum_m g_m\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{e}_j^{l-1}\right)\Delta\mathcal{W}_m^l, \end{aligned} \tag{11}$$

$$\begin{aligned} &f\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{v}_j^{l-1}; \theta_{eve-ent}^l + \Delta\mathcal{W}_n^l\right) - f\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{v}_j^{l-1}; \theta_{eve-ent}^l\right) \\ &\qquad\approx \sum_n g_n\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{v}_j^{l-1}\right)\Delta\mathcal{W}_n^l, \end{aligned} \tag{12}$$

$$\begin{aligned} &\sum_m g_m\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{e}_j^{l-1}\right)\Delta\mathcal{W}_m^l + \sum_n g_n\left(\boldsymbol{e}_i^{l-1}; \boldsymbol{v}_j^{l-1}\right)\Delta\mathcal{W}_n^l \\ &\qquad= \sum_{m+n} g_{m+n}\left(H_{i,j}^{l-1}\right)\Delta\mathcal{W}_{m+n}^l, \end{aligned} \tag{13}$$

where $g_{m+n}\left(H_{i,j}^{l-1}\right) = \frac{\partial f}{\partial\mathcal{W}_{m+n}}$ denotes the partial derivatives of the attention coefficients with respect to the parameter $\mathcal{W}_{m+n}$, and $H_{i,j}^{l-1}$ is the node representation of the output of the $l-1$ layer. The attention coefficients of the atomic event node e_i as the center node in layer l of the event prediction model form a multidimensional vector of:

$$\begin{aligned} atte_i^l &= atte_{eve-eve}^{i,l} \cup atte_{eve-ent}^{i,l} \\ &= \left[atte_{eve-eve}^{i,1,l}, \cdots, atte_{eve-eve}^{i,|\mathcal{N}_{eve}^i|,l}, atte_{eve-ent}^{i,1,l}, \cdots, atte_{eve-ent}^{i,|\mathcal{N}_{ent}^i|,l}\right] \end{aligned} \tag{14}$$

CDEP defines the structural feature loss as the square of the 2-paradigm of the multidimensional vector of attention coefficients for all atomic event nodes in the complex event instance graph, and computes each parameter bias:

$$g_{m+n}\left(H_{i,j}^{l-1}\right) = \frac{\partial f}{\partial\mathcal{W}_{m+n}} = \frac{\partial\left(\left|\left|\left[atte_1^l, \cdots, atte_{|E|}^l\right]\right|\right|_2^2\right)}{\partial w_{m+n}}. \tag{15}$$

This leads to an importance score $I_k = [||g_{m+n}(H_{i,j}^{l-1})||]$ for all parameters in the task $\mathcal{T}_k$ with respect to the topology of atomic event nodes. Catastrophic forgetting of event prediction tasks in cross-domain prediction tasks can be effectively mitigated by penalizing parameter changes that are critical to the impact of the old task, as shown in Figure 2. While maintaining the stability of important parameters allows the model to remember the learned task, it also results in a less malleable model that is unable to learn subsequent

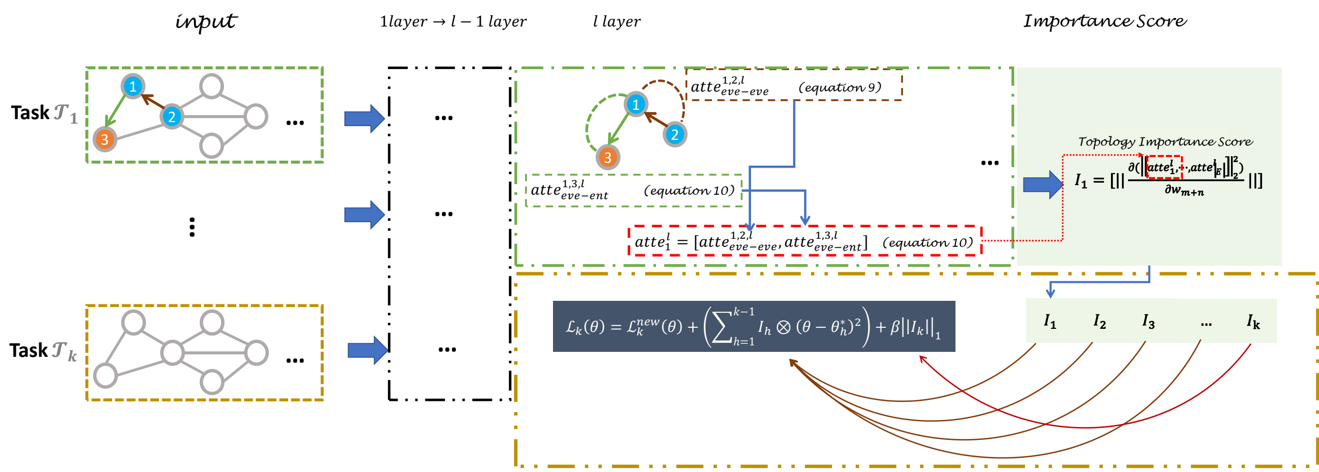

Figure 2: Continual Learning based on Topology Awareness

new tasks. In order to retain sufficient model capacity for future domains, CDEP promotes minimization of parameter importance scores by adding one-paradigm numbers as regularization terms to the importance scores. Thus, the total loss in training the current task $\mathcal{T}_k$ is:

$$\mathcal{L}_k(\theta) = \mathcal{L}_k^{new}(\theta) + \left(\sum_{h=1}^{k-1} I_h \otimes (\theta - \theta_h^*)^2\right) + \beta\, ||I_k||_1 \tag{16}$$

where $\otimes$ denotes elemental multiplication; $\mathcal{L}_k^{new}(\theta)$ denotes the prediction loss (cross-entropy loss) of the event prediction model at task $\mathcal{T}_k$; I_h represents the parameter on old task $\mathcal{T}_h$ importance score matrix; β is a hyperparameter controlling the degree of influence of the importance score.

4.3 Graph Matching Based Memory Replay Strategy

A substantial body of prior research on continual learning has demonstrated the efficacy of memory-playback-based strategies in mitigating catastrophic forgetting. Therefore, this paper proposes a graph-matching-based memory playback strategy to further alleviate the catastrophic forgetting problem of event prediction models in cross-domain tasks by playing back old tasks.

According to the objective of the memory replay strategy, CDEP needs to find the subset of data that best represents a task, and this subset of data needs to contain as many common features of the task as possible. For event prediction, CDEP selects the most representative complex event instance graphs in each domain and stores them in the memory playback module. Specifically, these filtered complex event instance graphs should be able to represent as many instance graphs of the same domain as possible, with the highest similarity between them and other instance graphs. Thus, the selection problem for memory can be converted into a problem of maximizing the total graph matching score.

When comparing two complex event instance graphs, denoted as $\mathcal{G}_N$ and $\mathcal{G}_M$, CDEP's attention is primarily directed towards the event types associated with the atomic event nodes within these graphs. Consequently, the specific atomic event nodes are abstracted into their respective event types, while the entity nodes are disregarded in this context. The assessment of graph similarity in CDEP involves a two-fold perspective: event type similarity and relation edge similarity. These two dimensions are pivotal in evaluating the likeness between complex event instance graphs. The calculation of these similarities is outlined as follows:

$$\mathrm{Sim}(\mathcal{G}_N, \mathcal{G}_M) = \sum_{X_{ij}=1} 1 + \sum_{X_{ik}=1, X_{jl}=1} 1, \tag{17}$$

where $X \in \{0,1\}^{|\mathcal{G}_N|, |\mathcal{G}_M|}$ denotes the binary matching matrix, and $X_{ij} = 1$ when, and only when, $i \in \mathcal{G}_N$ and $j \in \mathcal{G}_M$ matches. CDEP calculates the matching scores of any two complex event instance graphs in the dataset, and filters the complex event instance graph with the highest total matching score as the playback "memory".

5 EXPERIMENT

5.1 Datasets

In the present study, we engage in a systematic experimental evaluation of the proposed methodologies. To ensure robustness and generalizability, we utilize two prominent event prediction datasets: the Improved Explosive Device (IED) dataset[14] and the Chinese Emergency dataset.

The IED dataset is on improvised explosive device attacks, with corpus content taken from Wikipedia and relevant news. It contains 4 task types, namely car bomb, drone bomb, suicide bomber, and general IED attack. Its event schema definitions include 24 entity types, 48 relation types (including entity and event relation), 67 event types, and 85 argument types. For an in-depth statistical breakdown of the dataset, readers are directed to Table 1.

The Chinese Emergency Incident Dataset is a dataset about social security incidents from the Internet news corpus. It is small in size and contains five task types, namely "earthquake", "fire", "traffic accident", "terrorist attack", and "food poisoning". Its event schema

Table 1: Detailed description of the event prediction experiment dataset

Datasets	Number of tasks	Subsets	Documents	Event subsets	Events	Arguments	Relations
IED	4	Train	5247	343	41672	136894	122846
		Dev	575	42	4661	15404	13320
		Test	577	45	5089	16721	14054
the Chinese Emergency	5	Train	265	265	4763	15670	14579
		Dev	33	33	545	1793	1689
		Test	34	34	596	1963	1881

definition includes 35 event types and 25 argument types. Detailed statistics of its content are shown in Table 1.

5.2 Evaluation Metrics

Our evaluative measures for event prediction encompass Precision, Recall, and F1 score in this paper. To provide a more granular evaluation of the proposed method's performance after incremental task training, we introduce the performance matrix, denoted as $M^p \in \mathbb{R}^{T\times T}$. This matrix is on the basis of F1 scores, with T symbolizing the aggregate number of tasks. Specifically, The $M^p_{i,j}$ denotes the F1 score of the model on the jth task after training from the 1st task to the ith task. In seeking a comprehensive metric that encapsulates the holistic performance of the event prediction model, particularly in the realm of continual learning across varied domains, we adopted the AP (average performance) and AF (average forgetting) metrics, as conceptualized by Lopez et al.[16]. The foundational definitions for AP and AF are presented as follows:

$$\begin{aligned} AP &= \{\frac{\sum_{j=1}^{i} M^p_{i,j}}{i} | i = 1, \ldots, T\} \\ AF &= \{\frac{\sum_{j=1}^{i-1} M^p_{i,j} - M^p_{j,j}}{i-1} | i = 2, \ldots, T\} \end{aligned} \tag{18}$$

Specifically, for the metric AP, after the model has been trained on a series of tasks from 1 to i, the first i line of M^p: $\{M^p_{i,j} | j = 1, \ldots, i\}$ covers the model's performance on each of the previous tasks. Therefore the average of the i row is the model's average performance on all previous tasks. For the metric AF, $\{M^p_{i,j} - M^p_{j,j} | j = 1, \ldots, i-1\}$ denotes how much the model forgets on the task j after the training task i. Its average value is the average amount of forgetting for each trained task after the model has been trained from task 1 to task i.

5.3 Comparison Method

In this paper, we hope to analyze and validate the accuracy of the event prediction method we proposed in predicting the types of events that may occur in the future by comparing it with the baseline model, and the impact of a topology-aware continual learning module on the catastrophic forgetting problem in event prediction models which for cross-domain scenarios. Notably, this research stands as a pioneering effort in the realm of event prediction predicated on complex event subgraphs. Given this innovative approach, we have selected the advanced heterogeneous graph neural network as the benchmark for our event prediction method. Our baseline model bifurcates into two primary components: (1) the heterogeneous graph neural network and (2) the graph continual learning methodology.

5.3.1 Heterogeneous Graph Neural Network. The selected graph neural networks for our comparative evaluation are intricately designed to handle the heterogeneity of events and entities in complex event subgraphs. What follows is a concise overview of the sophisticated graph neural network models incorporated in our study:

(1) `GraphSAGE`[8]: A model for updating the embedding representation of each node using the embeddings of neighboring nodes.
(2) `GAT`[22](Graph Attention Networks): A graph neural network that uses a self-attention mechanism to compute the weights of neighboring nodes to update the representation of each node.
(3) `GCN`[12](Graph Convolutional Networks): A graph neural network for updating per-node representation using aggregated information from local neighbor nodes.
(4) `R-GCN`[18](Relational Graph Convolution Network):A graph neural network for modeling relational data.
(5) `HAN`[24](Heterogeneous Graph Attention Network): `HAN` can handle heterogeneous graphs consisting of multiple types of nodes and edges, and it uses different attention mechanisms to capture the relations between different types of nodes.
(6) `HetGNN`[26](Heterogeneous Graph Neural Network): `HetGNN` use a cross-type adjacency matrix to make connections between nodes of different types and aggregates them to generate feature representations.

5.3.2 Graph Continual Learning Method. This paper compares state-of-the-art graph continual learning methods based on the CGLB[27] platform. The realized graph continual learning methods are as follows:

(1) Bare model: Denotes a `CDEP` model without continual learning method, which can be considered as a lower bound for continual learning performance.
(2) `EWC`[13](Elastic Weight Consolidation): Based on the importance of the model weight to previous tasks, a secondary penalty was been added to the model weights to maintain its performance on existing tasks.
(3) `MAS`[2](Memory Aware Synapses): `MAS` assesses the importance of a parameter based on the sensitivity of the prediction to the parameter.

(4) EMR[5](Episodic Memory Replay): The method trains the model based on data from the current task and a small number of in-memory instances.
(5) GEM[16](Gradient Episodic Memory):The method stores representative data in scene memory. During learning, GEM uses the gradient computed from the stored data to modify the gradient of the current task in order to prevent an increase in the forgetting of previous tasks.
(6) Oracle Model: Does not follow the continual learning setting and trains all tasks simultaneously. Thus, the Oracle Model has no forgetting problem and its performance can be seen as an upper bound for continual learning.

5.4 Experimental Setup

For the event prediction model, a 3-layer graph neural network is used as the main structure with a uniform embedding dimension of 768. The AdamW optimizer is chosen to use, the initial learning rate is set to 1e-4, and the graph representation is uniformly operated using average pooling. In the event prediction model performance evaluation process does not consider the continual learning scene and only compares the prediction performance of heterogeneous graph neural networks. Therefore, in this experiment, the event prediction model performance needs to be evaluated by integrating data from all domains into a complete dataset, including training and testing data.

Within the scope of our graph continual learning experiments, we've established specific hyperparameters for the benchmarks under consideration. For both the EWC and MAS baselines, the regularization hyperparameter is determined at 10,000. For the topology-aware continual learning methodology, we've calibrated the hyperparameter within a range of 0.1 to 0.01. Furthermore, when employing the graph-matching-based memory playback approach, we typically select a memory capacity of 10, i.e., the memory module will store 10 items of training data under this task. For enhanced robustness and accuracy in our evaluations, each experiment within this study was executed five times, leveraging varied random seeds. The mean value across these iterations is then adopted as the definitive outcome for experimental appraisal.

5.5 Comparative Experimental Results and Analysis

Table 2 delineates a comparative evaluation of the performance of various graph neural networks on the event prediction task. An insightful scrutiny of these results elucidates that certain graph neural networks, namely GraphSAGE, GAT, and GCN, which lack the capability to model edge features, manifest suboptimal prediction efficacy. This limitation arises from their inability to distinguish between entities and event nodes, rendering them ineffective in discerning potential trajectories in event progression. In juxtaposition, advanced models such as R-GCN, HAN, and HetGNN, which are adept at handling heterogeneous graphs, exhibit superior prediction metrics. These models are distinguished by their proficiency in modeling features from diverse edges and have some ability to identify the trend of event evolution.

Notably, the R-GCN model, while being the most straightforward and simple in its design, outperforms the likes of HAN and HetGNN in predictive accuracy. For our event prediction model presented in this study, we introduce a particular information transfer mechanism catering to diverse inter-node relationships. By leveraging edge-aware attention, our model discerns trends in inter-event progression based on the relational magnitude between adjacent atomic events. Consequently, our proposed event prediction model registers state-of-the-art (SOTA) performance across both datasets, achieving an enhancement of 5.3% in the F1 score over other leading-edge methods.

Table 3 elucidates the performance matrix for cross-domain continual learning, detailing the prediction accuracy across each task within the event prediction continual learning scene. When juxtaposed against the Bare model, all continual learning methods exhibit notable enhancements, affirming the presence of a pronounced catastrophic forgetting challenge in the cross-domain event prediction task. Note that the AF metrics are not applicable to the Oracle Model, given its training regimen that concurrently utilizes the aggregated data from all tasks, without following the continual learning framework. Evaluating the resultant AP and AF metrics, each methodology surpasses the Bare Model, with the regularization-centric EWC, MAS, and TWP frameworks being particularly commendable. Such observations underscore the efficacy of regularization strategies in alleviating the catastrophic forgetting conundrum. Furthermore, the GEM framework stores representative data in situational memory and uses this to adjust the updating gradient for the current task.

The GEM framework exhibits a marked dependency on situational memory selection, leading to fluctuating performance outcomes. Nonetheless, we observed that the EMR methodology offers an elegantly simple yet effective solution for cross-domain continual learning via a straightforward memory playback strategy. Intriguingly, when benchmarked against established baseline models, the topology-aware continual learning strategy emerges superior, securing the first position in AF performance and outperforming all except the Oracle Model in AP metrics. This superior performance underlines the capability of the topology-aware approach to counteract catastrophic forgetting, achieved by slowing down the rate of change of topology-related important parameters.

5.6 Results and Analysis of Ablation Experiments

To elucidate the contributions of individual modules in our continual learning-based cross-domain event prediction model, this paper provides an in-depth exploration through ablation experiments. We employ varied subscripts to demarcate distinct experimental module configurations. Broadly, our ablation study pivots on two primary aspects: the event prediction module and the continual learning module.

For the event prediction model, the subscripts 'rgcn' and 'gat' denote the adoption of R-GCN and GAT models, respectively. The continual learning module will be appropriately adjusted according to the type of graph neural network adopted by the event prediction model, subscript 'nocl' indicates that the continual learning module is not used; 'tsacl' subscript indicates that only topology-aware based continual learning method is used; subscript 'gm' indicates that only graph-matching based memory playback based continual

Table 2: Experimental Results of Heterogeneous Graph Neural Networks

Method	IED			Chinese Emergency		
	Precision	Recall	F1	Precision	Recall	F1
GraphSAGE	54.69	90.33	68.13	62.15	87.83	72.79
GAT	63.01	81.71	71.15	70.47	78.27	74.17
GCN	56.41	92.77	70.16	63.87	83.12	72.23
R-GCN	73.19	76.13	74.63	80.65	74.80	77.62
HAN	71.94	76.12	73.97	74.01	78.45	76.17
HetGNN	75.12	73.40	74.25	82.58	70.93	76.31
CDEP	77.45	82.57	**79.93**	84.91	83.32	**84.11**

Table 3: Experiment Results of Graph Continual Learning

Method	IED		Chinese Emergency	
	AP	AF	AP	AF
Bare model	36.33	-44.76	40.59	-45.24
EWC	49.72	-31.37	53.98	-31.85
MAS	71.54	-9.55	75.80	-10.03
EMR	69.12	-11.97	73.38	-12.45
GEM	70.36	-10.73	74.62	-11.21
CDEP	**73.62**	**-7.47**	**77.88**	-7.95
Oracle Model	79.93	-	84.11	-

learning method is used; subscript full indicates that the continual learning based continual learning cross-domain event prediction in this paper model optimally. Table 4 demonstrates the results of the ablation experiments.

Table 4: Results of Ablation Experiments

Method	IED		Chinese Emergency	
	AP	AF	AP	AF
$CDEP_{rgcn}$	69.51	-10.92	73.25	-12.15
$CDEP_{gat}$	51.45	-29.94	55.74	-29.37
$CDEP_{nocl}$	36.33	-44.76	40.59	-45.24
$CDEP_{tsacl}$	68.61	-13.56	67.38	-13.61
$CDEP_{gm}$	58.26	-19.78	57.89	-20.01
$CDEP_{full}$	73.62	-7.47	77.88	-7.95

5.6.1 Event Predict Module. Ablation experiment results underscore the significance of information feature aggregation methods in graph neural networks for the performance of event prediction models. Of the models evaluated, the R-GCN emerges as the most direct and unadorned, yet it exhibits superior predictive performance over the GAT. This superior performance could be attributed to the challenges GAT faces in differentiating between the three types of edges present in complex event subgraphs. Notably, R-GCN's performance wanes in Dataset A, which encompasses more intricate event types and entity interrelations. Furthermore, as extended neighborhoods proliferate, the catastrophic forgetting challenge for R-GCN becomes more pronounced, revealing its limited scalability. In contrast, our proposed event prediction model introduces specialized information transfer methodologies tailored for distinct inter-node associations. By leveraging edge-aware attention mechanisms, the model discerns evolutionary trends of inter-events, grounded on the relational intensity of neighboring atomic events. Consequently, our model stands out, registering optimal results both in event prediction accuracy and in resistance to forgetting.

5.6.2 Continual Learning Module. Experimental outcomes reveal that both the topology-aware continual learning and the graph-matching memory-playback-based continual learning method significantly mitigate the catastrophic forgetting dilemma in event prediction models. Notably, the topology-aware continual learning method can maintain 58.63% prediction F1 on the first domain even after training the last domain task. This suggests that there are part of common structural features in the event prediction tasks of different domains, which imply the evolutionary pattern of events. Of significant interest is the observation that the graph-matching-based memory playback approach, despite its conceptual simplicity, delivers exceptionally promising results. Crucially, these two continual learning strategies are not conflicting, suggesting that their combined application could offer synergistic benefits. By harnessing them jointly, one can more robustly counteract model forgetfulness and elevate cross-domain event prediction accuracy.

6 CONCLUSION

In this paper, we propose a cross-domain event prediction method based on graph continual learning, CDEP. The method uses a graph neural network structure based on relation graph convolutional networks to model complex heterogeneous knowledge of entities and events in the event graph in order to obtain embedded representations of event graph nodes. Based on this by introducing a topology-aware continual learning approach in the event prediction model, the model encodes the common knowledge between different domains, thus better adapting to the cross-domain event prediction task. In addition, a memory playback sample sampling strategy for cross-domain event prediction is proposed. Comparison and ablation experiments for cross-domain event prediction are designed and implemented, and the experimental results verify the effectiveness of the event prediction method and continual learning approach in this paper.

7 ACKNOWLEDGMENTS

We thank the reviewers for their valuable comments. This work is partially supported by the National Nature Science Foundation of China under No. U21A20488.

REFERENCES

[1] Kian Ahrabian, Yishi Xu, Yingxue Zhang, Jiapeng Wu, Yuening Wang, and Mark Coates. 2021. Structure Aware Experience Replay for Incremental Learning in Graph-based Recommender Systems. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, Gianluca Demartini, Guido Zuccon, J. Shane Culpepper, Zi Huang, and Hanghang Tong (Eds.). ACM, 2832–2836. https://doi.org/10.1145/3459637.3482193

[2] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. 2018. Memory Aware Synapses: Learning What (not) to Forget. In *Proceedings of the 3rd European Conference on Computer Vision (Lecture Notes in Computer Science, Vol. 11207)*. 144–161.

[3] Jie Cai, Xin Wang, Chaoyu Guan, Yateng Tang, Jin Xu, Bin Zhong, and Wenwu Zhu. 2022. Multimodal Continual Graph Learning with Neural Architecture Search. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, Frédérique Laforest, Raphaël Troncy, Elena Simperl, Deepak Agarwal, Aristides Gionis, Ivan Herman, and Lionel Médini (Eds.). ACM, 1292–1300. https://doi.org/10.1145/3485447.3512176

[4] Zhibin Chen, Yuting Wu, Yansong Feng, and Dongyan Zhao. 2022. Integrating Manifold Knowledge for Global Entity Linking with Heterogeneous Graphs. *Data Intell.* 4, 1 (2022), 20–40. https://doi.org/10.1162/DINT_A_00116

[5] Cyprien de Masson d'Autume, Sebastian Ruder, Lingpeng Kong, and Dani Yogatama. 2019. Episodic Memory in Lifelong Language Learning. *CoRR* abs/1906.01076 (2019).

[6] Huilong Duan, Zhoujian Sun, Wei Dong, Kunlun He, and Zhengxing Huang. 2020. On Clinical Event Prediction in Patient Treatment Trajectory Using Longitudinal Electronic Health Records. *IEEE Journal of Biomedical and Health Informatics* 24, 7 (2020), 2053–2063.

[7] Mark Granroth-Wilding and Stephen Clark. 2016. What Happens Next? Event Prediction Using a Compositional Neural Network Model. *Proceedings of the AAAI Conference on Artificial Intelligence* 30, 1 (Mar. 2016).

[8] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS*.

[9] Jiawei Han, Micheline Kamber, and Jian Pei. 2011. *Data Mining: Concepts and Techniques* (3 ed.). Morgan Kaufmann.

[10] Danyang Hu, Meng Wang, Feng Gao, Fangfang Xu, and Jinguang Gu. 2022. Knowledge Representation and Reasoning for Complex Time Expression in Clinical Text. *Data Intell.* 4, 3 (2022), 573–598. https://doi.org/10.1162/DINT_A_00152

[11] Yiding Yang Huihui Liu and Xinchao Wang. 2021. Overcoming Catastrophic Forgetting in Graph Neural Networks. In *Proceedings of the 35th Annual Meeting of the American Association for Artificial Intelligence, AAAI*.

[12] Thomas N. Kipf and Max Welling. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *CoRR* abs/1609.02907 (2016).

[13] James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2016. Overcoming catastrophic forgetting in neural networks. *CoRR* abs/1612.00796 (2016). arXiv:1612.00796

[14] Manling Li, Sha Li, Zhenhailong Wang, Lifu Huang, Kyunghyun Cho, Heng Ji, Jiawei Han, and Clare R. Voss. 2021. Future is not One-dimensional: Graph Modeling based Complex Event Schema Induction for Event Prediction. *CoRR* abs/2104.06344 (2021).

[15] Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A Joint Neural Model for Information Extraction with Global Features. In *Proceedings of The 58th Annual Meeting of the Association for Computational Linguistics, ACL*.

[16] David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient Episodic Memory for Continual Learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS*. 6470–6479.

[17] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. 2019. Learning to Learn without Forgetting by Maximizing Transfer and Minimizing Interference. In *Proceedings of the 7th International Conference on Learning Representations, ICLR*.

[18] Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2017. Modeling Relational Data with Graph Convolutional Networks. *CoRR* abs/1703.06103 (2017).

[19] Zichun Su and Jialin Jiang. 2020. Hierarchical Gated Recurrent Unit with Semantic Attention for Event Prediction. *Future Internet* 12 (02 2020), 39.

[20] Sagar Sunkle, Deepak Jain, Krati Saxena, Ashwini Patil, Tushita Singh, Beena Rai, and Vinay Kulkarni. 2021. Integrated "Generate, Make, and Test" for Formulated Products using Knowledge Graphs. *Data Intell.* 3, 3 (2021), 340–375. https://doi.org/10.1162/DINT_A_00096

[21] Bayu Adhi Tama and Marco Comuzzi. 2019. An empirical comparison of classification techniques for next event prediction using business process event logs. *Expert Systems with Applications* 129 (2019), 233–245.

[22] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR*.

[23] Eli Verwimp, Matthias De Lange, and Tinne Tuytelaars. 2021. Rehearsal Revealed: The Limits and Merits of Revisiting Samples in Continual Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV*. 9385–9394.

[24] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S. Yu. 2019. Heterogeneous Graph Attention Network. In *WWW*. ACM, 2022–2032.

[25] Shuqi Yu, Linmei Hu, and Bin Wu. 2019. DRAM: A Deep Reinforced Intra-Attentive Model for Event Prediction. In *Knowledge Science, Engineering and Management: 12th International Conference, KSEM*. 701–713.

[26] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V. Chawla. 2019. Heterogeneous Graph Neural Network. In *Proceedings of the 25th SIGKDD Conference on Knowledge Discovery and Data Mining, KDD*. 793–803.

[27] Xikun Zhang, Dongjin Song, and Dacheng Tao. 2022. CGLB: Benchmark Tasks for Continual Graph Learning. In *NeurIPS*. http://papers.nips.cc/paper_files/paper/2022/hash/548a41b9cac6f50dccf7e63e9e1b1b9b-Abstract-Datasets_and_Benchmarks.html

[28] Xikun Zhang, Dongjin Song, and Dacheng Tao. 2023. Hierarchical Prototype Networks for Continual Graph Representation Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 4 (2023), 4622–4636. https://doi.org/10.1109/TPAMI.2022.3186909

[29] Fan Zhou and Chengtai Cao. 2021. Overcoming Catastrophic Forgetting in Graph Neural Networks with Experience Replay. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 4714–4722. https://doi.org/10.1609/aaai.v35i5.16602

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009