# TCEKG: A Temporal and Causal Event Knowledge Graph for Power Distribution Network Fault Diagnosis

Feilong Liao[1], Jianye Huang[1], Qichuan Liu[2(✉)], Xinjie Peng[2], Bingqian Liu[1], Xinxin Wu[1], and Jian Qian[1]

[1] State Grid Fujian Electric Power Research Institute, Fuzhou 350007, China
[2] School of Informatics, Xiamen University, Xiamen 361005, China
{30920231154354,30920231154356}@stu.xmu.edu.cn

**Abstract.** Power distribution network fault diagnosis is important to ensure the smooth operation of the power grid. Line-level fault diagnosis techniques for distribution networks are almost non-existent, and some techniques using knowledge graphs are limited to some static entities, ignoring both more specific lines and the dynamic logic of event evolution. Currently, the Distribution Management System (DMS) is widely used in the country, and a large number of fault tickets exist in it. In this paper, we propose a practical event knowledge representation for distribution network fault diagnosis. Specifically, we design a series of abstract models to organize fault tickets and construct a **T**emporal and **C**ausal **E**vent **K**nowledge **G**raph (TCEKG), which can record temporal causal information and can be well integrated with domestic DMS. In addition, we design two TCEKG-based Fault Diagnosis Models (FDMs). To make the FDM more focused on recent events, we design a temporal decay mechanism for filtering events. Extensive experiments and ablation studies on four real-world datasets show that our TCEKG-based FDMs can effectively perform the distribution network fault diagnosis task by using TCEKG and are efficient in a single inference.

**Keywords:** Event Knowledge Graph, Power Distribution Network, Fault Diagnosis, Practical Knowledge Graph.

## 1 Introduction

With the sinking and application of Knowledge Graph (KG) in different domains, a large number of domain-specific KGs, such as financial investment quantification [3], travel destination recommendation [18], can enable more intelligent decision-making. Recently, in the field of electric power, KGs have been leveraged for equipment management, assistant decision-making, and fault diagnosis [11,20,16,19,7,21]. However, KG-based fault diagnosis methods for the distribution network are almost non-existent. Moreover, KGs can only model static knowledge but ignore events and their logical relations, such as causal relations and temporal relations. Therefore, it is difficult for KGs to provide the event knowledge for distribution network fault diagnosis scenarios.

The Event Knowledge Graph (EKG), first presented in 2016, is an upgraded version of the KG that incorporates events [14], and it is often used to solve domain-specific problems. Currently, EKG-related tasks and methods are still essentially the key tasks of KGs, which are mainly from the information extraction domain, but these tasks and methods are still far from the EKG in real-world scenarios [9]. Due to the naturally complex correlations and event-centric fault tickets of distribution network lines, it is well suited to modeling the data with EKG for fault diagnostic tasks.

To design a practical EKG applied to distribution network fault diagnosis, we propose a practical **T**emporal and **C**ausal **E**vent **K**nowledge **G**raph (TCEKG). Our TCEKG is constructed based on abstract models that are close to real fault scenarios, and two TCEKG-based methods are designed to verify the effectiveness under real scenario data. Additionally, we introduce a temporal decay mechanism to improve the inference accuracy. Overall, our main contributions can be summarized as follows:

- To the best of our knowledge, we are the first to propose a practical **T**emporal and **C**ausal **E**vent **K**nowledge **G**raph (TCEKG) for the distribution network, which uses multiple abstract models to organize the temporal and causal information of faults and well integrates with domestic DMS.
- We propose two novel TCEKG-based Fault Diagnosis Models (FDMs) for personalized fault diagnosis of the distribution network for validating the effectiveness of TCEKG. In addition, a temporal decay mechanism is designed to adjust the weight of important historical events, so that the FDM focuses more on recent events.
- Extensive experiments and ablation studies on four real-world datasets demonstrate that our proposed TCEKG can effectively utilize causal and temporal information for fault diagnosis, and its single inference time costs 2.32s, which is suitable for fault diagnosis auxiliary decision-making in real scenarios.

## 2 Related Work

Knowledge Graphs (KGs) are widely used as a form of knowledge representation for retrieval and inference of domain-specific knowledge [2]. The limitations of KGs have led to the emergence of Event Knowledge Graphs (EKGs) [9], which reflect the evolutionary, temporal logic, or causal logic of events well [12,6,8]. With the development of Natural Language Processing (NLP) and Artificial Generative Intelligence (AGI), more and more KG tasks based on supervised or in-context learning [17] have achieved outstanding performance. For example, BERT-BiLSTM-CRF [10,5] is the most commonly used in various domains for the Named Entity Recognition (NER) task [13,4].

In the last few years KGs have been widely used in many fields. In power scenarios, KGs are mostly used in equipment management [11], assistant decision-making [20,16], which are mainly about re-organizing and improving the data quality for better knowledge query and analysis. For example, Li et al. proposed a general framework for constructing a KG for health management and intelligent operation of power equipment [11]. Ye et al. constructed a KG for fault disposal of distribution networks in a more detailed way by fusing four knowledge types, including equipment topology, defects knowledge base, fault contingency plan, and scheduling protocol [20].

In addition, some studies address KG-based fault diagnosis of main equipment, but there are fewer for the distribution network. For example, Xiao et al. combined

YOLOv4 visual detection and multimodal KG to achieve autonomous warning of substation faults [19]. Dong et al. used a Gradient Boosting Tree algorithm based on a transformer fault KG to achieve safety state assessment and fault analysis [7]. Zhou et al. proposed Fault Reasoning Graph Attention Networks (FRGAN), which achieves fault assessment and analysis in substation by establishing a fault information-knowledge Joint Reasoning Space to achieve fault causes reasoning [21].

## 3 Construction of TCEKG

In a broad sense, the EKG can be defined as $\mathcal{G}=\{\mathcal{E},\mathcal{R},\mathcal{T}\}$. $\mathcal{E}=\mathcal{E}_{en}\cup\mathcal{E}_{ev}$ is the node set, where $\mathcal{E}_{en}$ is an entity set and $\mathcal{E}_{ev}$ is an event set. $\mathcal{T}\subseteq\mathcal{E}\times\mathcal{R}\times\mathcal{E}$ is a triple set $\{(h,r,t)\}$, where $h,t\in\mathcal{E}$ and $r\in\mathcal{R}$. $r$ can be an entity-entity relation $\mathcal{R}_{en-en}$ or an entity-event relation $\mathcal{R}_{en-ev}$ or an event-event relation $\mathcal{R}_{ev-ev}$, where $\mathcal{R}=\mathcal{R}_{en-en}\cup\mathcal{R}_{en-ev}\cup\mathcal{R}_{ev-ev}$. In real distribution network scenarios, a large number of fault tickets are stored in the information system. The key issue is how to organize the data, mine the implicit knowledge, and make them play an important role in fault diagnosis. In this paper, we propose a practical **T**emporal and **C**ausal **E**vent **K**nowledge **G**raph (TCEKG), which is more compatible with real-world fault diagnosis in the distribution network.

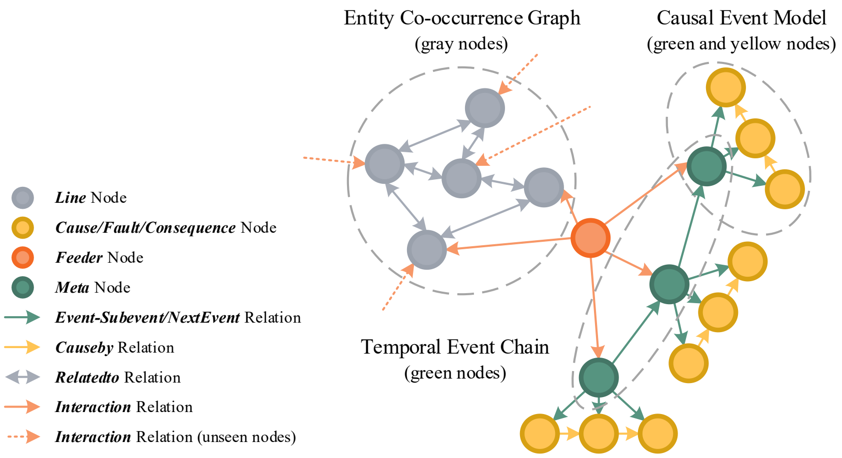


**Fig. 1.** The structure of our proposed TCEKG.

We collect and clean 32299 records from fault tickets and then design three abstract models for the TCEKG. The TCEKG is shown in **Fig. 1**, which consists of the following three abstract models: (1) The most basic abstract model is the Causal Event Model (CEM), which describes the cause and consequence of a fault event; (2) Multiple CEMs under the same feeder together form a new abstract model named Temporal Event Chain (TEC) based on temporal relation, and each TEC is connected to a special entity used for interaction; (3) Those special entities are all connected in the same Entity Co-occurrence Graph (ECG), an abstract model that enables linking TECs to each other.

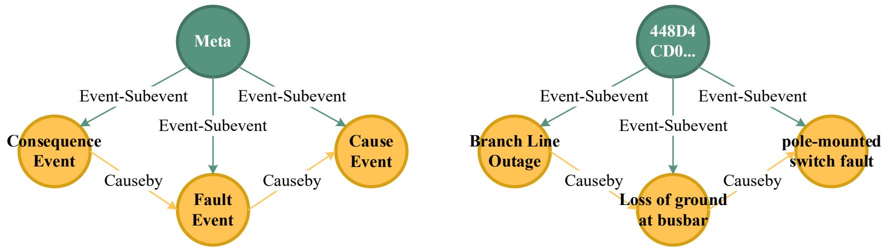


**Fig. 2.** The CEM abstract model (left) and an instantiated CEM (right).

### 3.1 Causal Event Model

Inspired by the Simple Event Model (SEM) [15], we abstract the CEM also as far as possible from the key elements, using metadata to store all information. A CEM can be defined as $\mathcal{CEM} = \{\{e_m\},\{e_{ca}\},\{e_{fa}\},\{e_{co}\},\mathcal{R}_{CEM}\}$. Each parent event $e_m$ named ***Meta*** contains three subevents, i.e. ***Cause Event*** $e_{ca}$, ***Fault Event*** $e_{fa}$ and ***Consequence Event*** $e_{co}$, where $e_m, e_{ca}, e_{fa}, e_{co} \in \mathcal{E}_{ev}$. $\mathcal{R}_{CEM}$ contains ***Event-Subevent*** relation and ***Causedby*** relation. The CEM and an instantiated CEM in **Fig. 2**. In addition, we have standardized event description in **Table 1**. We construct 32299 CEMs based on each record.

**Table 1.** The examples of standardized event description. "Coarse-grained" will only be localized to the equipment, while "fine-grained" can be localized to the equipment and its phenomena.

| Event Type | Event Example | Count |
|---|---|---|
| Cause Event (fine-grained) | Breakdown of No Gap Arrester, etc. | 328 |
| Cause Event (coarse-grained) | No Gap Arrester Fault, Switch Cabinet Fault, etc. | 36 |
| Fault Event | Tripping of Line switch, etc. | 6 |
| Consequence Event | Power outage on all main lines, etc. | 3 |

### 3.2 Temporal Event Chain

We connect CEMs from the same feeder according to the temporal relation and name it Temporal Event Chain (TEC). A TEC can show the evolution of faults on a feeder, which can be defined as $\mathcal{TEC} = \{\{e_m\},\mathcal{R}_{TEC}\}$. $\mathcal{R}_{TEC}$ includes only ***NextEvent*** relation and is used to connect parent events $e_m$ based on temporal relation. Finally, we obtain 9280 TECs from 9280 feeders. The TEC and an instantiated TEC are given in **Fig. 3**.


**Fig. 3.** The TEC abstract model (left) and an instantiated TEC (right).

### 3.3 Entity Co-occurrence Graph

Faults occur on lines in distribution network scenarios, the correlation between lines is particularly important. Real-world lines have a physical topology, which is capable of correlating different lines. However, due to the sensitivity of the topology, we use line co-occurrence as correlations and construct an Entity Co-occurrence Graph (ECG). The ECG can be defined as $\mathcal{ECG} = \{\mathcal{E}_{en}, \mathcal{R}_{en}\}$ . $\mathcal{R}_{en}$ includes only ***Relatedto*** relation.

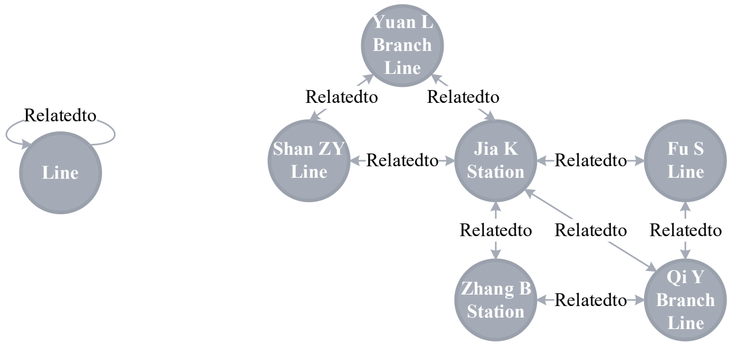


**Fig. 4.** The ECG abstract model (left) and an instantiated ECG (right).

Specifically, each record stores some natural language that contains a large number of ***Line*** entities. We divide the ECG construction into three stages. The first stage is Named Entity Recognition (NER), where we aim to extract all line entities present in natural language. The next stage is Vocabulary Construction, we will construct a standardized vocabulary of ***Line***. The final stage is ECG Construction, we assume that entities occurring in the same record are related to each other, and use the vocabulary to re-identify ***Line*** entities to construct an ECG. Finally, we get 10792 ***Line*** entities and 64966 relations. The ECG and an instantiated ECG are shown in **Fig. 4**.

**Named Entity Recognition.** NER task can help extract entities in the text. Due to the data leakage problem of black-box generative information extraction and the excellent performance of BERT-BiLSTM-CRF, we adopt it to extract the ***Line*** entities. Specifically, we manually label 1000 samples for training. The labeled entity specific to equipment-level, e.g. "*Xu D station 10kV Yao P line Kuang G branch line #03 pole KY1191 switch*", to facilitate subsequent Vocabulary Construction task.

**Vocabulary Construction.** Vocabulary Construction aims to further normalize the entities extracted in the first stage to obtain high-quality entities. We design a GFM algorithm, as shown in **Fig. 5**, to generate a high-quality vocabulary. For example, entity "*Xu D station 10kV Yao P line Kuang G branch line #03 pole KY1191 switch*" will be split into three high-quality entities "*Xu D station*", "*Yao P line*", "*Kuang G branch line*" and add them to the vocabulary. Our vocabulary including 10794 entities.

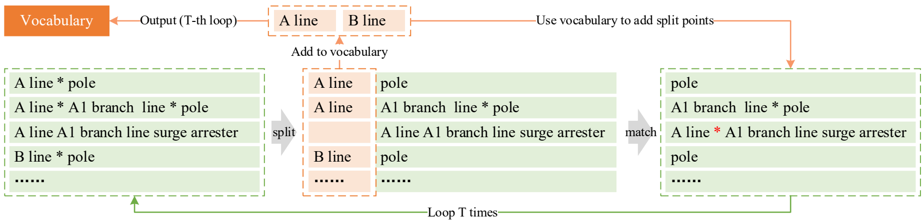


**Fig. 5.** The GFM algorithm will loop T times and each loop consists of two phases. The first is to split entities according to the first "*" and add the first half to the vocabulary; the second is to match the second half based on the vocabulary and add the "*" after the matching word.

**ECG Construction.** We verify the existence of high-quality entities in each record through the vocabulary and assume that the entities present in each record are related to each other, i.e., if there are n (n>1) entities in a record, then there are n(n-1) ***Relatedto*** relations (directed edges). Finally, our ECG contains 10972 nodes and 64966 edges.

## 3.4 Interaction between TEC and ECG

ECG creates rich associations between lines, and how to utilize these associations for mining the implicit knowledge between events becomes a key issue. Since ***Line*** entities are extracted from each record and a TEC is aggregated by a specific feeder, we use the feeder primary key as the interaction entity ***Feeder*** $e_i \in \mathcal{E}_{en}$ connecting ECG and TEC. $e_i$ associates to ECG and TEC are defined as $\{(e_i, r_i, e_{en})\}$ and $\{(e_i, r_i, \mathcal{TEC})\}$, respectively, where $e_{en} \in \mathcal{E}_{en}$ and $r_i \in \mathcal{R}$ . $r_i$ represents ***Interaction*** relation. In the end, we connected 9280 TECs and 10792 nodes in the ECG through 125092 ***Interaction*** relations. The interaction and an instantiated subgraph are shown in **Fig. 6**.

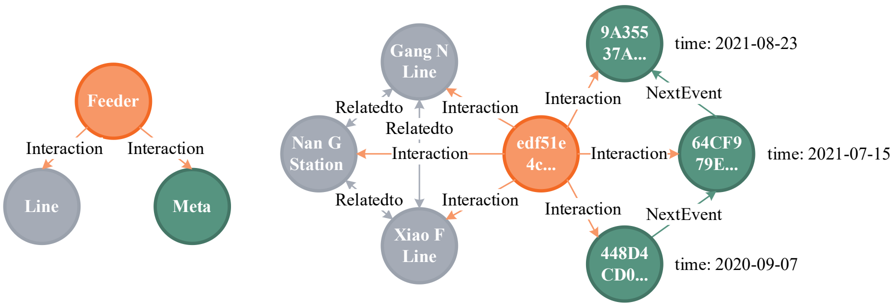


**Fig. 6.** The ***Feeder*** node connects the ***Line*** node in the ECG to the ***Meta*** node in the TEC (left). An instantiated ***Feeder*** connects three ***Line*** nodes and three ***Meta*** nodes (right).

# 4 An Application of TCEKG for Fault Diagnosis

## 4.1 TCKEG-based Fault Diagnosis Model

Current distribution network power fault diagnosis methods are based on on-site investigation, and the KG-based FDM also rarely considers line-level fault diagnosis. There are naturally complex correlations between distribution network lines, and more relevant information can be obtained in the graph structure composed of complex correlations for fault diagnosis. Therefore, we first propose TCEKG-based methods for distribution network fault diagnosis, which integrates well with domestic DMS and enables line-level fault cause prediction in distribution networks using TCEKG.

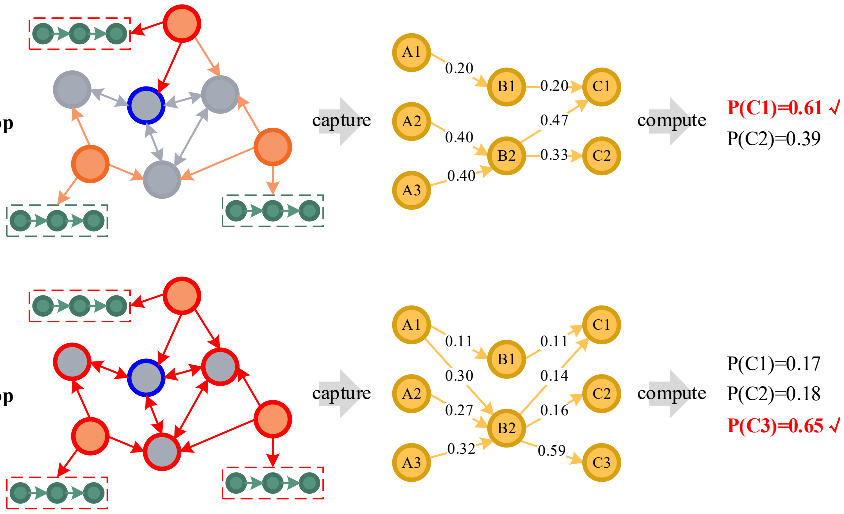


**Fig. 7.** Non-TCEKG method (top) and TCEKG-based method (bottom), where "A" is the consequence event, "B" is the fault event, and "C" is the cause event.

As shown in **Fig. 7**, the non-TCEKG method is equivalent to a 0-hop query method, and the TCEKG-based method is equivalent to a k-hop (k>0) query method. Due to the high time cost of multi-hop queries, we set k=1. Our method can capture more historical information (i.e., more CEMs), including capturing more cause event types, and more comprehensive probabilistic information to accomplish more accurate fault diagnosis. Moreover, considering the temporal characteristics between events, we introduce a temporal decay mechanism in the process of capturing information, which will act on each CEM. The temporal decay function $\tau(t)$ is as follows:

$$\tau(t) = exp\left\{-\lambda \cdot \frac{|t - t_*|}{C}\right\} \tag{1}$$

where $t$ is the time of CEM, $t_*$ is the time of test data, $|t - t_*|$ is the time gap, $C$ is the number of days in a year, $\lambda$ is hyper-parameter for the decay degree.

The TCEKG-based method includes two phases, the Capture Phase and the Compute Phase. The Capture Phase performs a multi-hop query based on TCEKG to capture multiple event records related to a particular line, and the Compute Phase performs calculations based on the captured event records and finally outputs the prediction results. Specifically, there are two models available for the Compute Phase, $\text{TCEKG}_{\text{BASE}}$ and $\text{TCEKG}_{\text{PROB}}$. $\text{TCEKG}_{\text{BASE}}$ will use all captured event records, assign a weight to each event record through a temporal decay function, and group and weight the final results directly based on the Causal Event. The $\text{TCEKG}_{\text{PROB}}$ will first calculate the conditional probability of the Consequence-Fault-Cause event chain based on the weights, and then perform a group-weighted summation to get the final result.

### 4.2 Experiments

**Datasets.** We construct TCEKG on four real-world datasets — Fault-2020, Fault-2021, Fault-2022 and Fault-full, and evaluate the effectiveness on Fault-2023.

- **Fault-2020** includes 31599 fault tickets from 12/26/2019 to 12/25/2020.
- **Fault-2021** includes 34074 fault tickets from 12/26/2020 to 12/25/2021.
- **Fault-2022** includes 31224 fault tickets from 12/26/2021 to 12/25/2022.
- **Fault-full** includes 96897 fault tickets from 12/26/2019 to 12/25/2022.
- **Fault-2023** includes 4242 fault tickets from 12/26/2022 to 06/25/2023.

**Baselines and Experimental Setting.** Due to the specificity of the TCEKG structure, some graph-based methods such as GNNs are currently difficult to apply to this task. We compare our methods with several baselines as follows:

- **K-Nearest Neighbor (KNN)** is a supervised learning algorithm based on k nearest neighbors, which determines the unseen class based on the distance between the testing and training data. We adopt k=[3, 10] and use Euclidean distance.
- **Counting Model (CM)** is a counter that filters the data by certain conditions, counts the number of classes, and outputs the sorted results.
- **Conditional Probability Model (CP)** uses the product of the conditional probabilities of the event transfers as the class scores, calculates the total score for each class, and outputs the sorted results. The event transfer satisfies the Markov assumption.
- **Multi-Layer Perceptron (MLP)** is a classical neural network that consists of the input layer, the hidden layer, and the output layer. We use a SGD optimizer with an initial learning rate of 1e-4 and a momentum of 0.9.
- **eXtreme Gradient Boosting (XGBoost)** [1] is a Gradient Boosting Decision Tree (GBDT) algorithm, which iteratively trains weak learners, and ultimately combining these weak learners into a strong learner. We use `xgboost` to implement it, and set `n_estimators=150`, `max_depth=20`, `subsample=0.6`.

**Results.** The experiment results are shown in **Table 2** (fine-grained) and **Table 3** (coarse-grained). We use hits@n (n=3,10) to evaluate the performance of the different methods. We summarize some major observations as follows: (1) From **Table 2**, our methods obtain outstanding performance on fine-grained event classification. For example, the hits@10 of our methods improve on average by +0.67% to +13.12%, +2.11% to +18.72%, +11.78% to +20.37% and +5.48% to +20.79% compared to other methods on four datasets, which demonstrates the fine-grained event classification ability of our methods. (2) From **Table 3**, our methods obtain outstanding performance on coarse-grained event classification on Fault-2022 and Fault-full. For example, the hits@10 of our methods improve on average by +2.74% to +16.05% and +5.48% to +20.92% compared to other methods, which demonstrates the coarse-grained event classification ability of our methods. In addition, the reason for the poorer performance than the non-TCEKG method on Fault-2020 and Fault-2021 is that the temporal decay punishes the data more at a greater distance from the year of the test data.

**Table 2.** Fine-grained cause event hits@3 and hits@10 on four datasets.

| Methods | Fault-2020 | | Fault-2021 | | Fault-2022 | | Fault-full | |
|---|---|---|---|---|---|---|---|---|
| | hits@3 | hits@10 | hits@3 | hits@10 | hits@3 | hits@10 | hits@3 | hits@10 |
| KNN | 7.46% | 20.72% | 7.86% | 22.00% | 7.46% | 21.17% | 7.65% | 22.66% |
| CM | 13.30% | 24.07% | 14.18% | 26.69% | 15.25% | 26.81% | 18.35% | 37.41% |
| CP | 13.00% | 25.18% | 14.18% | 27.86% | 15.17% | 27.94% | 17.49% | 37.97% |
| MLP | 11.13% | 19.70% | 10.77% | 15.79% | 10.77% | 19.35% | 10.40% | 24.91% |
| XGBoost | 10.90% | 32.14% | 11.42% | 32.40% | 9.67% | 25.51% | 10.25% | 27.04% |
| $TCEKG_{BASE}$ | **14.05%** | **32.92%** | **15.95%** | 34.19% | **19.68%** | 39.24% | 20.10% | 43.27% |
| $TCEKG_{PROB}$ | 13.58% | 32.71% | 15.06% | **34.82%** | 18.89% | **40.19%** | **20.22%** | **43.63%** |

**Table 3.** Coarse-grained cause event hits@3 and hits@10 on four datasets.

| Methods | Fault-2020 | | Fault-2021 | | Fault-2022 | | Fault-full | |
|---|---|---|---|---|---|---|---|---|
| | hits@3 | hits@10 | hits@3 | hits@10 | hits@3 | hits@10 | hits@3 | hits@10 |
| KNN | 17.62% | 41.01% | 17.05% | 42.20% | 17.67% | 42.29% | 17.17% | 41.13% |
| CM | 24.58% | 39.79% | 24.58% | 43.76% | 25.15% | 44.86% | 28.13% | 54.82% |
| CP | 24.68% | 41.44% | 24.71% | 45.56% | 24.71% | 45.82% | 27.55% | 56.27% |
| MLP | **31.80%** | 56.07% | 23.75% | 49.70% | 23.75% | 55.60% | 15.13% | 56.57% |
| XGBoost | 22.01% | **56.70%** | 22.58% | **58.48%** | 21.36% | 53.65% | 21.86% | 54.01% |
| $TCEKG_{BASE}$ | 24.95% | 51.18% | **25.95%** | 52.41% | **28.90%** | 57.78% | 28.77% | 61.78% |
| $TCEKG_{PROB}$ | 24.76% | 51.01% | 24.76% | 52.65% | 27.70% | **58.90%** | **28.80%** | **62.31%** |

**Result Analysis.** From **Table 2** and **Table 3**, we can observe that: (1) The performance rises as the training data vintage continues to approach the test data vintage, which demonstrates the temporal information is helpful for classification in our methods. For example, the hits@10 (fine-grained) of $TCEKG_{BASE}$ grows by 1.27% and 5.05% when Fault-2020 to Fault-2021 and Fault-2021 to Fault-2022, respectively. (2) The performance rises on Fault-full compared to a single year dataset, which demonstrates our methods make better use of the historical information for the classification task. For example, the hits@10 (fine-grained) of $TCEKG_{BASE}$ on the Fault-full increases 10.35%, 9.08% and 4.03% compared to Fault-2020, Fault-2021 and Fault-2022, respectively.

It is worth noting that the average time cost for a single inference of the TCEKG-based method is 2.32s, which greatly improves the efficiency of fault diagnosis compared to the on-site investigation method, and can be well applied in real scenarios.

**Ablation Study.** To verify the effectiveness of the temporal decay mechanism and multi-hop mechanism in our methods, we compared it with the following variants on four datasets: (1) $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\tau$ and $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\tau$: The variant removes temporal decay mechanism. (2) $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\varepsilon$ and $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\varepsilon$: The variant removes multi-hop mechanism. (3) $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\tau\varepsilon$ and $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\tau\varepsilon$: The variant removes temporal decay mechanism and multi-hop mechanism.

**Table 4.** Average results of ablation study on four datasets.

| Methods | Fine-grained | | Coarse-grained | |
|---|---|---|---|---|
| | hits@3 | hits@10 | hits@3 | hits@10 |
| $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\tau$ | 16.53% | 36.45% | 26.13% | 54.44% |
| $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\varepsilon$ | 15.99% | 29.93% | 26.73% | 47.77% |
| $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\tau\varepsilon$ | 14.96% | 29.51% | 24.88% | 46.86% |
| $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\tau$ | 16.21% | 36.82% | 25.65% | 54.81% |
| $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\varepsilon$ | 15.52% | 29.77% | 26.38% | 47.70% |
| $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\tau\varepsilon$ | 14.96% | 29.74% | 25.41% | 47.27% |
| $\text{TCEKG}_{\text{BASE}}$ | **17.45%** | **37.41%** | **27.14%** | **55.79%** |
| $\text{TCEKG}_{\text{PROB}}$ | **16.94%** | **37.84%** | **26.50%** | **56.23%** |

From **Table 4**, we can obtain the following observation: (1) The performance of $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\varepsilon$ and $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\varepsilon$ decrease greatly, which demonstrates the effectiveness of multi-hop mechanism. (2) The performance of $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\tau$ and $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\tau$ decrease, and compare to $\text{TCEKG}_{\text{BASE}}$-*w/o*-$\tau\varepsilon$ and $\text{TCEKG}_{\text{PROB}}$-*w/o*-$\tau\varepsilon$, removal of temporal decay mechanism further reduces performance, which demonstrates the effectiveness of temporal decay mechanism.

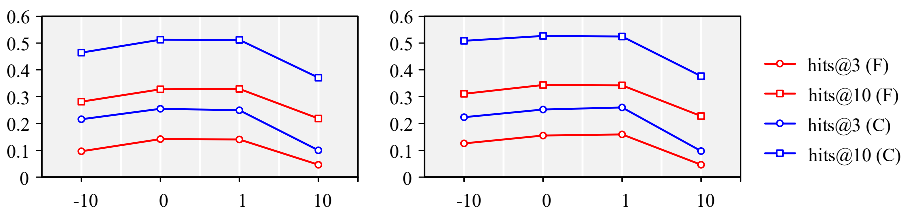


**Fig. 8.** Parameter sensitivity of $\text{TCEKG}_{\text{BASE}}$ on Fault-2020 (left) and Fault-2021 (right).

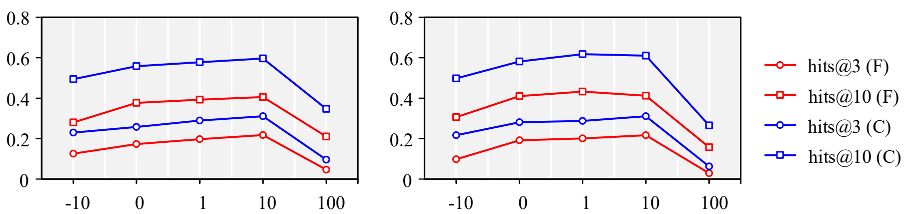


**Fig. 9.** Parameter sensitivity of $\text{TCEKG}_{\text{BASE}}$ on Fault-2022 (left) and Fault-full (right).

**Hyper-parameters Analysis.** The key hyper-parameter in our model is $\lambda$ , which is used to control the temporal decay (default $\lambda = 1$). To analyze the effects of temporal decay of TCEKG$_{\text{BASE}}$, we choose $\lambda \in \{-10, 0, 1, 10\}$ and $\lambda \in \{-10, 0, 1, 10, 100\}$ on Fault-2020/Fault-2021 and Fault-2022/Fault-full, respectively. The hits@n performances are reported in **Fig. 8** and **Fig. 9** ("F" stands for "Fine-grained" and "C" for "Coarse-grained"), and observations are summarized as follows: (1) From **Fig. 8**, the performance rises when the value of $\lambda$ approximate order of magnitude is $10^0$ on Fault-2020 and Fault-2021. When the order of magnitude of $\lambda$ is too large (e.g. $10^1$), the effect of temporal decay will be weakened, resulting in a performance decrease. The result demonstrates that the temporal decay is essential to the classification. (2) From **Fig. 9**, the performance rises when the value of $\lambda$ approximate order of magnitude is $10^0$ to $10^1$ on Fault-2022 and Fault-full. When the order of magnitude of $\lambda$ is too large (e.g. $10^2$), the temporal decay will not work, resulting in a significant performance decrease. The result also demonstrates that temporal decay is essential to the classification. In addition, the trend of hyper-parameter variation varies due to the difference in the gap between the years of the training data and the testing data, and the acceptability increases as the training and testing data become closer in years.

## 5 Conclusion

In this work, we propose a practical Temporal and Causal Event Knowledge Graph (TCEKG) for power distribution network fault diagnosis. In particular, we design a series of abstract models to construct fault event knowledge in real scenarios. Two TCEKG-based methods are specifically designed to demonstrate the effectiveness of our TCEKG. In addition, to better focus on the recent event, we introduce the temporal decay mechanism in the Fault Diagnosis Models (FDMs). Experiments and ablation studies on four real-world datasets show that our proposed TCEKG can perform distribution network fault diagnosis efficiently and is capable of efficient single inference.

**Acknowledgments.** This work is supported by the Research Funds from State Grid Fujian (Research on Key Knowledge-Data Driven Event Knowledge Graph Technologies for Intelligent Decision-Making in Distribution Networks, 521304230008).

## References

1. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. pp. 785–794 (2016)
2. Chen, X., Jia, S., Xiang, Y.: A review: Knowledge reasoning over knowledge graph. Expert systems with applications **141**, 112948 (2020)
3. Cheng, D., Yang, F., Wang, X., Zhang, Y., Zhang, L.: Knowledge graph-based event embedding framework for financial quantitative investments. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2221–2230 (2020)

4. Dai, Z., Wang, X., Ni, P., Li, Y., Li, G., Bai, X.: Named entity recognition using bert bilstm crf for chinese electronic health records. pp. 1–5. IEEE (2019)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
6. Ding, X., Li, Z., Liu, T., Liao, K.: Elg: an event logic graph. arXiv preprint arXiv:1907.08015 (2019)
7. Dong, L., Bai, L., Wu, N., Yang, D.: Research on fault prediction method of power transformer based on knowledge map. High Voltage Apparatus **58**, 151–159 (2022)
8. Gottschalk, S., Demidova, E.: Eventkg: A multilingual event-centric temporal knowledge graph. In: The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15. pp. 272–287. Springer (2018)
9. Guan, S., Cheng, X., Bai, L., Zhang, F., Li, Z., Zeng, Y., Jin, X., Guo, J.: What is event knowledge graph: A survey. IEEE Transactions on Knowledge and Data Engineering (2022)
10. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
11. Li, X., Tian, F., Guo, Y.: Key technologies for health management and intelligent operation and maintenance of power equipment in new power systems. Power System Technology **47**(09), 3710–3727 (2023)
12. Li, Z., Ding, X., Liu, T.: Constructing narrative event evolutionary graph for script event prediction. arXiv preprint arXiv:1805.05081 (2018)
13. Meng, F., Yang, S., Wang, J., Xia, L., Liu, H.: Creating knowledge graph of electric power equipment faults based on bert–bilstm–crf model. Journal of Electrical Engineering & Technology **17**(4), 2507–2516 (2022)
14. Rospocher, M., Van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T., Bogaard, T.: Building event-centric knowledge graphs from news. Journal of Web Semantics **37**, 132–151 (2016)
15. Van Hage, W.R., Malaisé, V., Segers, R., Hollink, L., Schreiber, G.: Design and use of the simple event model (sem). Journal of Web Semantics **9**(2), 128–136 (2011)
16. Wang, J., Yang, J., Pei, Y., Zhan, X., Zhou, T., Xie, P.: Distribution network fault assistant decision-making based on knowledge graph. Power System Technology **45**(06), 2101–2112 (2021)
17. Wei, X., Cui, X., Cheng, N., Wang, X., Zhang, X., Huang, S., Xie, P., Xu, J., Chen, Y., Zhang, M., et al.: Zero-shot information extraction via chatting with chatgpt. arXiv preprint arXiv:2302.10205 (2023)
18. Wu, J., Zhu, X., Zhang, C., Hu, Z.: Event-centric tourism knowledge graph—a case study of hainan. In: Knowledge Science, Engineering and Management: 13th International Conference, KSEM 2020, Hangzhou, China, August 28–30, 2020, Proceedings, Part I 13. pp. 3–15. Springer (2020)
19. Xiao, F., Wu, Y., Shen, X., He, Z., Qin, Y.: Intelligent fault diagnosis of substation equipment on the basis of deep learning and knowledge graph. Electric Power Construction **43**, 66–74 (2022)
20. Ye, X., Shang, L., Dong, X., Liu, C., Tian, Y., Fang, H.: Knowledge graph for distribution network fault handling. Power System Technology **46**(10), 3739–3749 (2022)
21. Zhou, C., Li, X., Jiao, R., Li, K.: Joint reasoning for transformer faults based on domain knowledge graphs. Proceedings of the CSEE pp. 1–12