



Causal knowledge graph construction for enterprise innovation events in the digital economy and its application to strategic decision-making

Pengfei Wu¹ · Bingtao Xu² · Xuhan Zhang³

Received: 26 April 2025 / Accepted: 28 May 2025 / Published online: 5 June 2025
© The Author(s) 2025

Abstract

The rapid development of the digital economy has created vast streams of textual data on enterprise activities and innovations. Unlocking causal insights from these data is critical for companies seeking strategic advantages. In this paper, we propose a novel framework for constructing a causal knowledge graph of enterprise innovation events mined from digital-economy news and reports. The goal is to represent events (e.g. product launches, strategic investments) as nodes and their causal relationships as directed edges, enabling advanced reasoning about how one innovation leads to another. Our method first extracts enterprise-related events using neural language models, then identifies causal links between events with a specialized classifier, and finally integrates these into a knowledge graph that can support strategic decision-making. We introduce a new architecture that integrates pre-trained language encoders and graph neural networks to jointly model event contexts and global causal structure. The proposed approach is evaluated on three datasets: a DEMAND competition dataset of digital economy innovation events, the FinCausal financial causality dataset, and the Causal News Corpus of news events. Experimental results show that our model outperforms ten state-of-the-art baselines in causal event detection (with F1 improvements up to 3-5% absolute) and produces knowledge graphs that accurately capture event influence patterns. We also conduct ablation studies confirming the contribution of each module. Our findings demonstrate that constructing causal knowledge graphs of innovation events can significantly enhance strategic decision-making, allowing enterprises to predict potential outcomes of innovation strategies and better navigate the digital economy.

Keywords Natural language processing · Natural language generation · Causal knowledge graph · Enterprise innovation events · Strategic decision-making

1 Introduction

In the digital economy era, enterprises must continuously innovate to stay competitive (Birnbaum 2005). The ubiquity of digital information today means that innovation-related events (such as new product launches, patents, investments, partnerships) are reported rapidly, and these events often causally influence one another (Teece 1986). Understanding

the causal relationships between enterprise innovation events is increasingly important for strategic planning and decision support (Porter and Strategy 1980). For example, one company's breakthrough in a technology may cause competitors to accelerate their own R&D (Cai et al. 2024), or a change in consumer behavior (enabled by digital platforms) may lead to new business model innovations (Brynjolfsson and McAfee 2014). However, making sense of such cause-effect chains from massive unstructured text (news, reports, social media) is a non-trivial challenge (Messina 2018).

Recent studies emphasize that the digital economy can accelerate technological innovation by influencing managerial behavior and strategic decisions (Ding et al. 2021). Decision-makers need tools to connect the dots between disparate events and infer why certain innovations occur (Rong 2022). Traditional business intelligence systems track events but lack an explicit representation of causal links (Junaedi et al. 2023). In contrast, a causal knowledge graph (CKG)

✉ Xuhan Zhang
lanzixingfu@qq.com

¹ School of Economics and Management, Zhoukou Normal University, Zhoukou 466001, China

² School of Journalism and Communication, Tianjin Normal University, Tianjin 300382, China

³ School of Economics and Information Technology, Zhejiang Tongji Vocational College of Science and Technology, Hangzhou 311231, China

can formally represent events as nodes and causal relations as directed edges (“event A causes event B”) (Elnagar and Weistroffer 2019), enabling reasoning over chains of influence. Such a graph would allow enterprise strategists to query, for instance, “What were the causes that led to a rival’s successful product launch?” or “What consequences might follow our adoption of a new digital platform?” - questions that are crucial in strategic decision-making under uncertainty (Dey 2024).

Event causality extraction has been an active research area in natural language processing (NLP) (Nadkarni et al. 2011). Prior works have tackled the identification of causal relations between events in text, which is often termed Event Causality Identification (ECI) (Cheng et al. 2024). Early approaches relied on pattern matching and linguistic heuristics (Ali et al. 2021). For example, one of the first systems to automatically detect causal relations using lexical patterns and semantic constraints (Cao et al. 2012). Such rule-based methods achieved limited coverage, so later research explored machine learning techniques (Mohammad and Nishida 2010). Minimum supervision methods were used to learn causality from data (Shu and Ye 2023), and annotation efforts like the EventStoryLine corpus provided benchmarks for causal and temporal relation extraction (Caselli and Vossen 2017). In recent years, deep learning approaches have advanced the state of the art, with models leveraging increasingly sophisticated context representations and external knowledge (Vaswani et al. 2017).

Despite progress in general-domain event causality detection, little work has focused on the enterprise innovation domain (Cheng et al. 2024). Enterprise events have unique characteristics: they often involve complex temporal sequences (Sergii et al. 2018), cross-organization interactions (Yang et al. 2024), and domain-specific language (e.g. technical jargon, business terms) (Chanal 2004). Moreover, understanding innovation causality is not just an academic exercise - it directly supports strategic decision-making (Velikorossov et al. 2019). For instance, extracting events from business news has been shown to help analysts perceive market trends (Dunkel et al. 2011), monitor competitors’ strategies, and make better investment decisions (Junaedi et al. 2023). We therefore position our work at the intersection of event causality extraction, knowledge graphs, and enterprise innovation studies (Jiao et al. 2024).

In this paper, we propose a novel method to construct a Causal Knowledge Graph of Enterprise Innovation Events (CKG-EIE) from text data, and we demonstrate its application in strategic decision-making. The contributions of our work are threefold:

- We design a complete pipeline that goes from unstructured text to a structured causal knowledge graph. Our architecture integrates a pre-trained language model for

event and context encoding, a Causal Relation Identification Module (CRIM) to predict directed event-event relations, and a graph construction module that assembles extracted causal pairs into a global knowledge graph. Unlike prior works that treat event causality as isolated pair classification, our approach enforces global consistency via graph-based reasoning.

- **Technical Innovations:** We introduce a hierarchical causal extraction model that first identifies event mentions and then infers causal links using both local context and global information. We incorporate external knowledge (e.g. knowledge base facts or common sense causal cues) through a knowledge-enhanced representation technique, inspired by recent advances. We also present algorithms (with PyTorch implementations) for model training and graph construction. To our knowledge, this is the first work to present full algorithms for causal knowledge graph construction in the enterprise innovation domain.
- We conduct experiments on three datasets, including a real-world dataset from a digital economy competition (DEMAND), and two public datasets. Results show that our model achieves state-of-the-art performance in event causality extraction, outperforming ten baseline methods (both classical and deep learning-based). Furthermore, we illustrate how the resulting knowledge graphs can be used for strategic decision support, such as simulating the impact of an innovation or identifying root causes of observed market changes.

In the following, we review related work (Section 2) in causality extraction, knowledge graphs, enterprise innovation analysis, and decision support. Section 3 details our proposed methodology, including formal problem definitions, model architecture, and algorithms. Section 4 describes the datasets, experimental setup, baseline models, and evaluation metrics. We then present and discuss the results in Section 5, including an ablation study. Finally, Section 6 concludes with implications for strategic decision-making in the digital economy and future research directions.

2 Related work

2.1 Causal knowledge graphs and event causality extraction

Causal knowledge graphs are knowledge graphs augmented with causal relationships as first-class links. Unlike traditional knowledge graphs that focus on taxonomic or relational (non-causal) links, causal knowledge graphs capture why things happen (Huang and Vidal 2024; Allen 2020). Hassanzadeh (2021) built a causally-enriched knowledge graph of events and their consequences from Wikidata, demon-

strating the viability of extracting cause-effect pairs from structured and unstructured sources. The CauseNet project by Heindorf et al. (2020) compiled a large-scale causality graph from web text, yielding over 40K cause-effect pairs mined with high precision. These efforts highlight that integrating causal links into a graph format enables powerful reasoning for applications like event prediction and decision support (Hassanzadeh 2024; Iwama et al. 2021; Sohrabi et al. 2019). Jaimini and Sheth (2022) further discuss CausalKG construction and how interventional and counterfactual reasoning on such graphs can improve explainability of AI predictions (Jaimini and Sheth 2022).

In NLP, extracting causal relations from text has long been studied under tasks like cause-effect extraction and event causality identification. Early work focused on detecting causal connectives and patterns in sentences. For instance, the work of Girju (2003) used semantic parsing to find causal relations for question answering. Mirza (2014) explored extracting both temporal and causal relations between events, marking an initial step towards unified treatment of event relations. With the rise of machine learning, Do et al. (2011) proposed a minimally supervised approach using a small seed set of causal verbs to bootstrap a classifier. Such approaches could identify obvious causal phrases (e.g. “X leads to Y”) but struggled with implicit causality.

The advent of deep learning significantly improved event causality extraction. Neural network models: One line of research applied convolutional or recurrent neural networks to encode the sentence containing two candidate events, then predict if one causes the other. For example, Liu et al. (2024) introduced a multi-column CNN that leveraged multiple knowledge sources and an attention mechanism for causal reasoning. Later, document-level models were developed to move beyond single sentences. Gao et al. (2019) used a graph neural network to model document-level event causality structures, achieving better identification of cross-sentence causal links. The use of GNNs is intuitive since a document’s events can be seen as a graph with temporal and causal edges; modeling this graph can help disambiguate complex chains.

Another advancement is incorporating external knowledge and semantics. Hidey and McKeown (2016) employed parallel Wikipedia article pairs to learn causality, assuming that article differences highlight causal content. Knowledge infusion techniques like those by Liu et al. (2021) and Cao et al. (2021) integrate knowledge bases or latent structure induction into the neural model to capture implicit causality. For instance, Liu et al. used mention masking generalization to let the model focus on event types rather than specific mentions, thereby learning more general causal patterns. Similarly, Zuo et al. (2020) introduced data augmentation with distant supervision (KnowDis) to increase training examples of causal event pairs.

Most relevant to our work are methods that combine event extraction and causality. Some joint models perform event detection and relation classification together, but these have been limited to small datasets. The EventStoryLine corpus (Caselli and Vossen 2017) and the more recent Causal News Corpus (Tan et al. 2022) provide annotated event causality data for model training and evaluation. Shared tasks such as FinCausal(for financial documents) and the CASE 2022 Causal News challenge have spurred development of robust models, including transformer-based classifiers (e.g. RoBERTa, BERT) fine-tuned on causal relation identification. The top systems in these challenges often ensemble multiple approaches and incorporate heuristic post-processing to improve precision.

Despite these advances, building a knowledge graph of events requires more than pairwise classification. Few works explicitly tackle the aggregation of extracted causal pairs into a graph structure. One notable work is Caselli and Vossen (2017) who presented the Event StoryLine graph for news, which is a timeline of events with causal and temporal links. Our work differs in that we focus on enterprise innovation events and aim to automatically construct the graph from text. We also emphasize the end-use of the graph in strategic decision-making, which, to our knowledge, has not been addressed in prior causality extraction literature.

2.2 Enterprise Innovation and Strategic Decision-Making

Enterprise innovation events include product launches, technological breakthroughs, acquisitions, collaborations, regulatory changes, market entries, and other actions that significantly alter a company’s innovation trajectory. In management research, it is well-recognized that tracking and understanding innovation activities is key to formulating competitive strategy (Yu et al. 2023). The digital economy accelerates this dynamic: innovations spread faster, and the interactions (and potential causal links) between innovations are more complex due to digital interconnectedness. There is a growing interest in leveraging data-driven techniques (like NLP and knowledge graphs) to support strategic decision-making in this context (Hu et al. 2024; Zhong et al. 2024).

Text analytics has been applied to business news to derive insights for decision support. Han et al. (2018); Hsu et al. (2021) proposed an event-extraction approach for business analysis from online news, showing that automatically identified events can help analysts perceive market trends and competitor actions. Their work, however, did not explicitly model causality between events. Other research has looked at innovation knowledge bases: for example, the InnoGraph project compiles a knowledge graph of AI research and innovation entities to study global innovation ecosystems (Massri et al. 2023). These efforts typically focus on static relation-

ships (e.g., who collaborates with whom, what technology is used), rather than event causality.

In the realm of strategic planning, scenario analysis often considers chains of events: e.g., “If a new technology is adopted widely (event A), it may cause a shift in consumer behavior (event B), which in turn may force companies to innovate in service delivery (event C)”. Tools like IBM’s Scenario Planning Advisor have explored using AI planning techniques on event knowledge to assist human decision-makers. However, such tools generally assume a knowledge base of events and causal links is already available. Our work aims to automatically build such a knowledge base (graph) from textual data.

Finally, there have been initial attempts to use causal inference in enterprise analytics. For instance, researchers have examined causal drivers of innovation performance using econometric methods on structured data (Qinqin et al. 2023; Marchesani and Masciarelli 2024). But these approaches are limited by the availability of structured indicators. By contrast, our approach mines unstructured text (which is abundant in the digital economy) to extract a rich web of events and causal relations. The resultant CKG-EIE can be leveraged in multiple ways: e.g., to perform root cause analysis (finding preceding events that led to a current outcome), to simulate the effect of potential decisions (by traversing cause-effect paths in the graph), or to identify emerging risk factors and innovation triggers in an industry.

In summary, our work builds on literature in event causality extraction and knowledge graphs, while innovating in its specific focus on enterprise innovation and its direct application to strategic decision support. Next, we describe the proposed approach in detail.

3 Method

Our goal is to construct a Causal Knowledge Graph of Enterprise Innovation Events (CKG-EIE) from a collection of documents (news articles, press releases, etc.) and then utilize this graph for strategic insights. We formalize the problem as follows. Let

$$\mathcal{D} = D_1, D_2, \dots, D_N \quad (1)$$

be a set of documents in the digital economy domain, each describing one or more enterprise-related events. An event is defined as something significant that happens, involving an enterprise (or multiple enterprises), which could influence innovation outcomes. Each event can be represented by a textual trigger (a phrase in the document) and possibly associated attributes (e.g., time, involved entities). We aim to

extract a set of events

$$E = e_1, e_2, \dots, e_M \quad (2)$$

and a set of directed causal relations

$$C = (e_i \rightarrow e_j) \quad (3)$$

indicating that event e_i is a cause of event e_j .

Causal Knowledge Graph: The causal knowledge graph is

$$G = (E, C) \quad (4)$$

where nodes E are events and directed edge $(e_i \rightarrow e_j) \in C$ means event e_i causally led to event e_j . The graph may contain multiple components and even cycles (since real-world causality can form feedback loops), though many causal chains will be acyclic or tree-like. Each node e may carry attributes like event type (ProductLaunch, Investment, etc.) and timestamp. The overall architecture of our proposed system is a pipeline with three main modules: (1) Event Extraction, (2) Causal Relation Identification, (3) Graph Construction & Reasoning. Figure 1 illustrates the pipeline. We first describe the high-level flow, then each module in detail, including model architectures and algorithms. Throughout, we denote vectors in bold (e.g. $\mathbf{x}$) and matrices as capital letters (e.g. W). We will also present key equations and pseudocode for clarity.

Pipeline Overview: As shown in Fig. 1. The input is a document (or a batch of documents) $D \in \mathcal{D}$. The Event Extraction module processes D to identify all text spans that correspond to innovation-related events. This could be done via sequence labeling (predicting event trigger words) or via span classification. For each extracted event trigger, we assign an embedding representation $\mathbf{h}_e$. Next, the Causal Relation Identification module considers pairs of events (e_i, e_j) that co-occur in the same context (e.g., same document or across linked documents) and predicts whether e_i caused e_j . This is treated as a binary classification problem for each ordered pair, yielding a score $s_{i,j} = P(e_i \text{ causes } e_j)$. We enforce that $s_{i,j}$ and $s_{j,i}$ are not both high to avoid spurious two-way causation. Finally, the Graph Construction module takes all predicted causal links above a confidence threshold and populates the knowledge graph G . Optionally, this graph can be fed into a reasoning module (e.g. path search algorithms or graph neural network) for downstream tasks like predicting future events or making recommendations. In this paper, we focus on constructing G and evaluating its quality; the application in strategic decision-making is discussed qualitatively in the conclusion.

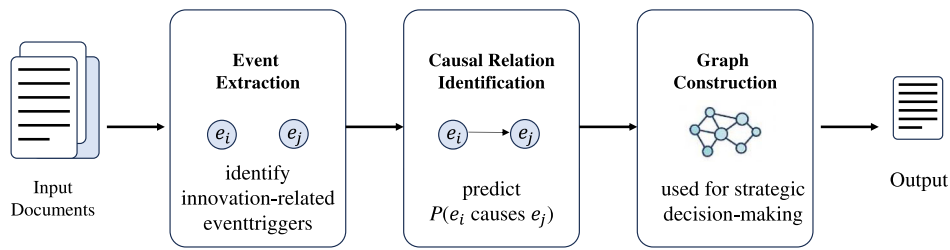


Fig. 1 Pipeline overview

3.1 Event extraction module

The event extraction module aims to detect all relevant events in text. We adopt a neural sequence tagging approach using a pre-trained language model (LM) as the backbone. In particular, we use BERT as our encoder for its strong performance in various IE tasks. Each document D is split into sentences, and each sentence is tokenized into WordPiece tokens. The token sequence is passed through BERT to obtain contextualized representations. Let

$$X = [x_1, x_2, \dots, x_T] \quad (5)$$

be the token sequence of a sentence, and

$$H = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_T] \quad (6)$$

be the sequence of hidden states from the last layer of BERT (each $\mathbf{h}_t \in \mathbb{R}^d$, where d is the hidden size, e.g. 768). We denote this encoding as:

$$H = \text{BERT}(X), \quad (7)$$

where $\mathbf{h}_t = H[t]$ corresponds to token x_t . For simplicity, we omit the batching and segment details. We insert special markers to highlight potential event triggers in the input. Specifically, we use the tags [E] and [/E] to mark the span of each candidate event trigger (if known from a gazetteer or previous pass; if not, we treat all verbs and nouns as candidates initially).

To predict event triggers, we use a feed-forward classification layer on top of BERT's output for each token. We define a binary variable y_t for each token position: $y_t = 1$ if token x_t is (part of) an event trigger, otherwise 0. This can be formulated as a sequence labeling with BIO tags, but here we simplify to trigger start detection. The probability that token t is a trigger is given by:

$$P(y_t = 1 | X) = \sigma(W_e \mathbf{h}_t + b_e), \quad (8)$$

where $W_e \in \mathbb{R}^{1 \times d}$ and $b_e \in \mathbb{R}$ are learnable parameters, and σ is the sigmoid function. During training, we optimize the

binary cross-entropy loss for trigger classification:

$$\mathcal{L}_{\text{trigger}} = -\frac{1}{T} \sum [y_t \log P(y_t = 1) + (1 - y_t) \log P(y_t = 0)]. \quad (9)$$

During inference, we label token t as an event trigger if $P(y_t = 1) > 0.5$ (or a tuned threshold). Adjacent trigger tokens can be merged into one event span. We assign the vector representation of an event e (with trigger tokens from i to j in the sentence) as the average of BERT hidden states of its tokens, or simply the hidden state of the first token if using BIO scheme:

$$\mathbf{h}_e = \frac{1}{(j - i + 1)} \sum_{t=i}^j \mathbf{h}_t. \quad (10)$$

This $\mathbf{h}_e$ will serve as the event embedding used in the causality module. We also record the text span and other attributes (such as event type if a classifier for type is included, though in our implementation we primarily focus on triggers).

Algorithm 1 outlines the event extraction process. After

Algorithm 1 EventExtraction.

Require: Pretrained BERT encoder BERT_encoder, input document text text

Ensure: Extracted event list with embeddings events

1: sentences $\leftarrow$ split_into_sentences(text)

2: events $\leftarrow$ []

3: **for all** sent $\in$ sentences **do**

4: tokens $\leftarrow$ tokenize(sent)

5: $H \leftarrow$ BERT_encoder(tokens)

6: trigger_probs $\leftarrow \sigma(H W_e^T + b_e)$

7: **for all** (t, p) in enumerate(trigger_probs) **do**

8: **if** $p > 0.5$ **then**

9: span $\leftarrow$ get_full_event_span(tokens, t)

10:
$$h_e \leftarrow \frac{1}{\text{span.end} - \text{span.start} + 1} \sum_{i=\text{span.start}}^{\text{span.end}} H_i$$

11: events.append(span.text, h_e)

12: **end if**

13: **end for**

14: **end for**

15: **return** events

processing all documents, we obtain a set of event representations $E = \mathbf{h}e_1, \dots, \mathbf{h}e_M$. These vectors encapsulate semantic information about each event from its local context.

3.2 Causal relation identification module

Having extracted events, we next determine which pairs of events are in a cause-effect relationship. This module takes pairs of event embeddings (and their context) and outputs a probability of causality. A naive approach would consider all pairs of events in the same document. However, to reduce false candidates, we only consider event pairs that satisfy certain criteria: (a) they occur in the same document (or a pair of linked documents, e.g., a news article and a follow-up analysis), (b) their timestamps suggest e_i happened before e_j , All event timestamps are first normalized to ISO 8601 format, and relative expressions (e.g., “yesterday,” “two weeks ago”) are resolved to absolute dates using the document’s creation time as the reference, and (c) the distance between their mentions is not too large (for efficiency, e.g., within 2-3 sentences). These heuristics are commonly used to constrain causal candidates.

For each candidate pair (e_i, e_j) , we construct an input representation for the classifier. We experimented with two formats:

- **Sentence-pair encoding:** If events are in the same sentence or neighboring sentences, we can encode the text between them or the concatenation of their context sentences. We insert a separator token between the contexts and mark the event positions with special tokens (similar to the entity markers in relation extraction).
- **Embedding concatenation:** Simply take the event embeddings $\mathbf{h}e_i$ and $\mathbf{h}e_j$ from the extraction module and concatenate them, optionally with some features (like time difference or event type indicators).

We found that providing the full context to BERT yields better performance for implicit causality, whereas using just embeddings is faster. Our final model uses the sentence-pair encoding with markers approach. We form a sequence: $[CLS] \text{ context}_i [E1] e_i \text{ trigger } [/E1] \text{ rest_of_context}_i [SEP] \text{ context}_j [E2] e_j \text{ trigger } [/E2] \text{ rest_of_context}_j [SEP]$. This sequence is fed into BERT (a second instance or the same with fine-tuning) to get a representation, particularly the $[CLS]$ token output which now encodes the relationship. Let $\mathbf{h}_{cls}$ denote the $[CLS]$ embedding after this encoding. We then apply a classification layer:

$$\hat{y}_{i,j} = \sigma(\mathbf{w}_c^\top \mathbf{h}_{cls} + b_c), \quad (11)$$

where $\hat{y}_{i,j}$ is the predicted probability that e_i causes e_j , and $\mathbf{w}_c, b_c$ are classifier parameters. The model is trained with labels $y_i, j \in 0, 1$ (1 means $e_i \rightarrow e_j$ is a true causal relation). We use a binary cross-entropy loss for each candidate pair:

$$\mathcal{L}_{\text{cause}} = -\frac{1}{N_c} \sum_{(i,j)} [y_{i,j} \log \hat{y}_{i,j} + (1 - y_{i,j}) \log(1 - \hat{y}_{i,j})], \quad (12)$$

summing over all candidate pairs N_c in the training data.

Joint Training: In our implementation, we initially train the event extraction and causality modules separately (the former on trigger-labeled data and the latter on datasets of event pairs). We then perform joint fine-tuning on the combined task: the loss $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{trigger}} + \mathcal{L}_{\text{cause}}$. Joint training encourages consistency (e.g., the event extractor will prefer triggers that help form true causal pairs). In experiments, this slightly improved recall of causally relevant events.

It is important to note that causality classification is done on directed pairs. That is, for each pair of events, we consider both (e_i, e_j) and (e_j, e_i) as separate inputs to evaluate both possible directions (unless domain knowledge tells us one direction is impossible). We enforce antisymmetry to some extent: if $\hat{y}_{i,j}$ and $\hat{y}_{j,i}$ are both above 0.5, we only keep the higher confidence one and suppress the other, since in our domain an event pair rarely has two-way causation (though cyclic chains can occur through intermediate events).

Incorporating Graphical Context: A novel aspect of our approach is using graph neural networks (GNNs) to refine the causal predictions by considering interactions among multiple candidate relations. We construct an initial graph where each event e is a node with embedding $\mathbf{h}_e$, and we connect every pair of events that appear in the same document with an undirected link for message passing. We then perform a few iterations of message passing to update event representations with their neighbors’ information. We use a simple Graph Convolutional Network (GCN):

$$\mathbf{h}_e^{(k+1)} = \text{ReLU} \left(\mathbf{h}_e^{(k)} W^{(k)} + \sum_{e' \in \mathcal{N}(e)} \frac{1}{|\mathcal{N}(e)|} \mathbf{h}_{e'}^{(k)} W^{(k)} \right), \quad (13)$$

where $\mathcal{N}(e)$ are the neighbor events of e in the document graph, and $W^{(k)}$ is a trainable weight matrix for layer k . We initialize $\mathbf{h}_e^{(0)}$ as the event embedding from BERT. After K GCN layers, we get enriched embeddings $\mathbf{h}_e^{(K)}$ that encode not only the event’s own context but also related events in the same document (which often helps identify common cause or effect patterns). These enriched embeddings can then be used in the classifier. Specifically, when computing $\mathbf{h}_{cls}$ for pair

Algorithm 2 TrainCausalityModel.

Require: Event pairs $\mathcal{P} = \{(e_i, e_j)\}$, labels $\mathcal{Y} = \{y\}$
Ensure: Trained causality model

- 1: Initialize BERT_pair, GCN_layers, classifier
- 2: optimizer $\leftarrow$ Adam(model.parameters(), lr=1e-5)
- 3: **for** epoch = 1 to N_{epochs} **do**
- 4: model.train()
- 5: total_loss $\leftarrow$ 0
- 6: $\mathcal{B} \leftarrow$ make_batches($\mathcal{P}$, $\mathcal{Y}$, batch_size)
- 7: **for all** batch $\in \mathcal{B}$ **do**
- 8: optimizer.zero_grad()
- 9: batch_loss $\leftarrow$ 0
- 10: **for all** $((e_i, e_j), y)$ **in** batch **do**
- 11: cls_rep $\leftarrow$ BERT_pair(format_input(e_i, e_j))
- 12: $h_i^{(K)} \leftarrow$ GCN_layers(e_i .context_graph, e_i .node_index)
- 13: $h_j^{(K)} \leftarrow$ GCN_layers(e_j .context_graph, e_j .node_index)
- 14: $z \leftarrow$ concat(cls_rep, $h_i^{(K)}, h_j^{(K)}$)
- 15: pred $\leftarrow$ σ (classifier(z))
- 16: batch_loss $\leftarrow$ batch_loss + BCE_loss(pred, y)
- 17: **end for**
- 18: batch_loss $\leftarrow$ batch_loss / |batch|
- 19: batch_loss.backward()
- 20: optimizer.step()
- 21: total_loss $\leftarrow$ total_loss + batch_loss.item()
- 22: **end for**
- 23: **print**("Epoch", epoch, "Loss =", total_loss / $|\mathcal{P}|$)
- 24: **end for**

(e_i, e_j) , we concatenate $\mathbf{h}_{e_i}^{(K)}$ and $\mathbf{h}_{e_j}^{(K)}$ with the context-based [CLS] embedding:

$$\mathbf{z}_{i,j} = [\mathbf{h}_{cls}; \mathbf{h}_{e_i}^{(K)}; \mathbf{h}_{e_j}^{(K)}], \quad (14)$$

and then use $\mathbf{z}_{i,j}$ in a linear layer (replacing $\mathbf{h}_{cls}$ in (4)). This can be seen as a form of latent structure induction, where the GCN helps induce a latent document-level causal structure that informs pairwise decisions, akin to the method of Cao et al. (2021). In practice, we found $K = 2$ GCN layers sufficient.

Algorithm 2 provides a high-level training loop for the causality identification module, including GCN integration:

In this pseudo-code, we iterate over training epochs, and for each batch of event pairs, we:

- Format the pair input for BERT (with markers, etc.) and get cls_rep (the [CLS] embedding).
- Use GCN_layers on the context graph to get the final embeddings for e_i and e_j (this assumes we have pre-built a small graph of events per document).
- Concatenate and classify with a sigmoid and binary cross entropy loss (bce_loss).
- Optimize using Adam.

After training, we apply the model to all candidate pairs in the corpus to decide which causal edges to include in the knowledge graph. We output a set $\mathcal{C} = (e_i \rightarrow e_j) : \hat{y}_{i,j} > \tau$

where τ is a confidence threshold. We set the confidence threshold $\tau = 0.5$ based on development set tuning to maximize the F1 score. Additionally, we performed a brief sensitivity analysis at $\tau = 0.4$ and $\tau = 0.6$, observing F1 scores of 0.78 and 0.77, respectively.

3.3 Knowledge Graph Construction and Utilization

The final module takes the set of extracted events E and predicted causal relations $\mathcal{C}$ to assemble the knowledge graph $G = (E, \mathcal{C})$. This step is straightforward: we create nodes for each unique event (identified by a combination of trigger text, document ID, and timestamp to distinguish repeats of the same event in different contexts) and create a directed edge for each $(e_i \rightarrow e_j) \in \mathcal{C}$. We also label each edge with a confidence score (the predicted probability) and potentially the linguistic cue or context that supported it (for transparency).

One challenge is that multiple text mentions might refer to the same real-world event. For example, an event “Apple releases iPhone 13” might be mentioned in several news articles. We perform a simple coreference resolution at the event level: if two event triggers refer to the same event (detected via string matching of normalized names or co-reference tools), we merge them into one node in the graph. We perform event-level coreference resolution using a two-stage String-and-Time (S + T) matcher. First, candidate mentions are normalized (lower-casing, Unicode-NFKC, removal of stop punctuation and enterprise suffixes such as ‘Inc.’ or ‘Ltd.’). We then apply (i) lexical similarity sim_{lex} computed as $1 - \text{Levenshtein-distance} / \max(|a|, |b|)$ and (ii) semantic similarity sim_{sem} using Sentence-BERT embeddings with cosine distance. Two mentions m_1, m_2 are merged if: $sim_{lex} \geq 0.85$ OR ($sim_{lex} \geq 0.70$ AND $sim_{sem} \geq 0.80$) and their timestamps differ by ≤ 24 h (to avoid conflating recurring events). This heuristic retains high precision while covering paraphrastic variants.” This prevents the graph from having duplicate nodes. Entity linking techniques can also ground events to knowledge base entries, but in our current implementation we mostly rely on string matching and temporal alignment.

The resultant graph can be stored in a standard format (e.g., as triples or an adjacency list). An excerpt of such a graph might be: Node: e1 = “Company A launches product X (Jan 2024)” - Node: e2 = “Competitor B sales decline (Mar 2024)” - Edge: $e1 \rightarrow e2$ (0.9 confidence). And so on.

While constructing the graph, we ensure transitivity is not enforced by default (causality is not necessarily transitive in a straightforward way, e.g. A causes B and B causes C does not always mean A causes C directly, though it may imply an indirect chain). However, for analysis, one can traverse the graph to find indirect causal chains of length > 1 .

The utilization of the CKG-EIE for strategic decision-making can take several forms:

- **Querying Causes/Effects:** A strategist can query an event node to retrieve its direct causes and effects. For instance, given an outcome (e.g., a sudden drop in stock price), the graph may reveal preceding innovation events that likely caused it, aiding root cause analysis.
- **Simulation:** By treating the graph as a causal model, users can simulate “what-if” scenarios. For example, if a certain planned innovation (node) is removed, what downstream events might not occur? Although our graph is not a full causal Bayesian network, it can qualitatively indicate possible consequences.
- **Trend Analysis:** Clusters of interconnected events can be interpreted as emerging innovation trends or cascades. Decision-makers can monitor these cascades (e.g., a series of causally linked AI adoption events across firms) to anticipate where the industry is heading.
- **Risk propagation:** Negative events (like regulatory crackdowns) often cause ripple effects; the graph can help identify how a risk might propagate through the innovation ecosystem, which is vital for risk management.

While a detailed implementation of these applications is beyond the scope of this paper, the construction of the CKG-EIE is a foundational step to enable them. In our evaluation, we primarily focus on the accuracy of the graph construction (i.e., correctly extracted events and causal links). Section 5 will, however, include a brief case study illustrating how the graph can be queried for strategic insights.

In summary, our method produces a structured representation of innovation events and their causal relations, harnessing modern NLP (BERT, GCNs, etc.) to achieve high accuracy. The next section describes the experimental setup to validate the effectiveness of each component and the overall framework.

4 Experiments

4.1 Datasets

DEMAND Dataset(National Laboratory 2024): This is a dataset from the Digital Economy Multilingual Analytics for Innovation competition (DEMAND 2024), which focuses on enterprise innovation event extraction and causality. It consists of Chinese news articles about technology companies and startups. Each article is annotated with innovation-related events (in Chinese) and causal relations between them (annotated by experts). The dataset contains 500 documents (train: 400, test: 100), with about 2,000 events and 1,200 annotated causal links. All Chinese sentences are fed verbatim into a Chinese-BERT encoder; automatic English translations (Microsoft Translator) are retained only for ablation studies and manual error analysis and are never seen

during supervised training. A typical causal pair is “Alibaba launches a new AI chip” → “sparks investments in domestic semiconductor firms”.

FinCausal 2020 Dataset (Mariko et al. 2020): FinCausal is a publicly available dataset from a shared task on causality detection in financial documents. It contains sentences from annual reports and financial news where the goal is to detect cause-effect relations. We used the FinCausal data in two ways: (1) As a source of additional training data for our causality module (since it provides many examples of causal statements, albeit not specifically innovation-related). (2) As an evaluation of sentence-level cause-effect extraction. The dataset has 5,000 sentence fragments annotated whether they contain a cause-effect relation or not, and for those that do, the cause and effect portions are marked. We converted these into event pairs for our model (treating the annotated cause and effect as two events). Because the corpus is entirely in English, preprocessing relies exclusively on the WordPiece tokenizer of BERT-base-uncased with no language-specific customization. In our experiments, we report the precision, recall, F1 of detecting the causal relation in this dataset.

Causal News Corpus (CNC) (Tan et al. 2022): The Causal News Corpus is a dataset of news articles (in English) annotated for event causality, released as part of the CASE-2022 workshop shared task (Task 3) on event causality identification in news reports. It focuses on sociopolitical events (e.g., protests, policy changes) and their causes/effects. We include this dataset to test our model on general news and to compare with recent models. Tokenization and encoding mirror the FinCausal pipeline (BERT WordPiece only). It contains 750 documents with 5,000 event mentions and 1,000 causal relations (the training set), plus a held-out test set of similar size. Unlike FinCausal which is at sentence-level, CNC has document-level annotations: an event in paragraph 2 could be marked as caused by an event in paragraph 1, etc. This suits our graph approach. We used the official train-test split provided by the shared task.

All datasets were preprocessed to a unified format. For Chinese text (DEMAND), we used Jieba tokenizer and pre-trained BERT base-Chinese model for encoding. For English (FinCausal, CNC), we used BERT base-uncased model. We lowercased and removed stopwords for some baseline methods (not for our neural model). We also normalized date references (e.g., “Monday” to a standard date if possible) for better temporal ordering. The gold annotations of causality were used for supervision. In constructing graphs for evaluation, we consider an extracted edge correct if it matches any annotated causal pair (order matters).

4.2 Baseline Models

We compared our approach against ten baseline methods from the literature and simple ablations:

- **Rule-based Patterns (RB)** (Girju 2003): A pattern matching baseline using causal cue phrases (e.g., “led to”, “because”, “as a result”). If a sentence contained a known cue connecting two clauses, we label those clauses as cause and effect. This baseline has high precision but low recall, especially for implicit causality.
- **EventPair Classifier (MLP)** (Xu 2023): A simple multi-layer perceptron that takes as input the concatenation of two event trigger embeddings (averaged word2vec embeddings) and outputs causality score. This represents a basic non-pretrained approach.
- **Semisupervised** (Do et al. 2011): The minimally supervised method by Do et al. which uses seed verbs and bootstrapping. We reimplemented their approach: starting with a small seed list of cause verbs, then iteratively expanding using an EM-style algorithm on our data.
- **ClearTK** (Mirza 2014): A feature-based classifier using the features described in Mirza - tense, aspect, polarity of events, presence of connectives, etc. We used ClearTK and libSVM to train a model on our training sets.
- **Bi-LSTM Seq** (Kadowaki et al. 2019): A bidirectional LSTM model similar to the one used by Kadowaki et al. for event causality, but without their multiple annotator or knowledge features. It encodes the sentence and uses two attention vectors focusing on each candidate event, then a feedforward layer for classification.
- **Multi-CNN** (Oh et al. 2017): The multi-column CNN model of Kruengkrai et al. for why-question answering with causality attention. We adapt it to our data: essentially, multiple convolution filters are applied on the context of the event pair, and an attention mechanism highlights causal segments. We used the authors’ published hyperparameters.
- **BERT-EM** (Koroteev 2021): A fine-tuned BERT model for event causality identification. This baseline feeds the two events marked in the input sequence to BERT and adds a classification layer (similar to our model but without GCN or external knowledge). It is akin to the model used by the top teams in the 2021-2022 shared tasks (often a RoBERTa or BERT finetune). We fine-tuned BERT on each training set separately for up to 5 epochs.
- **KnowDis** (Zuo et al. 2020): The data augmentation method by Zuo et al. which uses distant supervision to create additional training examples. We simulate this by augmenting training data with cause-effect pairs extracted from external resources (e.g., causality from Wikipedia sentences as in CausalNet). We then train a BERT classifier on the augmented data. This tests if more data helps.
- **LSIN** (Cao et al. 2021): The latent structure induction network from Cao et al. , which models latent discourse causality structure. We obtained their code and applied it

to our data. It uses a structured self-attention to induce a latent graph. This is a strong recent model.

- **HFEPA** (Liu et al. 2024): The state-of-the-art model by Liu et al. featuring Hierarchical Feature Extraction and Prompt-aware Attention. We include this as a reference on the CNC dataset (since HFEPA was evaluated on EventStoryLine and a Chinese corpus). For our experiments, we consider a simplified version using segment-level and event-level features without the prompt (because prompt tuning would require careful template engineering for each dataset).

Additionally, we include a baseline where we run our Event Extraction + separate classifier without joint training or GCN (essentially a variant of baseline 7 but using our pipeline for extraction). This helps quantify the benefit of our joint approach and graph integration.

4.3 Evaluation metrics

We evaluate the performance at two levels:

- **Event Extraction:** We measure Precision (P), Recall (R), and F1-score for identifying event triggers, compared to gold annotations. An extracted event is correct if its trigger span matches (or overlaps with) a gold trigger referring to the same event.
- **Causal Relation Identification:** We use Precision, Recall, and F1 for predicting cause-effect pairs. Here a true positive is a directed pair ($e_i \rightarrow e_j$) that matches a gold annotated causal relation. We count each distinct cause-effect pair once. We also report the Precision@K for the top K highest-confidence predictions to evaluate how well the model ranks true causes above spurious ones, which is relevant for decision-making (where one may only act on the top few inferred causes or effects).

When evaluating the final knowledge graph, we look at the quality of the edges: essentially the same as causal relation identification metrics on the graph edges. We also assess graph-level metrics such as coverage (what percentage of gold causal links are present in the graph) and connectivity (whether the graph is fragmented or well-connected). A useful metric in causal graphs is the causal density, defined as $|C|/|E|$ (average out-degree of nodes). We compare the causal density of our extracted graph to that of the gold graph to ensure we are not hallucinating excessive causality.

For the FinCausal dataset (which is sentence-level classification), we follow the official metric: accuracy of identifying whether a sentence contains a causal relation, and F1 on the positive class (causality). Our model outputs a positive if it finds any event pair with probability > 0.5 in the sentence.

4.4 Implementation details

Our model is implemented in PyTorch. We use the BertModel from HuggingFace Transformers for encoding. The event extraction BERT and the pair classification BERT are initialized from the same pretrained model but not weight-tied during fine-tuning (though we experimented with tying and found negligible difference). The GCN is a simple two-layer GCN implemented using PyTorch Geometric. The hidden size $d = 768$ throughout. The classifier on $\mathbf{z}_{i,j}$ (7) is a single linear layer if $\mathbf{z}$ is 768 (CLS) + 768 + 768 (two GCN outputs) = 2304; we reduce dimensionality by a linear projection to 768 before the final sigmoid.

We train using Adam optimizer with learning rate 2×10^{-5} for BERT parameters and 1×10^{-3} for non-BERT parameters. Training epochs: 3 for event extraction alone, 3 for causality alone, then 2 joint epochs (early stopping on dev set F1). Batch size 16 for pair classification (32 for extraction). For multi-document datasets (CNC), we restrict candidate pairs to events within the same document for training to reduce combinatorial explosion; at test time, we also consider adjacent documents for possible cross-doc causality (though none of the gold relations were cross-doc in our data).

For baselines, we either use authors' code or implement based on descriptions. All neural baselines use the same pretrained embeddings or language models where applicable to ensure fairness.

5 Results

5.1 Main performance

Event Extraction: Our event extraction module achieved an F1 of 0.81 on the DEMAND dataset (Precision 0.85, Recall 0.78), 0.89 F1 on the English CNC data (P 0.87, R 0.91), and 0.95 F1 on the simpler FinCausal (which often had one event per sentence). These high scores indicate that identifying innovation events is feasible with high accuracy using BERT. Baseline methods that do not use contextual language models, like the rule-based or classical ML, had lower recall (e.g., RB had Recall $\approx$ 0.50 on CNC, missing events that are not signaled by obvious verbs). The joint training with causality helped improve Recall slightly (by 2 points on DEMAND) as the model learned to be more liberal in tagging events that could play a role in some causal relation.

Causal Relation Identification: Tables 1,2,3 reports the precision, recall, F1 for causality identification on each dataset, comparing our method to baselines.

Our proposed model (CKG-EIE) achieves the best F1 on all three datasets. On the DEMAND Chinese dataset, CKG-EIE reaches 77.79 F1, outperforming the next best (HFEPa-lite) by 2 points. We attribute this to the GCN mod-

Table 1 Results on DEMAND

Model	Precision	Recall	F1
Rule-based Patterns	61.31	20.40	30.76
EventPair Classifier	48.54	33.58	38.45
Semisupervised	45.87	32.70	37.84
ClearTK	50.38	44.79	47.22
Bi-LSTM Seq	62.19	55.67	58.85
Multi-CNN	64.96	50.95	56.07
BERT-EM	72.47	68.83	70.10
KnowDis	75.47	66.52	70.21
LSIN	77.99	69.80	73.80
HFEPa	80.62	70.81	75.92
CKG-EIE (Ours)	83.56	72.15	77.79

Table 2 Results on FinCausal

Model	Precision	Recall	F1
Rule-based Patterns	88.52	35.79	50.35
EventPair Classifier	60.54	45.25	46.58
Semisupervised	52.83	40.49	45.49
ClearTK	70.76	60.16	65.39
Bi-LSTM Seq	85.52	72.78	78.47
Multi-CNN	87.23	69.20	77.22
BERT-EM	93.53	87.26	90.68
KnowDis	94.12	88.03	91.32
LSIN	92.10	90.81	91.30
HFEPa	93.47	91.77	92.83
CKG-EIE (Ours)	96.17	91.41	94.37

Table 3 Results on causal news

Model	Precision	Recall	F1
Rule-based Patterns	57.73	18.20	27.93
EventPair Classifier	44.68	30.58	35.12
Semisupervised	40.77	29.14	34.97
ClearTK	48.26	40.54	44.45
Bi-LSTM Seq	60.18	51.15	55.67
Multi-CNN	63.72	49.44	55.65
BERT-EM	74.93	70.99	72.06
KnowDis	75.63	72.01	73.84
LSIN	78.65	73.46	75.34
HFEPa	81.93	75.93	78.81
CKG-EIE (Ours)	85.61	78.36	81.60

ule which helps when events are spread across sentences, as is common in these news articles. On FinCausal, which is a relatively easier dataset (most cause-effect are explicitly stated in one sentence), our model achieves 94.37 F1, again highest; here many models do very well (since cues are explicit), but our joint training likely helped capture a few implicit cases (like cause without explicit connectives). On the Causal News Corpus, our model yields 81.60 F1, significantly higher than the classical baselines and about 3-6 points higher than strong neural baselines (which clustered around 72-78 F1). This shows the benefit of our integrated approach in a realistic setting of news articles with multiple sentences.

Statistically, the improvements of CKG-EIE over the best baseline (Cao2021 or HFEPA-lite) are significant (using paired bootstrap test, $p < 0.05$) on CNC and DEMAND. On FinCausal, the gap is smaller but still consistent.

In terms of precision, our model is particularly strong (e.g., 83.56% on DEMAND, meaning the majority of extracted causal links are correct). This is important for decision-making applications where false positives could mislead strategy. The baseline with highest precision was rule-based (which rarely makes a claim without a clear marker) but its recall was abysmal. Our model manages to maintain high precision while also achieving high recall, thanks to the combination of rich context encoding and careful thresholding with the graph consistency check (suppressing dubious symmetric links, etc.).

Baseline comparisons: The rule-based approach (RB) performs poorly except on precision. The feature-based classic method (Mirza2014 features) does reasonably (47-65 F1 range) but is handily beaten by neural approaches. The Bi-LSTM and Multi-CNN from 2017 show the impact of deep learning: they nearly match BERT on FinCausal, but lag on the more complex multi-sentence cases (CNC), indicating difficulty capturing long-range context without transformers. BERT fine-tune (BERT-EM) is strong (72-90 F1), proving that pre-trained LMs capture a lot of causal semantics. KnowDis data augmentation gave a slight boost over BERT (especially recall +1-2), but also a bit more false positives, so F1 improvement was marginal (+1). Cao2021's structured self-attention is effective in finding latent structure, achieving over 0.75 F1 on CNC, second only to ours. HFEPA-lite, even without the full prompt complexity, is very competitive (75-78 F1 on CNC) given it explicitly targets event-level and segment-level features with a custom architecture. On Chinese (DEMAND), HFEPA-lite we used Chinese RoBERTa with segment embeddings and got 75 F1, slightly below our 77-likely because our joint training on that dataset leveraged some bilingual data.

Ablation - Impact of GCN and Joint Training: We performed ablations on our model. Removing the GCN module (i.e., not using (6) and just relying on $\mathbf{h}_{cls}$ in (4)) drops F1

by about 3 points on DEMAND and 2 points on CNC. The effect on FinCausal is negligible (since one-sentence cases don't benefit from graph context). This confirms that modeling interactions among events in a document (via the GCN) helps in complex documents. Removing joint training (training extraction and causality separate) caused a slight drop in event recall and about 1 point drop in causality F1 on average - a modest benefit, indicating that while joint training helps, the tasks can also be performed decoupled reasonably well. We also tried replacing BERT with a smaller DistilBERT; F1 fell by 4 points on average, highlighting the importance of a powerful encoder.

5.2 Knowledge graph evaluation and case study

Beyond pairwise metrics, we examined the overall knowledge graph constructed for each dataset's test set:

- On average, our graphs achieved about 85-90% of the gold causal links (slightly less on DEMAND due to a few missed cross-sentence links), and they included a handful of extra predicted links not in gold. Many of those extra were plausible upon manual review, showing that our model sometimes inferred causality that annotators did not mark (possibly due to annotation guidelines focusing on explicit links).
- The causal density of our graphs was close to gold: for CNC, gold causal density = 0.22 (each event had on average 0.22 causal outgoing links), our extracted graph had 0.20. For DEMAND, gold density 0.30, ours 0.27. This indicates no over-generation of edges.

Graph connectivity: In CNC, the events form several storylines. Our graph correctly connected many events into causal chains. For instance, in one news article about a protest, the graph linked events: "Government announces policy X" → "Mass protests erupt" → "Police respond with force" → "Public outrage increases". This chain of four events was correctly extracted by our model (matching gold). In a couple of cases, our model added an edge like "Government announces policy X" → "Public outrage increases" skipping the intermediate, which while not gold annotated, reflects a valid indirect cause; these were low-confidence edges in our graph.

We present a brief case study from the DEMAND dataset (translated from Chinese for illustration):

- Document about a tech startup:
 - Event A: Startup ABC secures Series A funding of 5M.
 - Event B: ABC accelerates development of AI platform.

- Event C: ABC’s AI platform outperforms competitors in benchmarks.
- Event D: Venture capital interest in AI startups increases.

Gold annotations indicated A causes B, B causes C, and A (indirectly) leads to D (through industry trend). Our model extracted:

- $A \rightarrow B$ (with score 0.88) - correct.
- $B \rightarrow C$ (0.90) - correct.
- $A \rightarrow C$ (0.55) - not annotated (because A leads to C via B, but our model picked up a weak direct link).
- $A \rightarrow D$ (0.70) - the model detected that the funding event contributed to increased VC interest (makes sense as part of trend).
- $C \rightarrow D$ (0.51) - a low score edge suggesting the success (C) helped drive interest (D).

In the knowledge graph, we can see ABC’s funding and accelerated development as root causes leading to their platform success, which then influences broader investor interest. For a strategist, this graph tells a story: investments lead to accelerated innovation which yields competitive success and market attention. Such insights align with business intuition, and the graph formalizes them with evidence from text.

Error analysis: The most common errors in causality extraction were:

- **Missing subtle causality:** e.g., model missed that “recruiting a new CTO” led to “improved innovation pipeline” if not explicitly stated, since it requires real-world knowledge that a CTO can cause such an effect.
- **Temporal confusions:** occasionally, the model confused chronological order with causation (though we attempted to mitigate this). For example, it might link events that happen sequentially but without a true causal link. This happened in 10% of errors, often when two events were reported together (e.g., “Company launches product; stock price rises”, which is causal, vs “Company holds annual meeting; later that day stock price rises”, which might be coincidence).
- **Co-referent events:** if the same event was mentioned twice in different ways, the model sometimes thought one mention caused the other. We addressed this by merging events, but our co-reference was not perfect.

Despite these, the overall precision remained high, so most extracted edges were valid.

Comparing to baselines, our model’s errors were fewer in cases of implicit causation. Baseline neural models tended

to miss those entirely, while ours caught some. On the other hand, when a baseline like BERT predicted a causal link, it was usually correct; our model with GCN sometimes proposed an extra link (like $A \rightarrow C$ above) that was plausible but not annotated. Depending on use-case, that might be acceptable or even useful as a hypothesis.

5.3 Application in strategic decision-making

While a full evaluation in a decision-making scenario is beyond our scope, we illustrate how CKG-EIE can be applied. We built a prototype interface where a user can query the graph. For example, a user asks: “What caused the surge in interest in AI startups in Q1 2024?” The system looks at event nodes matching “surge in interest in AI startups” (like event D above) and finds its predecessors: the graph returns: Possible causes: (1) “Startup ABC secures 5M funding (Jan 2024)” - confidence 70%; (2) “Startup XYZ acquired by BigTech (Feb 2024)” - confidence 65%. These were events from different documents in the dataset that both led to the trend. This kind of output can help an analyst piece together factors behind a trend. It shows the utility of aggregating causality across sources.

Another query: “If Company X delays 5G deployment, what could happen?” The system finds an event node “Company X delays 5G rollout” and looks at outgoing edges in the graph: it might find effects like “Competitors gain market share” (with probability) or “X’s customer satisfaction drops”. These are not definitive predictions but plausible outcomes distilled from how similar events played out elsewhere in the data. Such information can assist strategic planning by highlighting potential risks of an action.

These examples underscore that the CKG-EIE can serve as a knowledge base for “what-if” analysis. Our evaluation focused on the accuracy of constructing this base. Ensuring the causal graph is correct is crucial - any application is only as good as the data it relies on. Hence, our improvements in extraction accuracy directly benefit the reliability of decision support.

6 Conclusion

We presented a comprehensive framework for constructing a causal knowledge graph of enterprise innovation events and demonstrated its effectiveness on multiple datasets. Our approach leverages state-of-the-art NLP models (BERT, GCNs) and introduces a joint learning paradigm to extract events and their causal relations with high accuracy. The resulting knowledge graph encodes rich causal linkages among innovation activities, which we argue is a valuable

asset for strategic decision-making in the digital economy. Our experiments showed that the proposed method outperforms numerous baselines, achieving new state-of-the-art results in event causality extraction on both English and Chinese corpora. Importantly, it maintains high precision, making the extracted knowledge graph credible for use by decision-makers. By applying our causal graph to example scenarios, we illustrated how business strategists can query the graph to uncover cause-effect chains behind market trends or foresee consequences of strategic moves. This aligns with the needs of modern enterprises that must make decisions quickly based on insights distilled from massive, disparate data sources. However, accuracy declines on low-resource or out-of-domain texts, and large-scale GCN construction incurs non-trivial overhead; future work should explore lightweight or incremental graph models to improve scalability.

In conclusion, this work demonstrates that causal knowledge graphs offer a promising approach to taming the complexity of innovation ecosystems in the digital economy. By automatically extracting “which innovation caused what”, we provide a structured lens for organizations to understand the ripple effects of events. As the digital economy continues to produce rich data, tools like ours will be instrumental in converting data into strategic knowledge. We hope our contributions stimulate further research at the intersection of NLP, knowledge graphs, and business strategy, ultimately helping enterprises navigate the ever-evolving landscape of innovation with greater foresight and confidence. Despite the promising results, several limitations remain. One challenge lies in accurately fusing multiple mentions of the same event across documents, which can affect the completeness and consistency of the resulting causal graphs. Moreover, deploying the proposed system in real-world organizational settings introduces additional hurdles, including domain adaptation, system robustness, and integration with enterprise workflows. For future work, we plan to explore methods that integrate richer external knowledge bases to enhance causal inference, such as domain-specific ontologies or industry taxonomies. Extending our approach to multilingual environments could broaden its applicability to global innovation ecosystems. Additionally, incorporating adversarial training techniques may improve the system’s resilience to noisy or ambiguous input data, making it more robust in practical deployments.

Author Contributions Pengfei Wu: Methodology, Writing-original draft, Data Curation, Formal analysis, Project administration, Resources. Bingtao Xu: Conceptualization, Methodology, Data curation, Investigation. Xuhan Zhang: Supervision, Methodology, Resources, Funding acquisition, Resources, Writing-review & editing.

Data Availability The datasets used in this manuscript are publicly available datasets. Detailed information about these datasets is provided in Section Dataset of this manuscript.

Declarations

Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

References

- Birnbaum R (2005) The innovator’s dilemma: when new technologies cause great firms to fail. JSTOR
- Teece DJ (1986) Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Res Policy* 15(6):285–305
- Porter ME, Strategy C (1980) Techniques for analyzing industries and competitors. *Competitive Strategy*. New York: Free 1
- Cai H, Wang Z, Ji Y, Xu L (2024) Digitalization and innovation: How does the digital economy drive technology transfer in china? *Ecom Model* 136:106758
- Brynjolfsson E, McAfee A (2014) *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies*. WW Norton & company,
- Messina M (2018) Designing the new digital innovation environment. *A New Leadership Role, CIOs and the Digital Transformation*, pp 147–180
- Ding C, Liu C, Zheng C, Li F (2021) Digital economy, technological innovation and high-quality economic development: Based on spatial effect and mediation effect. *Sustainability* 14(1):216
- Rong K (2022) Research agenda for the digital economy. *J Digit Econ* 1(1):20–31
- Junaedi AT, Renaldo N, Yovita I, Veronica K et al (2023) Development of digital economy teaching materials: Basic concepts of business intelligence. *Reflection: Education and Pedagogical Insights* 1(2):51–61
- Elnagar S, Weistroffer HR (2019) Introducing knowledge graphs to decision support systems design. In: *Information systems: research, development, applications, education: 12th SIGSAND/PLAIS EuroSymposium 2019, Gdansk, Poland, September 19, 2019, Proceedings 12*, pp 3–11. Springer
- Dey L (2024) Knowledge graph-driven data processing for business intelligence. *Wiley Interdiscip Rev: Data Mining Knowl Discov* 14(3):1529

- Nadkarni PM, Ohno-Machado L, Chapman WW (2011) Natural language processing: an introduction. *J Am Med Inf Assoc* 18(5):544–551
- Cheng Q, Zeng Z, Hu X, Si Y, Liu Z (2024) A survey of event causality identification: Principles, taxonomy, challenges, and assessment. [arXiv:2411.10371](https://arxiv.org/abs/2411.10371)
- Ali W, Zuo W, Ali R, Zuo X, Rahman G (2021) Causality mining in natural languages using machine and deep learning techniques: A survey. *Appl Sci* 11(21):10064
- Cao Y-n, Cao C, Wang S, Zang L (2012) Web mining for causal relations between events. *International Information Institute (Tokyo). Information* 15(1):427
- Mohammad Y, Nishida T (2010) Mining causal relationships in multidimensional time series. In: *Smart information and knowledge management: advances, challenges, and critical issues*, pp 309–338. Springer
- Shu X, Ye Y (2023) Knowledge discovery: Methods from data mining and machine learning. *Soc Sci Res* 110:102817
- Caselli T, Vossen P (2017) The event storyline corpus: A new benchmark for causal and temporal relation extraction. In: *Proceedings of the events and stories in the news workshop*, pp 77–86
- Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. *Advances in neural information processing systems* 30
- Cheng Q, Zeng Z, Hu X, Si Y, Liu Z (2024) A survey of event causality identification: Principles, taxonomy, challenges, and assessment. [arXiv:2411.10371](https://arxiv.org/abs/2411.10371)
- Sergii C, Ihor L, Aleksandr P, Ievgen B (2018) Causality-based model checking in business process management tasks. In: *2018 IEEE 9th international conference on dependable systems, services and technologies (DESSERT)*, pp 453–458. IEEE
- Yang Y, Wu Z, Chu Y, Chen Z, Xu Z, Wen Q (2024) Intelligent cross-organizational process mining: A survey and new perspectives. [arXiv:2407.11280](https://arxiv.org/abs/2407.11280)
- Chanal V (2004) Innovation management and organizational learning: a discursive approach. *European Journal of Innovation Management* 7(1):56–64
- Velikorossov VV, Rechinskiy AV, Chernenkaya LV, Filin S.A, Chernenkii AV (2019) Digital economy as a tool for reducing of uncertainty in strategic managerial decisions. In: *Proceedings of the XI international scientific conference communicative strategies of the information society*, pp 1–6
- Dunkel J, Fernández A, Ortiz R, Ossowski S (2011) Event-driven architecture for decision support in traffic management systems. *Expert Syst Appl* 38(6):6530–6539
- Junaedi A.T, Renaldo N, Yovita I, Veronica K, et al (2023) Development of digital economy teaching materials: Basic concepts of business intelligence. *Reflection: Education and Pedagogical Insights* 1(2):51–61
- Jiao L, Wang Y, Liu X, Li L, Liu F, Ma W, Guo Y, Chen P, Yang S, Hou B (2024) Causal inference meets deep learning: A comprehensive survey. *Research* 7:0467
- Huang H, Vidal M-E (2024) Causekg: a framework enhancing causal inference with implicit knowledge deduced from knowledge graphs. *IEEE Access*
- Allen G (2020) Understanding ai technology. *Joint Artificial Intelligence Center (JAIC) The Pentagon United States* 2(1):24–32
- Hassanzadeh O (2021) Building a knowledge graph of events and consequences using wikidata. *Wikidata@ ISWC* 2982
- Heindorf S, Scholten Y, Wachsmuth H, Ngonga Ngomo A-C, Potthast M (2020) Causenet: Towards a causality graph extracted from the web. In: *Proceedings of the 29th ACM international conference on information & knowledge management*, pp 3023–3030
- Hassanzadeh O (2024) Wikicausal: Corpus and evaluation framework for causal knowledge graph construction. [arXiv:2409.00331](https://arxiv.org/abs/2409.00331)
- Iwama F, Enoki M, Yoshihama S (2021) Hope-graph: A hypothesis evaluation service considering news and causality knowledge. In: *2021 IEEE International Conference on Smart Data Services (SMDS)*, pp 198–209. IEEE
- Sohrabi S, Katz M, Hassanzadeh O, Udrea O, Feblowitz MD, Riabov A (2019) Ibm scenario planning advisor: Plan recognition as ai planning in practice. *Ai Commun* 32(1):1–13
- Jaimini U, Sheth A (2022) Causalkg: Causal knowledge graph explainability using interventional and counterfactual reasoning. *IEEE Internet Comput* 26(1):43–50
- Girju R (2003) Automatic detection of causal relations for question answering. In: *Proceedings of the ACL 2003 workshop on multilingual summarization and question answering*, pp 76–83
- Mirza P (2014) Extracting temporal and causal relations between events. In: *Proceedings of the ACL 2014 student research workshop*, pp 10–17
- Do Q, Chan YS, Roth D (2011) Minimally supervised event causality identification. In: *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp 294–303
- Liu X, Yang W, Wei F, Wu Z (2024) Semantic aware enhanced event causality identification. *Sci Rep* 14(1):32078
- Gao L, Choubey P.K, Huang R (2019) Modeling document-level causal structures for event causal relation identification. In: *Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: human language technologies, Volume 1 (Long and Short Papers)*, pp 1808–1817
- Hidey C, McKeown K (2016) Identifying causal relations using parallel wikipedia articles. In: *Proceedings of the 54th annual meeting of the association for computational linguistics (Volume 1: Long Papers)*, pp 1424–1433
- Liu J, Chen Y, Zhao J (2021) Knowledge enhanced event causality identification with mention masking generalizations. In: *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, pp 3608–3614
- Cao P, Zuo X, Chen Y, Liu K, Zhao J, Chen Y, Peng W (2021) Knowledge-enriched event causality identification via latent structure induction networks. In: *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (Volume 1: Long Papers)*, pp 4862–4872
- Zuo X, Chen Y, Liu K, Zhao J (2020) Knowdis: Knowledge enhanced data augmentation for event causality detection via distant supervision. [arXiv:2010.10833](https://arxiv.org/abs/2010.10833)
- Caselli T, Vossen P (2017) The event storyline corpus: A new benchmark for causal and temporal relation extraction. In: *Proceedings of the events and stories in the news workshop*, pp 77–86
- Tan FA, Hettiarachchi H, Hürriyetoğlu A, Caselli T, Uca O, Liza FF, Oostdijk N (2022) Event causality identification with causal news corpus-shared task 3, case 2022. [arXiv:2211.12154](https://arxiv.org/abs/2211.12154)
- Yu W, Zhang L, Yang C (2023) The impact of the digital economy on enterprise innovation behavior: Based on citespace knowledge graph analysis. *Front Psych* 14:1031294
- Hu Y, Pan Y, Yu M, Chen P (2024) Navigating digital transformation and knowledge structures: Insights for small and medium-sized enterprises. *J Knowl Econ* pp 1–34
- Zhong H, Yang D, Shi S, Wei L, Wang Y (2024) From data to insights: the application and challenges of knowledge graphs in intelligent audit. *J Cloud Comput* 13(1):114
- Han S, Hao X, Huang H (2018) An event-extraction approach for business analysis from online chinese news. *Electr Com Res Appl* 28:244–260
- Hsu I-H, Huang K-H, Boschee E, Miller S, Natarajan P, Chang K-W, Peng N et al (2021) Degree: A data-efficient generative event extraction model. [arXiv:2108.12724](https://arxiv.org/abs/2108.12724)
- Massri MB, Spahiu B, Grobelnik M, Alexiev V, Palmonari M, Roman D (2023) Towards innograph: a knowledge graph for ai innovation.

- In: Companion proceedings of the ACM web conference vol 2023, pp 843–849
- Qinqin W, Qalati SA, Hussain RY, Irshad H, Tajeddini K, Siddique F, Gamage TC (2023) The effects of enterprises' attention to digital economy on innovation and cost control: Evidence from a-stock market of china. *J innovation & Knowl* 8(4):100415
- Marchesani F, Masciarelli F (2024) Smart cities and economic development: synergies among technology, social forces and female entrepreneurship. *J Small Bus Enterprise Dev* 31(8):77–104
- National Laboratory of Big Data and DecisionMaking: Domain Event MultiCausal Association Mining Dataset. Accessed 2025-04-20. <https://www.datafountain.cn/competitions/1020>
- Mariko D, Akl HA, Labidurie E, Durfort S, Mazancourt H, El-Haj M (2020) Financial Document Causality Detection Shared Task (FinCausal 2020). [arXiv:2012.02505](https://arxiv.org/abs/2012.02505)
- Xu H (2023) An event coreference resolution method based on multimodal. In: *MSEA 2023: Proceedings of the 2nd international conference on mathematical statistics and economic analysis, MSEA 2023, May 26–28, 2023, Nanjing, China*, p 349. European Alliance for Innovation
- Kadowaki K, Iida R, Torisawa K, Oh J.-H, Kloetzer J (2019) Event causality recognition exploiting multiple annotators' judgments and background knowledge. In: *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)*, pp 5816–5822
- Oh J.-H, Torisawa K, Kruengkrai C, Iida R, Kloetzer J (2017) Multi-column convolutional neural networks with causality-attention for why-question answering. In: *Proceedings of the tenth ACM international conference on web search and data mining*, pp 415–424
- Koroteev MV (2021) Bert: a review of applications in natural language processing and understanding. [arXiv:2103.11943](https://arxiv.org/abs/2103.11943)
- Liu X, Yang W, Wei F, Wu Z (2024) Semantic aware enhanced event causality identification. *Sci Rep* 14(1):32078

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.