information

MDPI

Article

TeCre: A Novel Temporal Conflict Resolution Method Based on Temporal Knowledge Graph Embedding

Jiangtao Ma [1,2], Chenyu Zhou [1], Yonggang Chen [3], Yanjun Wang [1], Guangwu Hu [4,*] and Yaqiong Qiao [5,6]

[1] College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[2] Songshan Laboratory, Zhengzhou 450000, China
[3] The State Information Center, Beijing 100045, China
[4] School of Computer Science, Shenzhen Institute of Information Technology, Shenzhen 518172, China
[5] School of Information Engineering, North China University of Water Resources and Electric Power, Zhengzhou 450046, China
[6] Henan Key Laboratory of Cyberspace Situation Awareness, Zhengzhou 450001, China
* Correspondence: hugw@sziit.edu.cn

Abstract: Since the facts in the knowledge graph (KG) cannot be updated automatically over time, some facts have temporal conflicts. To discover and eliminate the temporal conflicts in the KG, this paper proposes a novel temporal conflict resolution method based on temporal KG embedding (named TeCre). Firstly, the predicate relation and timestamp information of time series are incorporated into the entity–relation embedding representation by leveraging the temporal KG embedding (KGE) method. Then, taking into account the chronological sequence of the evolution of the entity–relation representation over time, TeCre constrains the temporal relation in the KG according to the principles of time disjoint, time precedence, and time mutually exclusive constraints. Besides that, TeCre further considers the sequence vectorization of predicate relation to discover the temporal conflict facts in the KG. Finally, to eliminate the temporal conflict facts, TeCre deletes the tail entities of the temporal conflict facts, and employs the link prediction method to complete the missing tail entities according to the output of the score function based on the entity–relation embedding. Experimental results on four public datasets show that TeCre is significantly better than the state-of-the-art temporal KG conflict resolution model. The mean reciprocal ranking (MRR) and Hits@10 of TeCre are at least 5.46% and 3.2% higher than the baseline methods, respectively.

Keywords: knowledge graph; entity–relation embedding; conflict detection; conflict resolution

check for updates

Citation: Ma, J.; Zhou, C.; Chen, Y.; Wang, Y.; Hu, G.; Qiao, Y. TeCre: A Novel Temporal Conflict Resolution Method Based on Temporal Knowledge Graph Embedding. *Information* **2023**, *14*, 155. https://doi.org/10.3390/info14030155

Academic Editor: Ryutaro Ichise

Received: 12 December 2022
Revised: 21 February 2023
Accepted: 22 February 2023
Published: 1 March 2023

1. Introduction

Knowledge graphs (KGs) are large-scale multi-relation graphs, in which nodes correspond to entities, and the types of edges represent the relations between entities. KG encodes facts in the form of triples <entity, relation, entity>, e.g., <Beijing, isCapitalOf, China>. Some of the KGs, such as DBpedia [1], NELL [2], YAGO [3] and Freebase [4], have been successfully applied to the fields of information retrieval [5], question answering systems [6], and recommendation systems [7].

In the past few years, KG embedding (KGE) has become a hot research area [8,9]. KGE methods learn representations of nodes and relations in a continuous vector space KG, while retaining graph structures and knowledge relations. However, the use of open-domain information extraction on expanding the KG usually leads to incorrect or inconsistent facts in the KG. Specifically, the fact is not always correct in the temporal KG (TKG), as they tend to be valid only in a certain time period. For example, <Donald Trump, isPresidentOf, USA, [2017, 2021]> is only correct between 2017 and 2021. Though the temporal information is available on several large KGs, such as YAGO [10], Wikidata [11], the mainstream KGE methods ignore the availability or importance of the temporal information when learning the KGE [12]. Existing KGE methods treat KG as a static graph and assume that the facts are always correct.

However, the existing temporal KGE (TKGE) method only pays attention to the factual rationality and ignores the temporal consistency. The temporal consistency models the interaction between facts and their context, so it can capture the fine-grained temporal relations, such as temporal order, temporal distance, and temporal overlap. DRF [13] incorporates the temporal information with the first-order logic Horn formula to express the constraint model, which is used to infer the temporal conflict between the consistency constraints and the queries in the resource description framework (RDF) knowledge base. The model optimization problem in the paper is defined as a scheduling task. It deals with temporal conflicts based on the approximate value of the scheduling algorithm. Since the internal operations are coarse-grained, this method can only resolve a small part of the temporal conflicts in the KG. For detecting the temporal conflicts in the uncertain time KG, a eliminate temporal conflicts (ETC) framework based on the maximum weight is proposed [14]. ETC constructs a detailed description of the constraint graph to identify the conflict detection based on time constraints. It also proposes implicit constraints and weighted conversion methods to solve temporal conflict in the KG. However, this method is not accurate as it is based on the weight conversion. These aforementioned methods are limited to a small number of time patterns, and the use of open-domain information extraction on expanding KG usually brings inaccurate pattern information. Moreover, though there are various inference rules and constraints, some of them are only suitable for specific fields, and the temporal conflicts cannot be effectively detected and resolved [15,16].

To incorporate the temporal information from the KG into the entity–relation embedding vector representations, researchers try to vectorize the temporal information. However, it is challenging due to the sparseness and irregularity of the time expression. To solve this issue, the time expression is converted into a sequence that represents the temporal information. In addition, we notice that the character-level architecture [17] used for language modeling operates on characters as atomic units to derive word embedding. Inspired by these models, our previous work Kgedl [18] detects conflicts by incorporating the temporal information into the predefined restrictions, and resolves them by evaluating confidence between entities. However, we find that Kgedl is increasingly unsuitable for large and complex KGs because of its incomprehensive restrictions and underutilization of important relational knowledge.

To solve the above problems, this paper proposes a novel temporal conflict resolution method based on temporal KG embedding (named TeCre). First, we train the LSTM network to learn the sequence embedding representations of the predicates and timestamp so that the temporal constraints, such as time disjoint, time precedence, and time mutual exclusion, can be integrated to discover the temporal conflict facts in the KG. Then, the tail entity of these conflicts is deleted, and the missing tail entity is complemented according to the score of the entity–relation embedding that is calculated through the link prediction method. In addition, we design a new loss function to guarantee the consistency between entities and the consistency between relations in the temporal space, which improves the effectiveness of TeCre in large and complex KGs. In summary, the main contributions of this paper are as follows:

(1) We propose a novel TKG-based temporal conflict detection method. The proposed method leverages the TKGE and the temporal conflict constraints to discover the temporal conflict of the facts in the KG.
(2) We propose a conflict resolution method based on the TKGE method to eliminate the conflicts. To solve the temporal conflict problems of the TKG, the proposed method deletes the conflicting temporal information from the KG and utilizes the knowledge completion method to complete the missing time information.
(3) Through a large number of experiments on four real datasets, the effectiveness of the proposed method is verified. Experimental results show that TeCre improves the MRR of the baseline method by at least 5.46% and improves at least 3.2% on Hits@10.

2. Related Work

This paper proposes to employ the TKGE method to detect the temporal conflicts of the fact in knowledge graphs and to use the link prediction method to resolve the temporal conflict issue. The related research work will be introduced from two aspects: TKGE methods and temporal conflict resolution methods.

2.1. Temporal Knowledge Graph Embedding

Recently, some works [17,19,20] focused on modeling the interaction between entity–relation and temporal information. The goal of TA-LSE [17] is to directly embed temporal information in the entity–relation. TA-TransE and TA-DistMult use recurrent neural networks to learn the time-aware representation of the relation, and use the standard scoring functions of TransE [21] and DistMult [22] to measure the distance between entities. These models can model temporal information in the form of time points with or without certain time modifiers. This method treats the timestamp as a sequence of numbers from 0 to 9, and then uses LSTM to encode the entity relation vector and the time sequence. TAE-ILP [19] found that there is a certain chronological sequence of different relations. Accordingly, a time-sensitive embedding model (TAE) is proposed to complement the KG. TAE adds temporal constraints to the embedding space, making the model temporal known and accurate. TAE captures the chronological order and other common-sense constraints that exist between certain relation types to provide more accurate link predictions. DVT [20] proposed a method of time-embedding learning using the side information of the time part of the graph. This method combines the time embedding vector with relational embedding vectors, such as concatenation, summation or dot product operation [23,24]. The translation distance scoring function [25] is adopted to measure the distance between entities, and the temporal information is encoded in the low-dimensional space of entity–relation with time embedding and time hyperplane. However, these methods cannot capture more interactions in the time dimension, such as the temporal consistency between facts and contexts, so the performance of the strategy is still limited. Besides that, HyTE [26] directly projected entities and relations into the hyperplane at a specific time and then modeled the plausibility of facts through TransE [21].

Recent studies have shown that by incorporating temporal information into the TKG, the KGE model's performance can be further improved. Know-Evolve [27] models the occurrence of facts as a point-in-time process. However, this method is based on the problem expression when dealing with concurrent events. Another method of Know–Evolve is to use the bilinear embedding learning methods to model KG elements' nonlinear time evolution. Know–Evolve deploys recurrent neural networks to capture embedded nonlinear dynamic features. However, they restrict their domain to event-based interaction types of data sets. DE-TKGC [28] integrates temporal information into diachronic entities to find and obtain the latest technological achievements on the event-based TKG. However, like TA-TransE and TA-DistMult, DE-SimplE cannot model facts involving time intervals (such as [2005, 2008]). Additionally, TEEs [29] encodes the year's representation as an entity embedding by summarizing the presentation of the entity that appears in the event-based description of the year. Usually, these methods convert time-aware facts into triples <head, relation, tail>, and then use the traditional KGE form [30,31] to measure the truth of the fact. t-TransE [32] learns time-aware embedding by learning relational ranking together with TransE. They try to impose chronological order on time-sensitive relations. t-TransE does not directly use temporal information. Different from t-TransE, we directly incorporate temporal information into the learning algorithm by TKGE [21].

2.2. Temporal Knowledge Graph Conflict Resolution

The conflict resolution of the TKG usually has two steps. The first step is to detect the conflict in the TKG, and the second step is to resolve the detected conflicts of the facts.

In terms of conflict detection, most existing outlier entity correction methods identify errors by discovering outlier entities. For instance, [33,34] identified outliers in KG through cluster mining of numeric entities. CCOD [33] employs an unsupervised method to cluster outliers. They proposed to employ the external knowledge base to distinguish the natural outliers and other outliers. The use of the knowledge base for cross-checking is able to identify the anomalies' numeric value. DIND-Dbpedia [34] employs the supervised method to identify digital outliers. This method can only identify digital outliers and cannot correct non-numerical data. MOD [35] corrects the outlier triples in the KG through clustering and classification methods. PED [36] identifies abnormal numeric entities by clustering digital RDF data, and uses probability model to learn arithmetic relations to find false links. Wrong links are viewed as abnormal triples, but in this method, triples are links between entities [37], and this method is only suitable for numeric data and dates. CN-KG [38] is committed to narrowing, identifying, and interpreting possible errors from the KG through language analysis and entity links. It uses optional source documents, provenance information, and confidence scores to evaluate the KG quality. ProbKB [39] debugs error facts by using a set of function constraint methods, which uses a set of function constraints to debug conflicts, so the method is limited to dealing with static and text facts. TeCoRe [40] is proposed for time inference and conflict resolution in uncertain time KGs. The core of TeCoRe is two state-of-the-art probabilistic reasoners, which can effectively deal with time constraints.

In terms of conflict resolution research, ETC [14] detects temporal conflicts in an uncertain time KG based on the maximum weight. It proposes to construct a detailed constraint graph to identify the time constraints of conflict detection. Implicit regulations and weight-conversion methods are proposed to resolve the temporal conflicts to ensure the time consistency of facts in the KG. Based on abnormal links in the KG, OEC [15] proposes a method named OEC to identify abnormal triples. OEC uses entity embedding methods to correct abnormal entities in the KG. In the process of EKG, each entity is projected into a shared vector space so that similar entities are close in the vector space. The embedding method of OEC is performed iteratively, and in each iteration, some vectors are far away from outlier entities. OEC corrects the KG by deleting the anomalous entity, whose embedding vectors are far from the group. MUTKG [16] reasons uncertain TKGs based on the Markov logic network and employs the Datalog restriction to detect the wrong facts in the uncertain TKGs. Then it uses maximum posterior probability reasoning to obtain the largest possible conflict-free TKG from the uncertain KG. However, this study did not consider the various temporal conflicts in the KG, and it did not consider the incompleteness of manual constraints. Since these methods do not view the semantic relation between entity–relations, they do not make full use of the timestamp information. In our previous work, Kgedl [18] detects time conflicts through three pre-defined restrictions, and evaluates the confidence between entities based on the semantic embedding and the path-based embedding, then completes the conflict resolution by replacing the entities in the conflict facts. However, the restrictions set by Kgedl for conflict detection are not comprehensive enough, and Kgedl only considers the consistency of entities but not the consistency of relations in conflict resolution. Therefore, this paper proposes to use the technique based on the KGE to detect and resolve the conflicts in the TKG.

3. Problem Statement

We define the TKG as a multi-relational directed graph with timestamped edges between any pair of nodes. In a TKG, the edge between two nodes represents an event in the real world, and the edge type represents the corresponding event type. TKG does not allow repeated edges and self-circulating edges. All edges have different time points, and each edge has other subject and object entities. Here, we utilize four-tuple $< h, r, t, [\tau_s, \tau_e] >$ to represent a fact in the knowledge graph, where τ_s and τ_e represent the effective start and end times of the triple $< h, r, t >$.

Let us take Keith Brian Alexander as an example to show his career facts. Figure 1 shows a TKG of Alexander's career life.

1. <Alexander, *retirement from*, NSA, [2014.3.28,now]>;
2. <Alexander, *work as*, IronNet Cybersecurity, [2014.5,now]>;
3. <Alexander, *work as*, 1st Commander of the USCC, [2010,2015]>;

4. <Alexander, *work as*, 16th Director of NSA, [2005,2014]>;
5. <Alexander, *work as*, Deputy Chief of Staff G-2, [2005,2014]>;
6. <Alexander, *work as*, Commanding General of the U.S. INSCOM, [2001,2003]>.

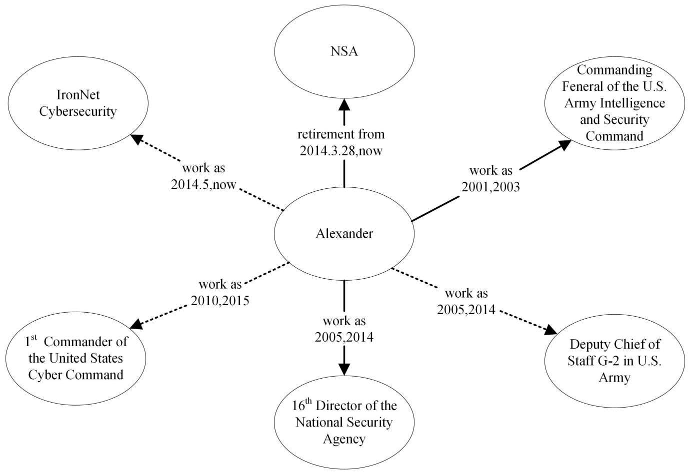

Figure 1. The TKG of Alexander's career life.

Suppose a question inquires about Alexander's job position from 2010 to 2014. In that case, we know from the above facts that fact 2 and fact 3 overlap in time and have conflict. Alexander worked for IronNet Cybersecurity after 2014 in fact 2. However, in the description of fact 3, he served as the first commander of the US Cyber Army from 2010 to 2015. This article aims to discover the facts with temporal conflicts in the TKG, that is, to find the facts corresponding to the dotted lines in Figure 1 and to resolve and correct these conflicting facts.

4. Proposed Method

This section describes the conflict resolution method based on the TKGE proposed in this paper. Figure 2 shows the overall framework of our method. First, the temporal conflicts in the KG are restricted by time sequence restriction. Then the predicate relation and time stamp information in the facts are vectorized. Next, the entity–relation vector is input to the scoring function, so the facts with temporal conflicts can be found according to the size of the scores. Finally, the conflicts with temporal conflicts are resolved using the link prediction method.

4.1. Temporal Conflict Constraint

There are many types of temporal conflict in the TKG. This paper mainly considers three types of restrictions: temporal disjoint, temporal precedence, and mutual exclusion. Here, we take four-tuples $< h_i, r_i, t_i, [\tau_{si}, \tau_{ei}] >$ and $< h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] >$ as examples to introduce the above three restrictions, where $i, j \in [1, n]$, n is the number of tuples in TKG.

Temporal Disjoint: To describe two time intervals with the same relation should not overlap, we utilize Equation (1) to describe the disjoint constraint.

$$\begin{gathered} < h_i, r_i, t_i, [\tau_{si}, \tau_{ei}] > \cap < h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] > \cap\, r_i = r_j \,\cap\, t_i \neq t_j \\ \rightarrow disjoint([\tau_{si}, \tau_{ei}], [\tau_{sj}, \tau_{ej}]) \end{gathered} \tag{1}$$

where $\cap$ represents the intersection operation, and $disjoint(,)$ represents the intersection operation. When r_i and r_j are the same relation but t_i and t_j are not the same entity, it means that the times of $[\tau_{si}, \tau_{ei}]$ and $[\tau_{sj}, \tau_{ej}]$ do not overlap. For example, an employee cannot work full time (*work as* relation) in two companies or institutions simultaneously.

Putting fact 3 <Alexander, *work as*, 1st Commander of the USCC, [2010,2015]> and fact 2 <Alexander, *work as*, IronNet Cybersecurity, [2014.5,now]> into Formula 1, we can obtain the following statement:

<Alexander, 1st Commander of the USCC, [2010,2015]> ∩ <Alexander, IronNet Cybersecurity, [2014.5,now]> ∩ $work\ as = work\ as$ ∩ IronNet Cybersecurity ≠ 1st Commander of the USCC → $disjoint([2010, 2015], [2014.5, now])$.

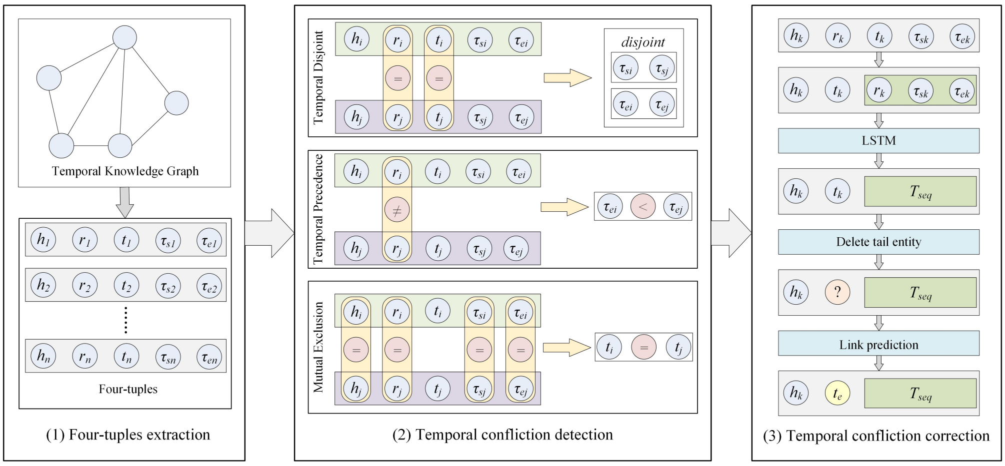

Figure 2. The proposed time conflict resolution framework, containing three parts: (**1**) four-tuples extraction, (**2**) temporal conflict detection, and (**3**) temporal conflict correction. In the four-tuples extraction, n tuples are extracted from TKG, where n is the number of tuples in TKG. In the temporal conflict detection, each tuple $< h_i, r_i, t_i, [\tau_{si}, \tau_{ei}] >$ is compared with the other tuple $< h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] >$ by three types of restrictions until all temporal conflict tuples are found, where $i, j \in [1, n]$, n is the number of tuples in TKG. In the temporal conflict correction, temporal conflict tuple $< h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] >$ is converted to $< h_j, t_j, T_{seq} >$ by LSTM, and then its tail entity is replaced by the entity through the link prediction method to obtain the correct tuple.

Here, for the same relation *work as*, Alexander worked at IronNet Cybersecurity after May 2014. Alexander worked as 1st Commander of the USCC in [2010,2015] has a temporal conflict, that is, $\tau_{sj} = 2014.5$ is smaller than $\tau_{ei} = 2015$, so there are temporal conflicts in the same kind of relation, and this fact violates the temporal disjoint restriction.

Temporal Precedence: Under the same head entity and different relations, the end time of one relation must be earlier than the start time of the other. Here, we assume that the end time of r_i is earlier than the start time of r_j, the definition of temporal precedence is as Equation (2):

$$< h_i, r_i, t_i, [\tau_{si}, \tau_{ei}] > \cap < h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] > \cap\, r_i \neq r_j \rightarrow \tau_{sj} < \tau_{ei} \tag{2}$$

In the different relations r_i and r_j, the start time of r_j is later than the end time of r_i. For example, in fact 1 <Alexander, *retirement from*, NSA, [2014.3.28,now]> and fact 3 <Alexander, *work as*, 1st Commander of the USCC, [2010,2015]>, Alexander's start time in fact 1 is later than the end time in fact 3, so these two facts violate different relations temporal precedence constraint.

Mutual Exclusion: If two facts have the same relation, the same head entity and the same timestamp information, then they must have the same tail entity, i.e., Equation (3):

$$\begin{gathered} < h_i, r_i, t_i, [\tau_{si}, \tau_{ei}] > \cap\ < h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] >\ \cap\, h_i = h_j \,\cap \\ r_i = r_j \,\cap \tau_{si} = \tau_{sj}\ \cap\ \tau_{ei} = \tau_{ej} \rightarrow t_i = t_j \end{gathered} \tag{3}$$

Some facts conflict with each other because they violate the mutual exclusion restriction. For instance, fact 4 <Alexander, work as, 16th Director of NSA, [2005,2014]> is conflicted with fact 5 <Alexander, work as, Deputy Chief of Staff G-2, [2005,2014]>, because Alexander can only work for one team in the same time period.

4.2. Temporal Sequence Vectoring

The fact of a given temporal KG is represented by a four-tuple $< h, r, t, [\tau_s, \tau_e] >$, TeCre decomposes the given (possibly incomplete) timestamp $[\tau_s, \tau_e]$ into a sequence. Inspired by TA-LSE [17], TeCre splits the numbers in year, month, and day to form a temporal. For each quadruple, TeCre can extract a sequence of predicate tokens, which are marked by the relation type and time modifier token composition, such as "since" or "until". TeCre calls the concatenation of the predicate token sequence and the time token sequence the predicate sequence T_{seq}. Therefore, the time KG can be expressed as a set of four-tuples of the form $< h, r, t, T_{seq} >$, where the predicate sequence can contain temporal information. Table 1 lists some examples of such facts from temporal KG and their corresponding predicate order. We use the suffixes y, m, and d to indicate whether the number corresponds to the year, month or day information. These marker sequences are used as input to LSTM.

Table 1. Facts.

Fact	Predicate and Timestamp Sequence	Head Entity	Tail Entity
<Alexander, *retirement from*, NSA, [2014.3.28,now]>	[*retirement from*,2y,0y,1y,4y,03m,2d,8d]	Alexander	NSA
<Alexander, *work as*, IronNet Cybersecurity, [2014.5,now]>	[*work as*,2y,0y,1y,4y,now]	Alexander	IronNet Cybersecurity
<Alexander, *work as*, 1st Commander of the USCC, [2010,2015]>	[*work as*, 2y,0y,1y,4y, 2y,0y,1y,5y]	Alexander	1st Commander of the USCC

First, each tag of the input sequence T_{seq} is mapped to its corresponding d-dimensional embedding through the linear layer, and the resulting sequence of embedding is used as the input of LSTM. Then, LSTM outputs the predicate timestamp sequence representation with temporal information $e_{T_{seq}}$. In addition, the last hidden state of LSTM represents each predicate timestamp sequence of length N [41].

Secondly, $e_{T_{seq}}$ combines with the subject and object embedding in the standard scoring function. According to the embedding vector representation of the entity–relation in the KG, the triple $y =< h, r, t, [\tau_s, \tau_e] >$ corresponding to the quadruple $< h, r, t, T_{seq} >$ has the following scoring function:

$$f(y) = \|e_h + e_{T_{seq}} - e_t\|_2 \tag{4}$$

where e_h and e_t are the subject and object embedding of the triple, which are trained by TransE [21]. $\| \cdot \|_2$ represents the L_2-norm of the matrix.

Finally, by comparing the scoring functions, the lower the score, the greater the probability that the entity–relation pair is correct. Next, TeCre utilizes the scoring function to correct the fact that there is a temporal conflict.

4.3. Error Time Correction

For the temporal conflicts facts, they can be represented as quadruples, TeCre first deletes the tail entities of these facts. Then complete the missing tail entities by using Equation (4). The link prediction method is used to link the missing tuple and the timestamp candidate set, and then the tail entities of the missing tuple are completed.

Here, TeCre regards the TKG timestamp completion as an optimization problem based on normalization. Given a training positive example quadruple $< h_k, r_k, t_k, [\tau_{sk}, \tau_{ek}] >\in \triangle$, $\triangle$ is the fact tuple set. TeCre finds a time-related four-tuple of the same head entity $< h_l, r_l, t_l, [\tau_{sl}, \tau_{el}] >\in \triangle$, and the temporal–relation pair $< r_k, r_l >$, where $k, l \in [1, n]$. If $\tau_{sk} < \tau_{ek}$, TeCre obtains a positive temporal–relation pair $x^+ =< r_k, r_l >$, and the corresponding negative relation pair $x^- =< r_k, r_l >^{-1} =< r_l, r_k >$. The optimization goal of TeCre is for the positive temporal–relation pair's score to be lower than the score of the negative relation pair. Therefore, TeCre defines the time-series relation scoring function as shown below:

$$s(x) = \|r_k \cdot T^k_{seq} - r_l \cdot T^l_{seq}\|_2 \tag{5}$$

If the sequence–relation pair is arranged in chronological order, the score is lower, otherwise the score is higher. Note that both T^k_{seq} and T^l_{seq} are asymmetric, and the loss function is also asymmetric to obtain the temporal information.

To make the embedding space compatible with the observed tuples, TeCre uses the fact tuple set $\triangle$ and follows the strategy adopted by the previous method. Then, TeCre minimizes the loss function through Equation (6):

$$L = \sum_{y^+ \in \triangle} \sum_{y^- \in \triangle'} [\gamma_1 + f(y^+) - f(y^-)] + \lambda \sum_{x^+ \in \Omega, x^- \in \Omega'} [\gamma_2 + s(x^+) - s(x^-)] \tag{6}$$

where L is the score calculated by Equation (6), $f(\cdot)$ is the scoring function defined in Equation (4), y^+ is a tuple of positive examples, and y^- is the corresponding tuple of negative examples after replacing the tail entity. λ is the hyperparameter, and γ_1 and γ_2 are the margins. Ω is the positive relation pair, and Ω' is the negative relation pair corresponding to the reverse relation pair.

The relation r_k and r_l have the same head entity. The first term of Equation (6) guarantees the consistency between entities in the temporal space, and the second term guarantees the consistency between relations in the temporal space. The hyperparameter λ balances these two items. In this article, TeCre utilizes the stochastic gradient descent method to minimize the problem. Through continuous training, TeCre makes the score of the positive four-tuples higher than that of all negative four-tuples by using Equation (6).

4.4. Proposed Algorithm

As shown in Algorithm 1, TeCre proposes to resolve the temporal conflict of the facts in the TKG in four stages. Firstly, the facts of the predicate and time are serialized to obtain the vector (see line 1–6 in Algorithm 1). Secondly, the serialized predicate and time are used as the input of LSTM to calculate the score of the time-serialized triples (see line 7–9 in Algorithm 1). Then, TeCre calculates scores of the entities with the same head entities temporal pair but different tail entities (see line 10–12 in Algorithm 1). Finally, TeCre calculates the loss score of the facts that have temporal conflict with the loss function. Resolve the temporal conflict by deleting the tail entities of the facts and complete tail entities through link prediction in the KG (see line 13–14 in Algorithm 1).

Algorithm 1 Temporal conflict resolution algorithm based on TKGE

Input: A set of facts $< h, r, t, [\tau_s, \tau_e] >$ with conflicts in KG G
Output: KG G_1 without temporal conflict facts
1: Initialize facts $y_i = < h_i, r_i, t_i, [\tau_{si}, \tau_{ei}] > \in G, G_2 = \emptyset$
2: **for all** facts $y_j = < h_j, r_j, t_j, [\tau_{sj}, \tau_{ej}] > \in G, y_i \neq y_j$ **do**
3: $y_i \cap y_j \cap r_i = r_j \cap t_i \neq t_j \rightarrow disjoint([\tau_{si}, \tau_{ei}], [\tau_{sj}, \tau_{ej}])$
4: $y_i \cap y_j \cap r_i \neq r_j \rightarrow \tau_{si} < \tau_{ej}$
5: $y_i \cap y_j \cap h_i = h_j \cap r_i = r_j \cap \tau_{si} = \tau_{sj} \cap \tau_{ei} = \tau_{ej} \rightarrow t_i = t_j$
6: Put temporal conflicts fact y_k into G_2
7: **for all** facts $y = < h, r, t, [\tau_s, \tau_e] > \in G_2$ **do**
8: $y = < h, r, t, [\tau_s, \tau_e] > \rightarrow y = < h, t, T_{seq} >$
9: $f(y) = \| e_h + e_{T_{seq}} - e_t \|_2$
10: Put corrected facts into G_2
11: **for all** $y^+ \in \triangle, y^- \in \triangle', x^+ \in \Omega, x^- \in \Omega'$ **do**
12: $s(x) = \| r_k \cdot T^k_{seq} - r_l \cdot T^l_{seq} \|_2$
13: $L = \sum_{y^+ \in \triangle} \sum_{y^- \in \triangle'} [\gamma_1 + f(y^+) - f(y^-)] + \lambda \sum_{x^+ \in \Omega, x^- \in \Omega'} [\gamma_2 + s(x^+) - s(x^-)]$
14: **return** G_1

5. Experimental Results and Analysis

This paper compares the performance of TeCre with the baseline methods in the aspects of temporal conflict detection and temporal conflict resolution of the TKG on four public datasets.

5.1. Datasets

The statistics of the aforementioned datasets are listed in Table 2. YAGO15K [17] is sourced from FB15K [21]. The entities from FB15K to YAGO are aligned with the SAMEAS relation contained in YAGO dump, and all facts related to these entities are retained. Baseline methods use the temporal information in the "yagoDateFacts" dump to expand the fact collection. YAGO15K includes the "occursSince" and "occursUntil" time modifiers. All facts keep the temporal information at the same level of granularity as the original dump from which these datasets come. The Integrated Crisis Early Warning System (ICEWS) is a database containing political events with specific timestamps. These political events associate entities with other entities through logical predicates. The database includes events that occurred each year from 1995 to 2015. ICEWS 2014 [17] contains all events in 2014, and ICEWS 2005–2015 [17] contains all events that occurred between 2005 and 2015. WIKIDATA is a KG that can be edited by both humans and machines. The facts in the WIKIDATA [17] dataset are framed by time intervals (that is, they contain the time modifiers "occursSince" and "occursUntil"). The fact annotated with a single point in time is associated with that point in time as the start time and end time.

Table 2. Statistics of the datasets.

Dataset	YAGO15K	ICEWS14	ICEWS05-15	WIKIDATA
Entities	15403	6869	10094	11134
Relations	34	230	251	95
Facts	138056	96730	461329	150079
Time Span	1513–2017	2014	2005–2015	25–2020

For the experiments with TeCre, this paper searches the learning rate δ for Adam among $\{0.0001, 0.001, 0.01, 0.1\}$, the embeddings of entities, relation d ranges from 1 to 200, the hyperparameter λ is set as $[0, 1]$, the margins γ_1 and γ_2 are selected from $[0, 1]$. Through grid search for the area under the precision and recall curve of the verification set, the best configurations are as follows: $\delta = 0.001, d = 128, \lambda = 0.43, \gamma_1 = 0.12, \gamma_2 = 0.36$ on YAGO15K; $\delta = 0.01, d = 64, \lambda = 0.68, \gamma_1 = 0.21, \gamma_2 = 0.57$ on ICEWS14; $\delta = 0.001, d = 128, \lambda = 0.13, \gamma_1 = 0.34, \gamma_2 = 0.25$ on ICEWS05-15; $\delta = 0.001, d = 128$, $\lambda = 0.85, \gamma_1 = 0.75, \gamma_2 = 0.83$ on WIKIDATA. We train at most 1000 epochs for all datasets.

5.2. Link Prediction Settings

Following prior work RE-NET [40], CyGNet [41], and CEN [21], this paper employs a link prediction method to eliminate temporal conflicts. Therefore, the temporal conflict resolution model is evaluated by testing the model's performance on the link prediction task on TKG. This link prediction task is to use missing entities to complete a time-related fact. In this paper, for a positive four-tuple $< h, r, t, [\tau_s, \tau_e] >$ in the KG, TeCre generates the negative four-tuple $< h, r, t', [\tau_s, \tau_e] >$ by replacing the tail entity t with all possible entities. In the training phase, TeCre uses Equation (6) to make the score of the positive four-tuple higher than that of all negative four-tuples. In the testing phase, TeCre uses Equation (4) to sort the scores of all four-tuples, then gives one best result to output. Two evaluation metrics, mean reciprocal ranking (MRR) and Hits@k, are adopted to evaluate the performance of the proposed temporal conflict resolution model. MRR is the mean of the reciprocal of all calculated rankings Hits@k is the proportion of positive four-tuples in the top k four-tuples. In addition, we also employ precision (P) and recall(R) to evaluate the performance of TeCre. P represents the ratio of the number of the correct resolved temporal conflict four-tuples to the number of the resolved conflict four-tuples, and R represents the ratio of the number of the resolved conflict four-tuples to the number of all ground truth temporal conflict four-tuples.

5.3. Baseline Methods

This paper compares TeCre with three state-of-the-art TKG conflict resolution methods. OEC [15] uses entity embedding methods to correct the abnormal entity–relation in the KG. ETC [14] uses implicit constraints and weight conversion methods to resolve the temporal conflicts in the KG to ensure the time consistency of facts in the KG. MUTKG [16] reasons the uncertain TKG based on the Markov logic network method, and used the maximum posterior probability reasoning to obtain the maximum possible conflict-free TKG from the uncertain KG. RE-NET [42] uses a recurrent event encoder and a neighborhood aggregator to model past events and some events in the same timestamp, respectively. CyGNet [43] proposes a copy-generation model that learns historical facts from TKG to predict future facts. CEN [44] uses a length-aware convolutional neural network to model the fact that sequences vary in length. Our previous work Kgedl [18] detected time conflicts through three pre-defined restrictions, and resolved conflicts by modeling the confidence between entities.

5.4. Experimental Results

Figure 3 shows the comparison results of the TeCre with baseline methods on the YAGO15K. TeCre is obviously superior to the baseline methods. When the recall is 10%, the precision of TeCre, CyGNet, RE-NET, Kgedl, ETC, MUTKG, CEN, and OEC are 0.930, 0.902, 0.884, 0.869, 0.851, 0.830, 0.697, and 0.628, respectively. When the recall's value is 50%, the precision of TeCre, CyGNet, RE-NET, Kgedl, ETC, MUTKG, CEN, and OEC are 0.731, 0.726, 0.706, 0.676, 0.657, 0.550, 0.545, and 0.502, respectively. When the recall rate is 90%, the precision of TeCre, CyGNet, RE-NET, Kgedl, ETC, MUTKG, CEN, and OEC are 0.253, 0.217, 0.202, 0.187, 0.178, 0.169, 0.163, and 0.103, respectively. TeCre is 0.4∼30.2% higher than the baselines.

Figure 4 shows the comparison result of the TeCre and baseline methods on the ICEWS'14 dataset. TeCre is significantly better than the baseline methods. The area under the precision and recall curves of the TeCre is 0.697, the areas under the precision and recall curves of the CyGNet, RE-NET, Kgedl, ETC, MUTKG, CEN, and OEC are 0.647, 0.656, 0.543, 0.532, 0.385, 0.512, and 0.455, respectively. TeCre is 6.25% higher than the second-place RE-NET.

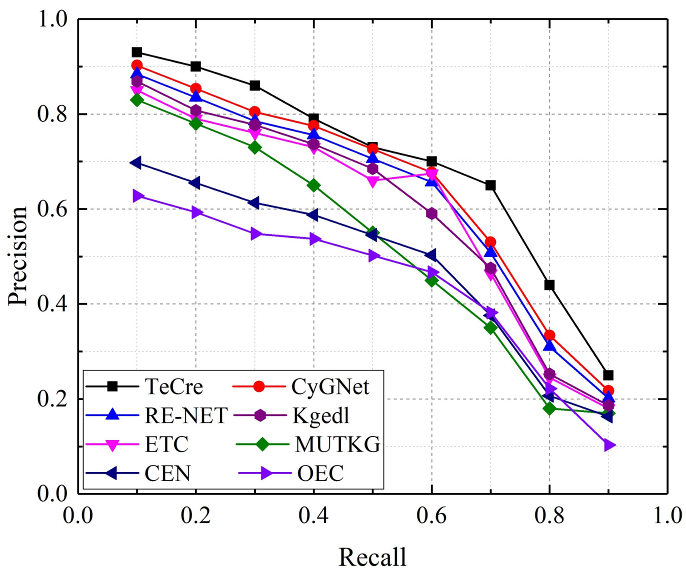

Figure 3. The recall accuracy curve of TeCre and the baseline method on the YAGO15K dataset.

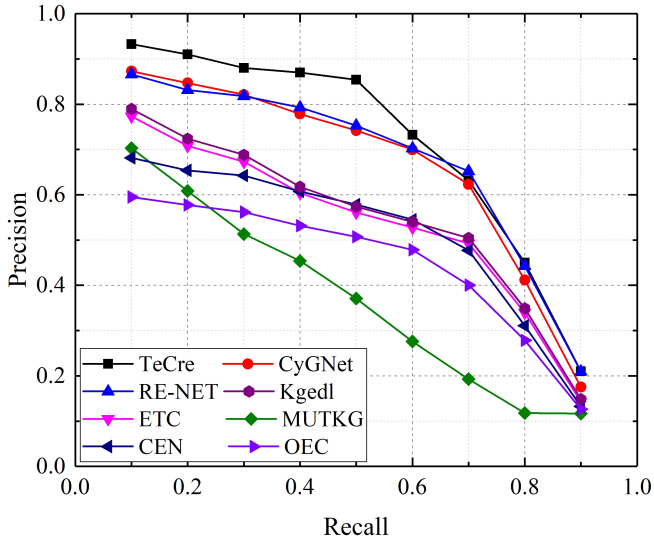

Figure 4. Comparison results on the ICEWS'14 dataset.

Figures 5 and 6 show the temporal conflict fact detection results of baseline methods on ICEWS05-15 and WIKIDATA, respectively. TeCre achieved the best results on both datasets. On the ICEWS05-15 dataset, the area under the precision and recall curve of TeCre (0.633) is 20% higher than the second-place RE-NET (0.565). On the WIKIDATA dataset, the area under the accuracy and recall curve of TeCre (0.758) is 13% higher than that of the second-place CyGNet (0.694).

In the experiment, the default facts in the KG are all true, but the facts that are not in the KG according to the closed-world assumption are all false. In order to verify the robustness of TeCre in the face of noisy data, randomly generated facts are added to the data set as false facts in the KG of the experiment. In the experiment, 10%, 20%, 30%, 40%, 50%, 60%, 70%, and 80% of noise facts were added as false facts in the KG. In Figures 7 and 8, we show the accuracy and recall rate of TeCre and the baseline methods with different proportions of error temporal facts in YAGO15K. It can be seen from the figure that TeCre has achieved the best results in these methods. When the error proportion is 20%, TeCre's precision and recall rates reached 0.961 and 0.896, respectively; when the error proportion is 50%, TeCre's precision and recall rates reached 0.870 and 0.827, respectively. Even with 80% false facts added to the dataset, TeCre still achieved 0.818 and 0.798 accuracy and recall rates.

In Figures 9 and 10, we show the accuracy and recall rates of TeCre and the baseline methods when different proportions of error temporal facts are added to ICEWS'14. As can be seen from the figures, TeCre achieved the best results in these methods. When the error proportion was 30%, TeCre's precision and recall rates reached 0.902 and 0.869, respectively, which are higher than the second-place CyGNet precision and recall, as they were 4.15% and 5.21% higher, respectively. When the error proportion was 60%, TeCre's precision and recall rate reached 0.885 and 0.822, respectively, which are higher than the second-place CyGNet precision and recall, as they were 8.45% and 9.13% higher, respectively.

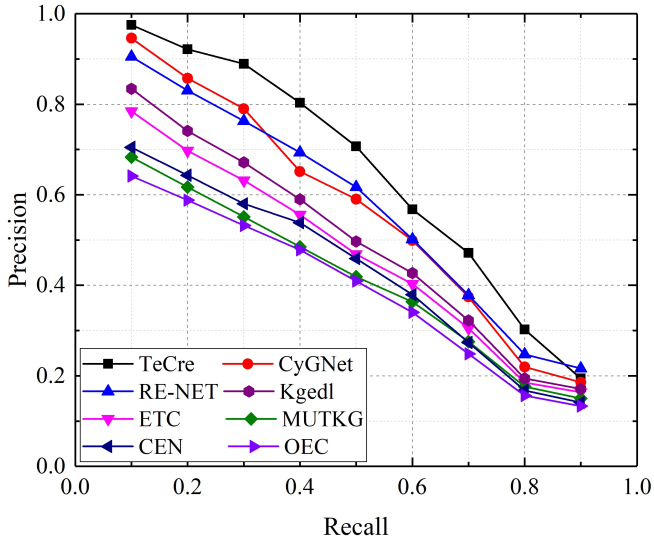

Figure 5. Comparison results on the ICEWS05-15 dataset.

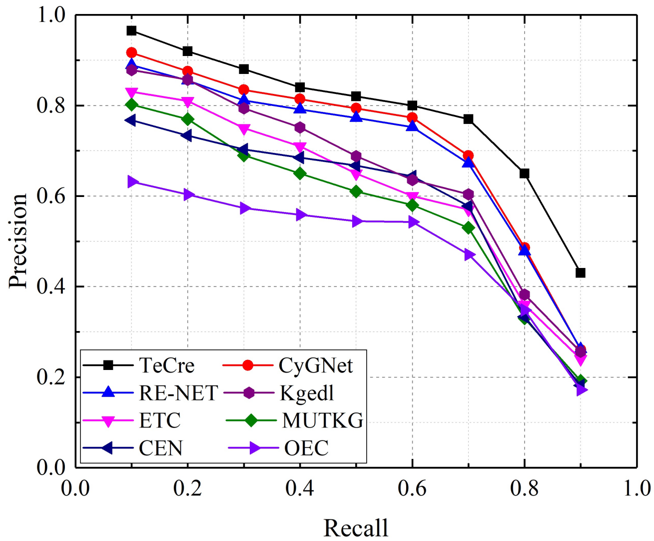

Figure 6. Comparison results on the WIKIDATA dataset.

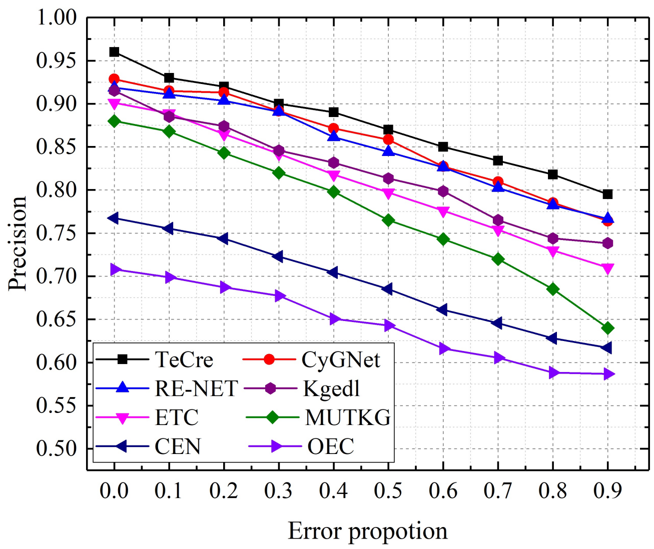

Figure 7. Precision on YAGO15K with different error proportion.

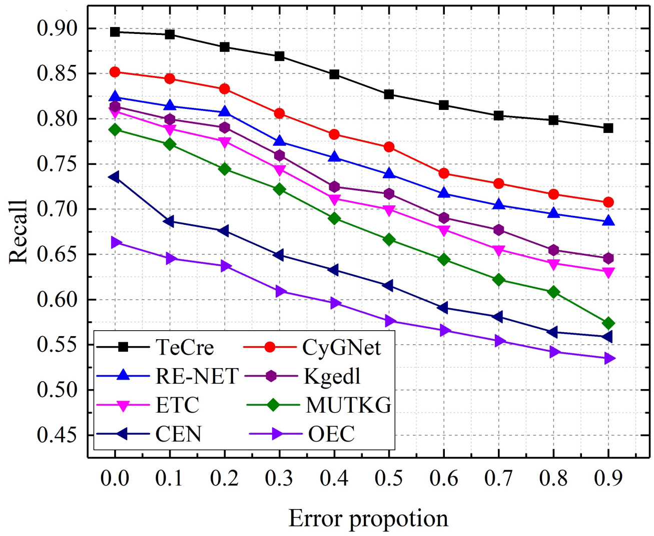

Figure 8. Recall on YAGO15K with different error proportion.

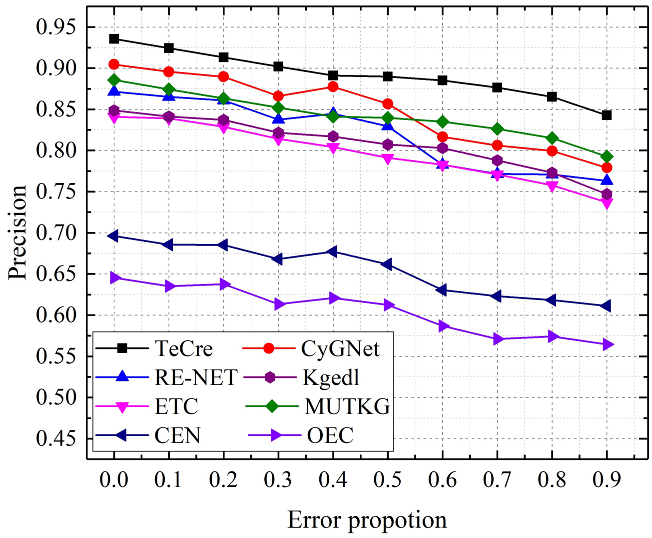

Figure 9. Precision on ICEWS'14 with different error proportion.

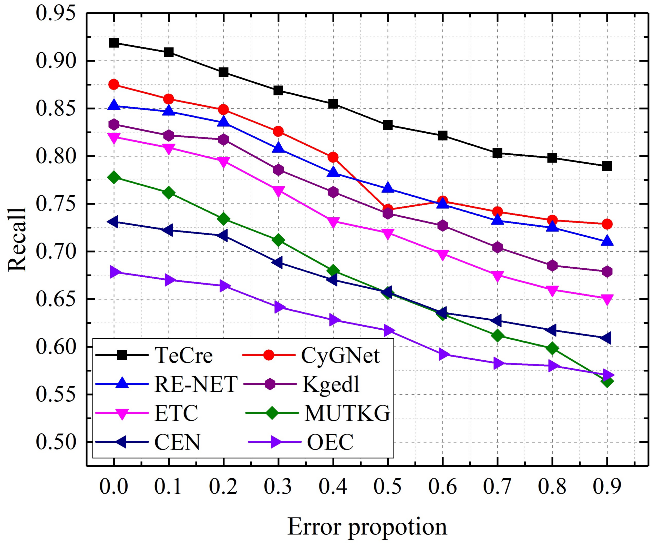

Figure 10. Recall on ICEWS'14 with different error proportion.

To verify the effectiveness of the proposed method on the resolution of temporal conflicts, TeCre deletes the timestamp information in the detected temporal conflict facts and then completes the missing timestamps with the link prediction method. Table 3 shows the results of all methods on the link prediction task. The actual experimental results on YAGO15K, ICEWS'14, ICEWS05-15 and WIKIDATA show that TeCre achieves the best results in MRR and Hits@10. The experimental results on YAGO15K show that TeCre is 8.2% higher than the second-place CyGNet on MRR, and 3.2% higher than the CyGNet on Hits@10. On the ICEWS'14 dataset, TeCre is 9.77% higher than CyGNet on MRR, and 6.2% higher than CyGNet on Hits@10. On ICEWS05-15, TeCre is 5.46% higher than CyGNet on MRR, and 7.7% higher than CyGNet on Hits@10. On WIKIDATA, TeCre is 13.28% higher than CyGNet on MRR, and 5.5% higher than CyGNet on Hits@10. In summary, TeCre achieves the best results with MRR increased by at least 5.46% and Hits@10 increased by at least 3.2%, respectively.

Table 3. Link prediction results on the four datasets (the best results are shown in bold).

	YAGO15K		**ICEWS'14**		**ICEWS05-15**		**WIKIDATA**	
Metrics	**MRR**	**Hits@10**	**MRR**	**Hits@10**	**MRR**	**Hits@10**	**MRR**	**Hits@10**
OEC	0.133	0.178	0.141	0.201	0.258	0.244	0.218	0.306
CEN	0.140	0.197	0.149	0.203	0.285	0.255	0.264	0.341
MUTKG	0.156	0.215	0.177	0.235	0.297	0.267	0.285	0.357
ETC	0.163	0.265	0.186	0.338	0.334	0.368	0.308	0.397
Kgedl	0.171	0.311	0.192	0.356	0.378	0.397	0.324	0.428
RE-NET	0.175	0.334	0.215	0.388	0.376	0.385	0.356	0.446
CyGNet	0.183	0.383	0.256	0.403	0.403	0.406	0.384	0.501
TeCre	**0.198**	**0.415**	**0.281**	**0.465**	**0.425**	**0.483**	**0.435**	**0.556**

According to the above experimental results, the following conclusions are drawn: (1) OEC and CEN only consider a certain time when the fact occurs and do not take into account both the start and end time of the fact, so they cannot capture the long-term dependence of the fact and do not perform well. (2) ETC and MUTKG are more suitable for timestamps time-series facts with shorter intervals, so their performance in ICEWS'14 and ICEWS05-15 datasets is better than that in the other two datasets. (3) CyGNet and Re-Net model historical facts to predict future facts. However, they do not consider the time consistency and the possible time conflicts between facts. (4) Kgedl and TeCre both detect time conflicts through three pre-defined restrictions. However, Kgedl can only capture the time consistency between entities to eliminate conflicts, while TeCre can capture the time consistency between entities and the time consistency between relations to resolve conflicts. Therefore, TeCre performs better than Kgedl. (5) By modeling the time consistency between facts that different lengths of timestamps, TeCre can capture fine-grained temporal features, so TeCre performs best in all datasets.

6. Conclusions

This paper proposes an embedding representation method TeCre based on TKG, which employs TKGE to resolve conflict facts in the KG. TeCre uses the scoring function between entity–relations to discover the facts of temporal conflicts in the TKG according to the temporal conflict constraints. For conflicting facts, the tail entities of these facts are removed from the quadruple. Then the missing tail entities is added to the facts through the KG completion method, thereby solving temporal conflicts in the TKG. The experimental results show that TeCre is significantly better than the best KGE model and the existing temporal KGE model in the conflict detection and conflict resolution on four time-series KGs. Through extensive experiments on real-world datasets, the effectiveness of TeCre compared to traditional and time-sensitive embedding methods is verified. In future work, we will merge the type consistency information to improve the TeCre model's performance, and try to utilize TeCre to complete the missing information of the KG.

Author Contributions: Conceptualization, J.M., C.Z. and Y.W.; methodology, J.M., C.Z. and G.H.; software, J.M., C.Z., Y.C. and Y.Q.; validation, Y.C., Y.W., G.H. and Y.Q.; formal analysis,Y.C., C.Z.; investigation, J.M. and Y.Q.; resources J.M.; data curation, Y.W.; writing—original draft preparation, C.Z.; writing—review and editing, J.M.; visualization, Y.C., Y.W.; supervision, G.H., Y.Q.; project administration, Y.C., G.H.; funding acquisition, Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 62272163; the Songshan Laboratory Pre-research Project, grant number YYJC012022023; the Henan Province Science Foundation for Youths, grant number 222300420230; the Henan Province Science and Technology Department Foundation, grant number 222102210027; the Open Foundation of Henan Key Laboratory of Cyberspace Situation Awareness, grant number HNTS2022005; the Science and Technology Plan Projects of State Administration for Market Regulation, grant number 2021MK067; the Natural Science Foundation of Guangdong Province, China, grant number 2022A1515010417; the Key project of Shenzhen municipality, grant number JSGG20211029095545002; the School-enterprise Collaborative Innovation Project of SZIIT, grant number XQ2021; and the Undergraduate Universities Smart Teaching Special Research Project of Henan Province, grant number 489-29.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets YAGO15K, ICEWS14, ICEWS05-15, and WIKIDATA investigated in this work are publicly available at https://github.com/mniepert/mmkb (accessed on 28 October 2021).

Acknowledgments: The authors would like to thank all anonymous reviewers for their comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P.N.; Hellmann, S.; Morsey, M.; Van Kleef, P.; Auer, S.; et al. Dbpedia—A large-scale, multilingual knowledge base extracted from wikipedia. *Semant. Web* **2015**, *6*, 167–195. [CrossRef]
2. Mitchell, T.; Cohen, W.; Hruschka, E.; Talukdar, P.; Yang, B.; Betteridge, J.; Carlson, A.; Dalvi, B.; Gardner, M.; Kisiel, B.; et al. Never-ending learning. *Commun. ACM* **2018**, *61*, 103–115. [CrossRef]
3. Hoffart, J.; Suchanek, F.M.; Berberich, K.; Weikum, G. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artif. Intell.* **2013**, *194*, 28–61. [CrossRef]
4. Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; Taylor, J. Freebase : A collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, 10–12 June 2008; pp. 1247–1250. [CrossRef]
5. Gaur, M.; Gunaratna, K.; Srinivasan, V.; Jin, H. Iseeq: Information seeking question generation using dynamic meta-information retrieval and knowledge graphs. In Proceedings of the 34th AAAI Conference on Artificial Intelligence, online, 22 February–1 March 2022; Volume 36, pp. 10672–10680. [CrossRef]
6. Liu, L.; Du, B.; Xu, J.; Xia, Y.; Tong, H. Joint Knowledge Graph Completion and Question Answering. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 14–18 August 2022; pp. 1098–1108. [CrossRef]
7. Chen, Y.; Yang, Y.; Wang, Y.; Bai, J.; Song, X.; King, I. Attentive Knowledge-aware Graph Convolutional Networks with Collaborative Guidance for Personalized Recommendation. In Proceedings of the 38th IEEE International Conference on Data Engineering, Kuala Lumpur, Malaysia, 9 May 2022; pp. 299–311. [CrossRef]
8. Xue, B.; Zou, L. Knowledge Graph Quality Management: a Comprehensive Survey. *IEEE Trans. Knowl. Data Eng.* **2022**, 1. [CrossRef]
9. Wang, M.; Qiu, L.; Wang, X. A survey on knowledge graph embeddings for link prediction. *Symmetry* **2021**, *13*, 485. [CrossRef]
10. Thomas, P.T.; Gerhard, W.; Fabian, S. YAGO 4: A Reason-able Knowledge Base. In Proceedings of the The Semantic Web, Athens, Greece, 2–6 November 2020; pp. 583–596.
11. Vrandečić, D.; Krötzsch, M. Wikidata: a free collaborative knowledgebase. *Commun. ACM* **2014**, *57*, 78–85. [CrossRef]
12. Xu, C.; Nayyeri, M.; Alkhoury, F.; Yazdi, H.S.; Lehmann, J. Temporal knowledge graph embedding model based on additive time series decomposition. *arXiv* **2019**, arXiv:1911.07893.
13. Dylla, M.; Sozio, M.; Theobald, M. Resolving Temporal Conflicts in Inconsistent RDF Knowledge Bases. *Coord. Chem. Rev.* **2012**, *2*, 474–493.
14. Lu, L.; Fang, J.; Zhao, P.; Xu, J.; Yin, H.; Zhao, L. Eliminating temporal conflicts in uncertain temporal knowledge graphs. In Proceedings of the International Conference on Web Information Systems Engineering, Dubai, United Arab Emirates, 12–15 November 2018, pp. 333–347. [CrossRef]

15. Abedini, F.; Keyvanpour, M.R.; Menhaj, M.B. Correction Tower: A general embedding method of the error recognition for the knowledge graph correction. *Int. J. Pattern Recognit. Artif. Intell.* **2020**, *34*, 2059034. [CrossRef]
16. Chekol, M.; Pirrò, G.; Schoenfisch, J.; Stuckenschmidt, H. Marrying uncertainty and time in knowledge graphs. In Proceedings of the 31st AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31, pp. 88–94.
17. García-Durán, A.; Dumančić, S.; Niepert, M. Learning Sequence Encoders for Temporal Knowledge Graph Completion. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October–November 2018; pp. 4816–4821. [CrossRef]
18. Wang, Y.; Qiao, Y.; Ma, J.; Hu, G.; Zhang, C.; Sangaiah, A.K.; Zhang, H.; Ren, K. A Novel Time Constraint-Based Approach for Knowledge Graph Conflict Resolution. *Appl. Sci.* **2019**, *9*, 4399. [CrossRef]
19. Jiang, T.; Liu, T.; Ge, T.; Sha, L.; Chang, B.; Li, S.; Sui, Z. Towards time-aware knowledge graph completion. In Proceedings of the 26th International Conference on Computational Linguistics, Osaka, Japan, 11–16 December 2016; pp. 1715–1724.
20. Leblay, J.; Chekol, M.W. Deriving Validity Time in Knowledge Graph. In Proceedings of the Companion of the The Web Conference 2018 on The Web Conference 2018, Lyon, France, 23–27 April 2018; p. 1771–1776. [CrossRef]
21. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-relational Data. In Proceedings of the 27th Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 2787–2795.
22. Yang, B.; Yih, S.W.t.; He, X.; Gao, J.; Deng, L. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
23. Jia, B.; Wang, C.; Zhao, H.; Shi, L. An Entity Linking Algorithm Derived from Graph Convolutional Network and Contextualized Semantic Relevance. *Symmetry* **2022**, *14*, 2060. [CrossRef]
24. Shen, M.; Xu, K.; Yang, K.; Chen, H.H. Towards efficient virtual network embedding across multiple network domains. In Proceedings of the 22nd International Symposium of Quality of Service, Hong Kong, China, 26–27 May 2014, pp. 61–70. [CrossRef]
25. Messner, J.; Abboud, R.; Ceylan, I.I. Temporal knowledge graph completion using box embeddings. In Proceedings of the AAAI Conference on Artificial Intelligence, online, 22 February–1 March 2022; Volume 36, pp. 7779–7787. [CrossRef]
26. Dasgupta, S.S.; Ray, S.N.; Talukdar, P.P. HyTE: Hyperplane-based Temporally aware Knowledge Graph Embedding. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates, 7–11 December 2018; pp. 2001–2011. [CrossRef]
27. Trivedi, R.; Dai, H.; Wang, Y.; Song, L. Know-Evolve: Deep Temporal Reasoning for Dynamic Knowledge Graphs. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 3462–3471.
28. Goel, R.; Kazemi, S.M.; Brubaker, M.; Poupart, P. Diachronic embedding for temporal knowledge graph completion. In Proceedings of the 34th AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 3988–3995.
29. Bianchi, F.; Palmonari, M.; Nozza, D. Towards encoding time in text-based entity embeddings. In Proceedings of the 17th International Semantic Web Conference, Monterey, CA, USA, 8–12 October 2018; Volume 11136, pp. 56–71. [CrossRef]
30. Xiao, H.; Chen, Y.; Shi, X. Knowledge Graph Embedding Based on Multi-View Clustering Framework. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 585–596. [CrossRef]
31. Fu, X.; Sun, X.; Wu, H.; Cui, L.; Huang, J.Z. Weakly supervised topic sentiment joint model with word embeddings. *Knowl.-Based Syst.* **2018**, *147*, 43–54. [CrossRef]
32. Jiang, T.; Liu, T.; Ge, T.; Sha, L.; Li, S.; Chang, B.; Sui, Z. Encoding Temporal Information for Time-Aware Link Prediction. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 2350–2354. [CrossRef]
33. Fleischhacker, D.; Paulheim, H.; Bryl, V.; Völker, J.; Bizer, C. Detecting Errors in Numerical Linked Data Using Cross-Checked Outlier Detection. In Proceedings of the 13th International Semantic Web Conference, Riva del Garda, Italy, 19–23 October 2014; Volume 8796, pp. 357–372. [CrossRef]
34. Wienand, D.; Paulheim, H. Detecting Incorrect Numerical Data in DBpedia. In Proceedings of the Semantic Web: Trends and Challenges - 11th International Conference, Anissaras, Crete, Greece, 25–29 May 2014; Volume 8465, pp. 504–518. [CrossRef]
35. Paulheim, H. Identifying Wrong Links between Datasets by Multi-dimensional Outlier Detection. In Proceedings of the 3rd International Workshop on Debugging Ontologies and Ontology Mappings, Hersonissou, Greece, 26 May 2014; Volume 1162, pp. 27–38.
36. Li, H.; Li, Y.; Xu, F.; Zhong, X. Probabilistic Error Detecting in Numerical Linked Data. In Proceedings of the 26th International Conference on Database and Expert Systems Applications, Valencia, Spain, 1–4 September 2015; Volume 9261, pp. 61–75. [CrossRef]
37. Gao, Y.; Feng, L.; Kan, Z.; Han, Y.; Qiao, L.; Li, D. Modeling Precursors for Temporal Knowledge Graph Reasoning via Auto-encoder Structure. In Proceedings of the 31st International Joint Conference on Artificial Intelligence, IJCAI-22, Vienna, Austria, 23–29 July 2022; pp. 2044–2051. [CrossRef]
38. Padia, A. Cleaning Noisy Knowledge Graphs. In Proceedings of the Doctoral Consortium at the 16th International Semantic Web Conference, Vienna, Austria, 22 October 2017; Volume 1962.
39. Chen, Y.; Wang, D.Z. Knowledge expansion over probabilistic knowledge bases. In Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, Snowbird, UT, USA, 22–27 June 2014; pp. 649–660. [CrossRef]

40. Chekol, M.W.; Pirrò, G.; Schoenfisch, J.; Stuckenschmidt, H. TeCoRe: Temporal Conflict Resolution in Knowledge Graphs. *Proc. VLDB Endow.* **2017**, *10*, 1929–1932. [CrossRef]
41. Zhang, C.; Pang, H.; Liu, J.; Tang, S.; Zhang, R.; Wang, D.; Sun, L. Toward Edge-Assisted Video Content Intelligent Caching With Long Short-Term Memory Learning. *IEEE Access* **2019**, *7*, 152832–152846. [CrossRef]
42. Jin, W.; Qu, M.; Jin, X.; Ren, X. Recurrent Event Network: Autoregressive Structure Inferenceover Temporal Knowledge Graphs. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Barcelona, Spain, 8–12 November 2020, pp. 6669–6683. [CrossRef]
43. Zhu, C.; Chen, M.; Fan, C.; Cheng, G.; Zhang, Y. Learning from history: Modeling temporal knowledge graphs with sequential copy-generation networks. In Proceedings of the 32rd AAAI Conference on Artificial Intelligence, online, 22 February–1 March 2021; Volume 35, pp. 4732–4740.
44. Li, Z.; Guan, S.; Jin, X.; Peng, W.; Lyu, Y.; Zhu, Y.; Bai, L.; Li, W.; Guo, J.; Cheng, X. Complex Evolutional Pattern Learning for Temporal Knowledge Graph Reasoning. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Dublin, Ireland, 22–27 May 2022; pp. 290–296. [CrossRef]