applied sciences

MDPI

Article

Cluster-Level Evidence-Enhanced Truth Inference for Knowledge Graph Fusion

Yukun Dong [1,2,3,*], Xinran Meng [1], Shuo Wang [1], Mingcheng Chen [1] and Zheng Yu [1]

[1] Qingdao Institute of Software, College of Computer Science and Technology, China University of Petroleum (East China), Qingdao 266580, China; z24070142@s.upc.edu.cn (X.M.); z24070154@s.upc.edu.cn (S.W.); s24070064@s.upc.edu.cn (M.C.); yu022021@163.com (Z.Y.)
[2] Shandong Provincial Key Laboratory of Intelligent Oil and Gas Industrial Software, Qingdao 266580, China
[3] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
* Correspondence: dongyk@upc.edu.cn

Abstract

Factual conflicts after entity alignment in knowledge graph fusion (KGF) can affect the consistency and reliability of fusion results. To address the difficulty of directly transferring existing traditional truth inference and fact verification methods to knowledge graph fusion, a cluster-level evidence-enhanced truth-inference method for knowledge graph fusion (CETI-KGF) is proposed. The method adopts a two-stage fact modeling strategy to organize semantically related multi-source facts and perform truth inference at the cluster level. Semantic-aware fact clustering (SFC) is constructed to divide facts into different fact groups. A cluster-level credibility evidence modeling (CEM) module is then employed to construct cluster-level evidence based on source support, count-based support, and local consistency within the knowledge graph. Finally, structured prompts guide the large language model (LLM) to jointly infer truths in a mixed single-truth and multi-truth setting, and ultimately map the resulting decisions to the facts within each cluster. The FR-EN subset of the entity alignment (EA) dataset DBP15K is used as the experimental setting, and a dataset containing 5429 conflict groups and 17,067 candidate facts is constructed for experimentation. The proposed method achieves the highest Error Precision and Error F1 among the compared methods, reaching 81.0% and 75.7%, respectively. Compared with directly using an LLM, the proposed method improves Error Precision and Error F1 by 10.5 and 6.0 percentage points, respectively.

Keywords: knowledge graph fusion; truth inference; entity alignment; fact verification; large language models

Check for updates

Academic Editor: Christos Bouras

Received: 24 September 2026
Revised: 3 November 2026
Accepted: 5 November 2026
Published: 8 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Knowledge graphs (KGs), as a form of structured knowledge representation, have increasingly been integrated with large language models (LLMs) in recent years to support tasks such as knowledge augmentation, knowledge-based question answering, and complex reasoning [1]. With the continuous construction of KGs across different domains and languages, a single KG is often insufficient to provide comprehensive knowledge coverage, which in turn creates a growing demand for integrating newly constructed KGs into existing ones. Consequently, knowledge graph fusion has become an important problem in KG construction and application.

Knowledge graph fusion mainly involves several subtasks, including schema alignment (SA), entity alignment (EA), and truth inference (TI). Schema alignment focuses

on knowledge correspondence at the class level and aims to establish mappings among concepts, attributes, and relations across different KGs. Entity alignment seeks to identify entities from different KGs that refer to the same real-world object, and related research has evolved from traditional representation learning approaches toward methods based on pretrained models and LLMs [2]. Truth inference is typically performed after entity alignment and addresses conflicts caused by multiple distinct facts associated with the same entity–relation pair, determining which facts should be retained, merged, or discarded. Existing studies have primarily focused on the entity alignment stage. However, truth inference is equally important in knowledge graph fusion. Without an effective conflict resolution mechanism, the fused KG may contain inconsistent or erroneous facts, thereby degrading both its overall accuracy [3] and the reliability of downstream tasks.

Existing studies related to truth reasoning can be broadly divided into two categories. The first category comprises traditional truth-inference methods. These methods are generally designed for generic scenarios and identify credible facts from multiple conflicting candidates by exploiting source reliability, candidate-value voting, iterative optimization, and other mechanisms [4]. Traditional truth inference can be further divided into single-truth and multi-truth settings. However, knowledge graph fusion involves a mixture of these two settings, and candidate facts from heterogeneous sources may exhibit aliases, differences in semantic granularity, or differences in information completeness. Treating such facts as independent competing candidates may fragment their supporting evidence and affect subsequent truth inference. Therefore, directly transferring conventional truth-inference methods to knowledge graph fusion is not sufficiently reliable, and a truth-inference method specifically tailored to fusion scenarios is required.

The second category consists of KG fact-checking and error detection methods. Such approaches typically verify the validity of individual triples within a KG by exploiting paths, local graph structures, embedding representations, or language models to determine whether a given fact is credible. However, conflict resolution in knowledge graph fusion is not merely a matter of determining whether a triple is correct within the internal structure of a KG. These methods essentially rely on internal information; in other words, a triple may be regarded as valid as long as it is consistent with the logical relations already present in the KG. Nevertheless, such internal consistency does not necessarily imply that the factual statement represented by the triple is objectively correct. Therefore, when addressing truth inference in knowledge graph fusion, internal KG information can only serve as auxiliary evidence rather than the primary basis for determining factual correctness.

In summary, although conventional truth-inference methods and KG fact-checking approaches are closely related to this study, neither can be directly applied to knowledge graph fusion. This limitation mainly arises from the unique characteristics of KGs and fusion scenarios. On the one hand, effective mechanisms from traditional truth inference for acquiring external knowledge should be exploited so that heterogeneous facts provided during KG fusion can be transformed into a form suitable for verification using external evidence. On the other hand, internal information from the KG should not be completely discarded. Instead, appropriate internal signals should be retained as auxiliary evidence to complement the credibility assessment of candidate facts. Moreover, a unified mechanism capable of handling both single-truth and multi-truth cases is required.

To address these challenges, this paper proposes a cluster-level evidence-enhanced truth-inference method for knowledge graph fusion (CETI-KGF). The central idea is to semantically organize facts from different sources into distinct clusters and use cluster-level evidence to guide an LLM in jointly performing truth inference for both single-truth and multi-truth cases. CETI-KGF first employs a three-layer semantics-aware clustering strategy to group candidate facts into clusters representing different semantic alternatives. It then

constructs cluster-level evidence based on source support, intra-cluster support strength, and local consistency. Finally, a structured prompting mechanism is used to guide the LLM in reasoning about the truth type of each fact cluster and simultaneously performing truth inference for both single-truth and multi-truth facts. The facts ultimately incorporated into the fused KG are selected from the corresponding clusters. In this way, CETI-KGF organizes semantically related facts at the cluster level while preserving their supporting evidence for subsequent truth inference.

The main contributions of this work are summarized as follows:

- We propose a cluster-level evidence-enhanced truth-inference method for knowledge graph fusion. By providing reasoning evidence at the cluster level, the proposed method aggregates supporting information from semantically related facts and uses the resulting cluster-level evidence to improve truth inference in knowledge graph fusion.
- We design a fact clustering strategy that distinguishes between weak conflicts and distinct candidate alternatives. Distinct candidate alternatives refer to semantically different facts assigned to separate clusters, whereas weak conflicts refer to non-mutually-exclusive semantic variations. This distinction enables semantically related facts to be organized into the same cluster so that their supporting evidence can be jointly utilized in subsequent truth inference.
- We extract factual evidence at the cluster level and use it to guide the LLM in first reasoning about the underlying truth type and then performing truth inference for both single-truth and multi-truth cases through a unified prompting template.

2. Related Work

In the context of knowledge graph fusion, multiple subtasks are involved. Among them, conflict detection and truth inference are mainly addressed through four categories of approaches: traditional truth-inference methods, knowledge graph fact-checking methods, pretrained model-based methods, and large language model-based methods.

2.1. Traditional Truth-Inference Methods

Traditional truth-inference methods determine true information by analyzing conflicting facts provided by multiple data sources for the same object and attribute, while jointly estimating source reliability and fact credibility. The underlying assumption is that reliable sources are more likely to provide true facts, whereas facts supported by a larger number of reliable sources are more likely to be credible.

TruthFinder performs truth selection by iteratively updating source trustworthiness and fact confidence [5]. The Accu family of methods further considers source dependencies and information copying relationships [6]. LTM employs a Bayesian model to characterize the relationship between source quality and fact truthfulness and explicitly allows multiple values associated with the same object–attribute pair to be inferred as true, thereby supporting multi-truth discovery [7]. CRH formulates truth discovery as a joint optimization problem involving source reliability and fact credibility [8]. CATD introduces confidence-interval estimation to alleviate insufficient source reliability estimation in long-tail data scenarios [9].

In summary, traditional truth-inference methods can effectively exploit source reliability to resolve general data conflicts, and some approaches, such as LTM, explicitly support multiple true values. However, these methods generally do not explicitly model entity aliases, semantic variations among related facts, and graph-structural evidence in knowledge graph fusion, which limits their direct applicability to the heterogeneous semantic conflicts considered in this work.

2.2. Fact-Checking Methods

Fact-checking methods determine the credibility of a target triple by exploiting structural evidence formed by existing entities and relations in a knowledge graph. These methods generally assume that a fact is more likely to be credible when strong semantic paths or multiple mutually supportive relational paths exist between its head and tail entities.

PRA learns relational paths between entities through path-constrained random walks [10]. Knowledge Stream estimates fact credibility according to the strength of information flow within a knowledge graph [11]. PaTyBRED-RF, an implementation of PaTyBRED using a random forest classifier, performs fact classification by extracting relational-path features between head and tail entities [12]. COPAAL identifies corroborative paths in a knowledge graph to provide interpretable structural evidence for fact checking [13]. Tan et al. constructed a causal knowledge graph to enhance deductive reasoning capabilities for fact checking [14].

Therefore, fact-checking methods can effectively exploit structural information within knowledge graphs. However, their performance is susceptible to KG sparsity, and they have difficulty handling semantically equivalent facts and multi-truth relations within the same conflict group.

2.3. Pretrained Model-Based Methods

Pretrained model-based methods exploit semantic knowledge learned from large-scale textual corpora by transforming entity names, relations, and triples into textual or vector representations and assessing fact plausibility according to the degree of correspondence between these representations. Compared with path-based methods, pretrained models can leverage textual semantics for inference even when explicit structural paths are unavailable.

KG-BERT converts triples into textual sequences and employs BERT for triple classification [15]. StAR separately encodes the head entity–relation pair and the tail entity, thereby improving computational efficiency while preserving semantic information about entities and relations [16]. SimKGC employs contrastive learning and large-scale negative sampling to improve candidate entity matching and knowledge graph completion performance [17].

Although pretrained models exhibit strong semantic representation capabilities, they are primarily designed for link prediction and knowledge graph completion. Consequently, they lack explicit modeling of data provenance, dependencies among candidate facts, and conflict structures involving multiple truths.

2.4. Large Language Model-Based Methods

Large language models acquire strong knowledge memorization, semantic understanding, and contextual reasoning capabilities through large-scale pretraining. Existing studies typically organize target triples together with relevant evidence into prompts and instruct LLMs to determine whether a fact holds. Few-shot examples can also be incorporated to improve the adaptability of LLMs to specific tasks.

KAPING incorporates relevant facts from knowledge graphs as external knowledge for LLMs, thereby improving the reliability of knowledge-based question answering [18]. StructGPT enables LLMs to perform multi-step reasoning over structured data, including knowledge graphs, by invoking structured-data interfaces [19]. Wu et al. proposed the BEKO framework to systematically investigate bidirectional enhancement mechanisms between LLMs and knowledge graphs [20]. KARMA employs multiple specialized agents to collaboratively perform entity discovery, relation extraction, schema alignment, and conflict resolution, while cross-agent verification is used to improve the reliability of newly acquired knowledge [21]. In addition, models such as GPT, Qwen, and DeepSeek [22–24]

have increasingly been applied to knowledge graph question answering, fact checking, and knowledge completion.

Despite their strong semantic reasoning capabilities, LLM-based methods remain sensitive to prompt formulation and biases in their internal knowledge. When unorganized conflicting facts are directly provided as input, the models may still produce erroneous judgments.

To address these limitations, this paper focuses on multi-truth conflict scenarios in knowledge graph fusion. Candidate facts are first organized using semantics-aware clustering, after which cluster-level evidence is constructed from semantic information, source support, and graph structure. The resulting structured evidence is then incorporated into LLM prompts to jointly distinguish genuine multiple truths, weak-conflict fact variants, and erroneous facts.

3. Our Method

In the process of multi-source knowledge graph fusion, entity alignment can only identify equivalent entities originating from different sources and cannot resolve conflicting facts. Conflicting facts refer to cases in which aligned entities are associated with different tail entities under the same relation. For example, given (h, r, t_1) from knowledge graph KG_1 and (h, r, t_2) from knowledge graph KG_2, the fusion process introduces two different tail entities, t_1 and t_2. The objective of the proposed method is to resolve such conflicts arising during knowledge graph fusion and perform truth inference over the conflicting facts.

Directly determining the truthfulness of individual facts while ignoring their semantic relationships may cause weak-conflict variants to be treated as independent competing candidates. This treatment fragments the supporting evidence associated with one semantic alternative and may obscure the most appropriate factual representation under the current entity–relation context. Candidate facts are often interrelated and numerous; presenting them as an unorganized set can also increase the difficulty of LLM-based inference. In addition, truth inference in knowledge graph fusion inherently combines single-truth and multi-truth settings. The proposed method must therefore distinguish and handle both task types without imposing a universal one-truth constraint.

To address these issues, we propose a cluster-level evidence-enhanced truth-inference method for knowledge graph fusion, as illustrated in Figure 1. The core idea of the framework is to semantically group facts provided by different knowledge graphs into distinct clusters and leverage cluster-level evidence to guide a large language model in performing truth inference for both single-truth and multi-truth facts. First, a three-layer semantics-aware clustering method is employed to group candidate facts into different clusters. Second, cluster-level evidence is constructed based on source support, intra-cluster support strength, and local consistency. Finally, structured prompts are used to guide the large language model in reasoning about the truth type of each fact cluster and simultaneously performing truth inference for both single-truth and multi-truth facts.

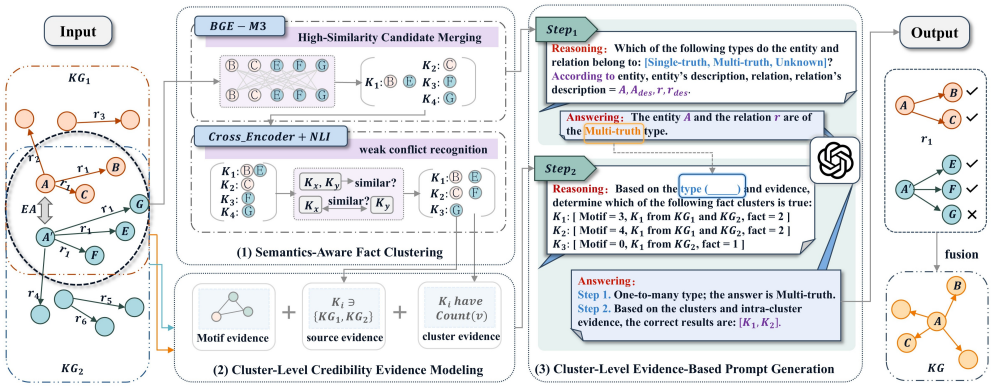


Figure 1. Overall framework of the proposed CETI-KGF method.

3.1. Task Definition

Entity, relation, and schema alignment are treated as prerequisites rather than components evaluated in this study. After alignment, suppose that the triples to be fused from two knowledge graphs, KG_1 and KG_2, are denoted as $(a, r, v_i) \in KG_1$ and $(a', r', v_j) \in KG_2$, where $a \equiv a'$ and $r \equiv r'$ denote aligned entities and relations. The candidate tail values satisfy $v_i \in V_1$, $i = 1, 2, \ldots, m$, and $v_j \in V_2$, $j = 1, 2, \ldots, n$.

Knowledge graph fusion contains both single-truth and multi-truth entity–relation pairs. The task is therefore to identify the complete set of true facts in $V_1 \cup V_2$, rather than forcing every group to contain exactly one truth. Candidate facts with related semantics are organized into clusters to aggregate their supporting evidence. When a concrete representation is required for downstream output, representative selection is performed after cluster-level truth inference and does not change the inferred cluster labels. The current evaluation measures truth-inference correctness; the quality of the final fused-KG output is outside its scope.

3.2. Preprocessing

Before fact clustering and truth inference, the multi-source knowledge graphs are preprocessed to construct conflict sets. The proposed method assumes that entity, relation, and schema alignment have already been completed. The resulting correspondences are used to aggregate facts associated with the same aligned entity–relation pair into one candidate fact set. Alignment quality is not evaluated in this study.

3.3. Semantics-Aware Fact Clustering

For the same entity under the same relation, different knowledge graph sources often introduce strong conflicts at the fact level. For example, knowledge graph KG_1 may provide (Tim Cook, employer, Apple Inc., Cupertino, CA, USA), whereas knowledge graph KG_2 may provide (Tim Cook, employer, Microsoft Corporation, Redmond, WA, USA). However, conflicts introduced by knowledge graph fusion extend beyond those considered in conventional truth-inference tasks and give rise to additional types of weak conflicts. These weak conflicts can be categorized into three types: (1) synonym/alias conflicts; (2) fine-grained differences; and (3) differences between partial and complete information. At the fact level, these three types of weak conflicts do not necessarily represent erroneous information, yet they cannot be directly eliminated during preprocessing. This is because such candidate facts are not independent observations, as typically assumed in conventional truth inference, but are instead interdependent and complementary. If all candidate facts are directly evaluated within the original candidate set, weak conflicts may

be incorrectly treated as independent competing items, leading to fragmented supporting evidence and potential evidence loss.

To address this issue, we first design a semantics-aware fact clustering method (SFC). It distinguishes conflicting facts into strong and weak conflicts by exploiting both semantic similarity and directional semantic relations among candidate facts. Semantically related weak-conflict facts are grouped into the same cluster, whereas distinct factual alternatives are assigned to different clusters. In the experiments, this module initializes clusters with candidate facts from the target knowledge graph KG_2 and sequentially assigns candidate facts from the source knowledge graph KG_1 to existing clusters or creates new clusters when necessary. For reproducibility, the KG_1 candidate values in V_1 are processed in ascending order of their retained unique candidate-fact identifiers.

Let the aligned entity be e and the aligned relation be r. Knowledge graphs KG_1 and KG_2 provide the following candidate tail-value sets, respectively:

$$V_1 = \{v_1^1, v_1^2, v_1^3, \ldots, v_1^m\} \tag{1}$$

$$V_2 = \{v_2^1, v_2^2, v_2^3, \ldots, v_2^n\} \tag{2}$$

The n candidate values in V_2 are used to initialize the candidate clusters, while the m values in V_1 are treated as objects to be assigned. After initialization, the set of candidate clusters is obtained as

$$\mathcal{K}_{e,r} = \{K_1, K_2, \ldots, K_n\}, \qquad K_i = \{v_2^i\} \tag{3}$$

Here, K_i denotes the i-th candidate cluster of entity e under relation r.

Each cluster initially contains one candidate value from KG_2. For each candidate value v_1^j in V_1, a three-layer semantic decision strategy is then applied to determine its relationship with the existing candidate clusters.

3.3.1. High-Similarity Candidate Merging with BGE-M3

During candidate fact clustering, the first step is to identify synonymous facts that are highly semantically consistent and can therefore be merged into the same candidate cluster. To this end, BGE-M3 [25] computes the semantic similarity between each candidate value to be assigned and every member of an existing candidate cluster. BGE-M3 maps text into a continuous vector space, where semantically similar texts are positioned closer to each other.

Let the candidate value to be assigned be v, and let u denote a member of an existing candidate cluster K_i. After encoding with BGE-M3, their semantic vector representations are obtained as follows:

$$\boldsymbol{h}_v = BGE(v) \tag{4}$$

$$\boldsymbol{h}_u = BGE(u) \tag{5}$$

The semantic compatibility between v and K_i is defined as the maximum cosine similarity between v and any member of the cluster:

$$Sim(v, K_i) = \max_{u \in K_i} cos(\boldsymbol{h}_v, \boldsymbol{h}_u) = \max_{u \in K_i} \frac{\boldsymbol{h}_v \cdot \boldsymbol{h}_u}{\|\boldsymbol{h}_v\| \cdot \|\boldsymbol{h}_u\|} \tag{6}$$

For each candidate value v to be assigned, its compatibility with every existing candidate cluster is computed, and the cluster with the highest score is selected:

$$K^* = \arg\max_{K_i \in \mathcal{R}} Sim(v, K_i) \tag{7}$$

where $\mathcal{R}$ denotes the set of candidate clusters that have already been constructed. If the maximum similarity exceeds the predefined threshold τ_s, the candidate value v is considered highly semantically consistent with candidate cluster K^* and can therefore be directly assigned to that cluster:

$$\max_{K_i \in \mathcal{R}} Sim(v, K_i) \geq \tau_s \tag{8}$$

The merging operation is then performed as

$$K^* = K^* \cup \{v\} \tag{9}$$

3.3.2. Directional Inference-Based Weak-Conflict Identification

After the first-stage semantic model assigns a portion of the candidate facts, some weak conflicts involving differences in granularity and information completeness may still remain unclustered. Typical examples include (Satya Nadella, employer, Microsoft Corporation) from knowledge graph KG_1 and (Satya Nadella, employer, Microsoft) from KG_2. Such candidate values may not exceed the high-similarity threshold for any member of an existing cluster. Treating them immediately as new clusters may split one semantic alternative and fragment its supporting evidence. To identify these weak conflicts, pairwise semantic matching and directional semantic inference are applied between the candidate value and each member of the existing clusters.

Specifically, for a candidate value v and a member $u \in K_i$, a Cross-Encoder model [26] performs pairwise matching and produces $CE(v, u)$. Unlike vector-based similarity models, the Cross-Encoder jointly encodes the two values as a single input and can capture fine-grained semantic differences.

Meanwhile, a natural language inference (NLI) model performs bidirectional entailment inference. The candidate value and each cluster member are converted into natural-language statements:

$$s_v = \text{The value of relation } r \text{ is } v \tag{10}$$

$$s_u = \text{The value of relation } r \text{ is } u \tag{11}$$

Then, $NLI(s_v, s_u)$ and $NLI(s_u, s_v)$ are computed separately to obtain the entailment probabilities in both directions:

$$Entail(v \rightarrow u), \quad Entail(u \rightarrow v) \tag{12}$$

For each input pair, the logits corresponding to contradiction, neutral, and entailment are normalized using Softmax, and the probability of the entailment class is used as the directional entailment score. Candidate v is eligible for cluster K_i when at least one member $u \in K_i$ simultaneously satisfies

$$CE(v, u) \geq \tau_c \tag{13}$$

$$\min\{Entail(v \rightarrow u), Entail(u \rightarrow v)\} \geq \tau_e \tag{14}$$

Among all eligible clusters, the cluster containing the strongest passing pair is selected, and v is merged into it. Both directional entailment scores must therefore satisfy τ_e for the same pair. Here, τ_c and τ_e denote the predefined thresholds. In the experiments, cross-encoder/stsb-roberta-base [27] is adopted as the Cross-Encoder, while roberta-large-mnli [28] is used as the NLI model.

3.3.3. Distinct Candidate Cluster Generation

If a candidate value v to be assigned does not satisfy either of the above two levels of similarity criteria with any existing candidate cluster, it is considered to represent a new candidate fact group. A new candidate cluster is therefore created for it:

$$K_{new} = \{v\} \tag{15}$$

Through the above three-stage decision procedure, each original candidate fact is either merged into an existing cluster or assigned to a newly created cluster. In this way, semantically related weak-conflict facts are organized into candidate fact groups, providing the basis for subsequent cluster-level truth inference. Algorithm 1 summarizes the implemented initialization, member-level comparison, sequential assignment, cluster merging, and new-cluster generation procedure. Comparing a candidate with all current cluster members avoids the need for an undefined representative-update operation during clustering.

Algorithm 1 Semantics-Aware Fact Clustering (SFC)

Require: Candidate value sets $V_1 = \{v_1^1, v_1^2, \ldots, v_1^m\}$ and $V_2 = \{v_2^1, v_2^2, \ldots, v_2^n\}$; thresholds τ_s, τ_c, and τ_e
Ensure: Candidate cluster set $\mathcal{K}$
Initialize $\mathcal{K} \leftarrow \{K_1, K_2, \ldots, K_n\}$, where $K_i \leftarrow \{v_2^i\}$
Sort V_1 in ascending order of retained unique candidate-fact identifiers
for each candidate value $v \in V_1$ **do**
  Compute $Sim(v, K_i) = \max_{u \in K_i} cos(\boldsymbol{h}_v, \boldsymbol{h}_u)$ for each $K_i \in \mathcal{K}$
  $K^* \leftarrow \arg\max_{K_i \in \mathcal{K}} Sim(v, K_i)$
  if $Sim(v, K^*) \geq \tau_s$ **then**
    $K^* \leftarrow K^* \cup \{v\}$
    continue
  end if
  $K^* \leftarrow \varnothing$; $s_{\text{best}} \leftarrow -\infty$
  for each cluster $K_i \in \mathcal{K}$ **do**
    for each member $u \in K_i$ **do**
      $q_{nli} \leftarrow \min\{Entail(v \rightarrow u), Entail(u \rightarrow v)\}$
      if $CE(v, u) \geq \tau_c$ **and** $q_{nli} \geq \tau_e$ **then**
        $s \leftarrow \max\{cos(\boldsymbol{h}_v, \boldsymbol{h}_u), \min\{CE(v, u), q_{nli}\}\}$
        if $s > s_{\text{best}}$ **then**
          $K^* \leftarrow K_i$; $s_{\text{best}} \leftarrow s$
        end if
      end if
    end for
  end for
  if $K^* \neq \varnothing$ **then**
    $K^* \leftarrow K^* \cup \{v\}$
  else
    $K_{\text{new}} \leftarrow \{v\}$; $\mathcal{K} \leftarrow \mathcal{K} \cup \{K_{\text{new}}\}$
  end if
end for
return $\mathcal{K}$

3.4. Cluster-Level Credibility Evidence Modeling

After clustering, candidate facts are organized into multiple candidate clusters, each representing a distinct candidate fact group. Facts within the same cluster generally correspond to different expressions or weak-conflict variants of the same underlying fact, whereas different clusters represent semantically distinct candidate alternatives. The clustering stage itself does not determine whether these alternatives are factually contradictory;

their credibility is evaluated during subsequent truth inference. Therefore, subsequent truth inference is no longer performed at the level of individual facts, but instead evaluates the credibility of candidate clusters.

To support this process, cluster-level credibility evidence modeling (CEM) is constructed from three perspectives: source support, intra-cluster support strength, and local consistency within the knowledge graph. The resulting evidence is then used to assist the large language model in subsequent cluster-level truth inference.

The candidate cluster set is defined as $\mathcal{K}_{e,r} = \{K_1, K_2, K_3, \ldots, K_m\}$ For each candidate cluster K_i, its cluster-level evidence is represented as

$$S(K_i) = \{S_{src}(K_i), S_m(K_i), S_{ctx}(K_i)\} \tag{16}$$

where S_{src} denotes source support, S_m denotes intra-cluster support strength, and S_{ctx} denotes local consistency.

3.4.1. Source Support

Source support measures how many distinct knowledge sources support a candidate cluster. In multi-source knowledge graph fusion, if the facts represented by a candidate cluster are provided by multiple sources, the cluster is generally considered more credible. For example, let $K_1 = v_1, v_2$ and $K_2 = v_1$. Since K_1 is supported by both KG_1 and KG_2, its credibility is considered higher than that of K_2. However, this does not imply that K_1 is necessarily correct.

Unlike traditional truth-inference methods, which count source support for individual candidate values, the proposed method measures source support at the candidate-cluster level. Specifically, if a fact from a given source is assigned to a candidate cluster, that source is regarded as supporting the cluster.

Let $src(K_i)$ denote the set of sources supporting candidate cluster K_i. The source support is defined as

$$S_{src}(K_i) = |src(K_i)| \tag{17}$$

where $|src(K_i)|$ denotes the number of distinct sources supporting the candidate cluster.

3.4.2. Intra-Cluster Support Strength

Intra-cluster support strength measures how many facts are aggregated within a candidate cluster. Traditional methods typically count votes only for completely identical candidate values. In the context of knowledge graph fusion, however, if multiple facts are assigned to the same cluster after clustering, this indicates that they semantically support the same candidate fact group. A larger number of members within a cluster generally implies stronger internal support for that candidate fact group and therefore a higher likelihood of being correct. For example, given $K_1 = v_1, v_2, v_3$ and $K_2 = v_1$, the three members in K_1 indicate that K_1 has higher credibility than K_2.

Let $|K_i|$ denote the number of members in candidate cluster K_i. The intra-cluster support strength is defined as

$$S_m(K_i) = \frac{|K_i|}{\sum_{j=1}^{n} |K_j|} \tag{18}$$

The denominator represents the total number of members across all candidate clusters in the current conflict group. The normalized $S_m(K_i)$ falls within the range $(0, 1]$. This score reflects the relative internal support received by candidate cluster K_i within the current

conflict group and is jointly used with source support and local graph consistency for subsequent truth inference.

3.4.3. Local Graph Consistency

The first two types of evidence are mainly derived from information external to the knowledge graph. However, due to the inherent graph structure of KGs, the credibility of a fact can also be assessed using internal structural evidence. In fact-checking tasks over a single knowledge graph, confidence estimation is commonly based on graph paths or embedding representations. However, embedding-based evidence tends to reflect the internal coherence of a knowledge graph. That is, a fact may be logically consistent with the current KG even if it is not objectively correct. Therefore, path-based evidence is considered more general for this task.

Path-based methods assess fact credibility according to structural paths that exist within the knowledge graph. Common approaches include shortest-path-based methods, the Path Ranking Algorithm (PRA), node degree, and motif-based methods [10]. In this study, node degree and motifs are used to extract structural evidence for candidate facts.

Definition 1. *Given a knowledge graph $KG = (V, E)$, where V denotes the set of nodes and E denotes the set of edges, the degree of any node $v \in V$, denoted by $deg(v)$, is defined as the number of edges directly connected to that node. In a directed knowledge graph, the degree of a node can be expressed as the sum of its in-degree and out-degree: $deg(v) = deg_{in}(v) + deg_{out}(v)$, where $deg_{in}(v)$ denotes the number of edges pointing to node v, and $deg_{out}(v)$ denotes the number of edges directed from node v to other nodes.*

Definition 2. *Given a knowledge graph $KG = (V, E)$, a motif containing i nodes is defined as $H = (M, V_H)$, where $V_H \subseteq V$ denotes the set of nodes in the motif and $|V_H| = i$. $M \in \{0,1\}^{i \times i}$ is a binary adjacency matrix representing the connectivity pattern among nodes within the motif. If $M_{ab} = 1$, an edge exists between the a-th node and the b-th node; otherwise, no such edge exists.*

Node degree and motifs characterize local consistency from different perspectives. Node degree reflects the connectivity activity of a candidate value within the knowledge graph, whereas motifs characterize its local structural support. For each candidate fact $f = (h, r, t)$ originating from source KG_s, structural evidence is calculated only on its original source graph $G^{(s)}$, without merging the two source KGs in advance. For the degree-based score, the in-degree and out-degree of both the head and tail entities are used. If the candidate edge (h, r, t) already exists in the original source graph, this edge is excluded when calculating the degrees to prevent the candidate fact itself from directly contributing to its structural evidence. The resulting degree value is log-transformed and robustly normalized using the 5th and 95th percentiles:

$$D(f) = \text{clip}\left(\frac{\ln\left[1 + \tilde{d}_f(h) + \tilde{d}_f(t)\right] - Q^D_{0.05}}{Q^D_{0.95} - Q^D_{0.05}}, 0, 1\right) \tag{19}$$

where $\tilde{d}_f(h)$ and $\tilde{d}_f(t)$ denote the degrees of the head and tail entities after masking the candidate edge, and $Q^D_{0.05}$ and $Q^D_{0.95}$ denote the 5th and 95th percentiles of the raw degree scores, respectively. For the motif-based score, we count the two-hop common neighbors of the head and tail entities in the undirected neighborhood view of the corresponding source graph. The direct candidate edge itself is not counted; only third-party common

neighbors provide motif evidence. The common-neighbor count is also log-transformed and normalized using the 5th and 95th percentiles:

$$M(f) = \text{clip}\left(\frac{\ln[1 + |N_s(h) \cap N_s(t)|] - Q^M_{0.05}}{Q^M_{0.95} - Q^M_{0.05}}, 0, 1\right) \tag{20}$$

where $N_s(x)$ denotes the neighbor set of node x in the source graph, and $Q^M_{0.05}$ and $Q^M_{0.95}$ denote the corresponding percentile values for the raw motif scores. For each candidate cluster, $S_{deg}(K_i)$ and $S_{motif}(K_i)$ are obtained by averaging the normalized degree and motif scores of all member facts, respectively. Injected erroneous edges are not added to the original source graph when structural evidence is calculated. The local consistency of the cluster is then defined as

$$S_{\text{ctx}}(K_i) = \lambda \cdot S_{\text{deg}}(K_i) + (1 - \lambda) \cdot S_{\text{motif}}(K_i) \tag{21}$$

where $S_{deg}(K_i)$ denotes the degree-based consistency score, $S_{motif}(K_i)$ denotes the motif-based consistency score, and λ is a weighting parameter.

3.5. Cluster-Level Evidence-Based Prompt Generation for Large Language Models

Unlike conventional truth-inference tasks with clearly defined task settings, knowledge graph fusion involves a hybrid scenario in which single-truth and multi-truth facts coexist. A unified processing procedure without distinguishing between these two settings is therefore unlikely to handle both effectively. Large language models provide general-purpose external knowledge and can support a unified inference process through a two-stage structured reasoning procedure. In addition, guiding LLM reasoning with cluster-level evidence helps consolidate supporting information and reduces erroneous judgments caused by semantically similar facts.

To this end, we design a cluster-level evidence-based prompt generation (CEP) method. Truth selection from candidate clusters is performed through two reasoning stages: (1) relation-type inference and (2) cluster credibility inference. Relation cardinality provides a prior on truth multiplicity rather than an absolute constraint. Cluster credibility inference then evaluates candidate clusters using their aggregated evidence. Finally, the trustworthy cluster-level results are mapped back to their member facts, from which a representative can be selected when a concrete output is required.

3.5.1. Relation-Type Inference and Distinction

Under the same entity and relation, multiple candidate facts may be provided, and the expected truth multiplicity is related to the empirical cardinality of relation r. Relations with almost one tail per head usually favor a single-truth interpretation, whereas relations with multiple tails per head allow multiple valid truths. This signal is used only as prior guidance because a relation-level tendency does not determine the correct number of truths for every entity.

Before cluster-level truth inference, one relation-type request is generated for each distinct relation and its result is cached for all conflict groups involving that relation. The request uses relation-level statistics computed from the original source graphs, without web retrieval or entity/relation descriptions. The statistics are

$$\mathbf{G}_r = \left(n_r, n_r^{(h)}, \bar{n}_{t|h}, Q_{0.95}(n_{t|h}), \rho_{>1}, \bar{n}_{h|t}, h_r\right), \tag{22}$$

where n_r is the raw triple count, $n_r^{(h)}$ is the number of distinct heads, $\bar{n}_{t|h}$ is the average number of distinct tails per head, $Q_{0.95}(n_{t|h})$ is its 95th percentile, $\rho_{>1}$ is the fraction of heads

with multiple tails, and $\bar{n}_{h|t}$ is the average number of heads per tail. The deterministic hint h_r is set to *single* when $\rho_{>1} \leq 0.01$ and $\bar{n}_{t|h} \leq 1.05$, to *multi* when $\rho_{>1} \geq 0.05$ or $Q_{0.95}(n_{t|h}) > 1$, and to *uncertain* otherwise.

The relation-type inference prompt template is formulated as

$$Prompt_{r\text{-}type} = Concat(I_{r\text{-}type}, r, \mathbf{G}_r) \tag{23}$$

Here, $I_{r\text{-}type}$ denotes the relation-cardinality instruction. With this prompt template, the output of the large language model is defined as

$$Type(r) \in \{single, multi, uncertain\} \tag{24}$$

Here, *single* indicates a prior preference for one credible cluster, *multi* indicates that multiple clusters may be valid, and *uncertain* indicates that the relation statistics do not provide a clear prior. None of these outputs fixes the number of selected clusters.

3.5.2. Cluster-Level Evidence-Based Truth Inference

A structured prompting strategy based on the inferred relation prior is adopted. The task is treated as multi-truth by default, and $Type(r)$ guides rather than constrains the final decision. When $Type(r) = single$, the prompt emphasizes competition among clusters but still permits multiple selections if the candidate facts and evidence support them. When $Type(r) = multi$, multiple credible clusters may be retained. When $Type(r) = uncertain$, the decision relies more strongly on candidate-cluster semantics and cluster-level credibility evidence.

The single-truth prompt template can be formalized as

$$Prompt_{single} = Concat(I_{single}, e, r, \mathcal{K}_{e,r}, S). \tag{25}$$

Here, I_{single} denotes the instruction for the single-truth prior, e denotes the entity, r denotes the relation, and $\mathcal{K}_{e,r}$ denotes the set of candidate clusters for entity e under relation r. Each cluster is represented by its representative value, member values, and evidence. Specifically, $S = \{S(K_1), S(K_2), \ldots, S(K_m)\}$, where $S(K_i)$ denotes the evidence associated with the i-th candidate cluster.

Similarly, prompt templates can be constructed for multi-truth tasks and for cases in which the relation type is uncertain:

$$Prompt_{multi} = Concat(I_{multi}, e, r, \mathcal{K}_{e,r}, S) \tag{26}$$

$$Prompt_{uncer} = Concat(I_{uncer}, e, r, \mathcal{K}_{e,r}, S) \tag{27}$$

Finally, the constructed prompt guides the large language model to infer the true candidate clusters for the current entity–relation pair. The complete system and user prompt templates, required output formats, and LLM inference configurations used in the experiments are provided in Appendix A.

For candidate clusters determined to be true, a source-independent semantic representative selection strategy is used to obtain a concrete fact-level output. For each true cluster, the candidate fact with the highest average cosine similarity to the other facts in the same cluster, computed using the same BGE-M3 representations as in SFC, is selected as the representative fact. This selection is performed after cluster-level truth inference and does not affect the inferred truth label of the cluster. If the selected fact originates from the source knowledge graph, it is mapped to the entity and relation schema of the target knowledge graph before being output. The current experiments focus on the correctness of

truth inference rather than the quality of the final representative selection, which will be further investigated in future work.

4. Experiments

4.1. Experimental Setup

4.1.1. Dataset

Due to the lack of dedicated datasets containing factual conflicts for knowledge graph fusion, we construct a truth-inference dataset for conflicting facts in knowledge graph fusion based on the FR-EN subset of DBP15K [29,30] by controllably injecting erroneous facts. The two knowledge graphs after entity alignment are treated as simulated multi-source inputs for the fusion scenario. Since this setting still differs from real-world knowledge graph fusion environments, the constructed dataset should be regarded as a controlled evaluation benchmark for distinguishing injected erroneous facts from facts originally present in DBP15K, rather than as a fully verified benchmark of naturally occurring real-world KG-fusion conflicts.

DBP15K is a cross-lingual entity alignment dataset. In our experiments, the French and English knowledge graphs are regarded as two heterogeneous knowledge sources, and the FR-EN subset can therefore be viewed as a cross-lingual instance of multi-source knowledge graph fusion. We consider a task setting in which a source knowledge graph is fused into a target knowledge graph. The entity alignment results provided by the dataset are used as a prerequisite, and the entity alignment task itself is not evaluated in this study.

Original Positive Samples: As shown in Table 1, relation triples associated with the same entity are first aggregated according to the entity alignment results. If multiple candidate tail values exist for the same entity–relation pair, they are grouped into a candidate fact set. A total of 16,245 candidate facts are extracted in this manner. After removing exact duplicate facts, 16,214 facts remain, forming 5429 conflict groups. Since the original triples in DBP15K are derived from existing knowledge graphs, the 16,214 deduplicated candidate facts are treated as the original positive samples during dataset construction. This label only indicates that the facts originate from the original knowledge graphs and should not be interpreted as independent verification of their real-world veracity.

Table 1. Statistics of Original Positive Samples.

KG_{FR} Triples	KG_{EN} Triples	Multi-Truth Facts	Conflict Groups
105,998	115,722	16,214	5429

Erroneous Fact Injection: Following the noise construction settings used in knowledge graph error detection tasks, such as CAGED [31] and CCA [32], erroneous facts accounting for 5% of the final candidate facts are injected into the clean multi-truth dataset. To simulate both source-independent errors and cross-source correlated errors that may occur during knowledge graph fusion, two injection strategies are designed at a ratio of 7:3: (1) single-source independent error injection: following the entity replacement strategy commonly used in negative sampling [33], a fact is randomly selected from triples that share the same relation as the target candidate fact but have a different head entity, and is then injected into the corresponding candidate fact group; (2) cross-source correlated error injection for aligned entities: based on entity alignment mappings, two different surface forms of the same tail entity from the two knowledge graphs are selected and injected into a candidate fact group, respectively. During error injection, candidate tail entities that already appear as valid tail values for the target entity–relation pair, or that are aligned with existing valid tail entities in the candidate group, are excluded; duplicate injected facts are also removed. For

cross-source correlated error injection, the two surface forms must correspond to the same aligned tail entity, and this entity must be distinct from all original valid tail entities in the candidate group. In addition, all 180 injected erroneous facts in the test set were manually reviewed to exclude potentially ambiguous labels. The resulting controlled fusion dataset contains 5429 conflict groups and 17,067 candidate facts.

Dataset Split: As shown in Table 2, the multi-truth fusion dataset is divided into training, development, and test sets at a ratio of 7:1:2. To prevent label leakage, the test set is used exclusively for final model evaluation. Accordingly, the constructed dataset should be regarded as a controlled evaluation benchmark for distinguishing injected erroneous facts from facts originally present in DBP15K, rather than as a fully verified benchmark of naturally occurring real-world KG conflicts.

Table 2. Dataset Split Statistics.

Dataset Split	Fact Groups	Original Facts	Injected Errors	Total Facts
Training Set	3784	11,171	588	11,759
Development Set	532	1626	85	1711
Test Set	1113	3417	180	3597
Total	5429	16,214	853	17,067

4.1.2. Evaluation Metrics

This study adopts Precision, Recall, and F1-score to evaluate the performance of different methods in detecting erroneous facts. For clarity, erroneous facts are treated as the positive class in the error-detection evaluation; therefore, the corresponding metrics are denoted as Error Precision, Error Recall, and Error F1. Error Precision represents the proportion of truly erroneous facts among all facts predicted as erroneous by the model, whereas Error Recall represents the proportion of erroneous facts that are correctly identified among all ground-truth erroneous facts. Error F1 is defined as the harmonic mean of Error Precision and Error Recall. To additionally evaluate the preservation of true knowledge, we report the corresponding true-class metrics, including True Precision, True Recall, and True F1. The corresponding formulas are defined as follows:

$$P_{\text{err}} = \frac{TP}{TP + FP}, \quad R_{\text{err}} = \frac{TP}{TP + FN}, \quad F1_{\text{err}} = \frac{2P_{\text{err}} \cdot R_{\text{err}}}{P_{\text{err}} + R_{\text{err}}} \tag{28}$$

$$P_{\text{true}} = \frac{TN}{TN + FN}, \quad R_{\text{true}} = \frac{TN}{TN + FP}, \quad F1_{\text{true}} = \frac{2P_{\text{true}} \cdot R_{\text{true}}}{P_{\text{true}} + R_{\text{true}}} \tag{29}$$

where TP denotes the number of erroneous facts correctly identified as erroneous, FP denotes the number of true facts incorrectly classified as erroneous, FN denotes the number of erroneous facts that are not identified, and TN denotes the number of true facts that are correctly retained. Accordingly, R_{true} is the True Recall and corresponds to specificity under the erroneous-fact-positive convention.

4.1.3. Comparison Methods

To evaluate the effectiveness of the proposed method, four categories of baselines are selected for comparison: traditional truth-inference methods, knowledge graph fact-checking methods, pretrained language model-based methods, and large language model-based methods.

The traditional truth-inference baselines include **Voting**, which determines truth based on majority source support; **TruthFinder** [5], which iteratively updates source trustwor-

thiness and fact confidence; **LTM** [7], which employs a probabilistic graphical model for truth inference; **CATD** [9], which incorporates confidence estimation into source reliability assessment; and **CRH** [8], which is designed for conflict resolution in heterogeneous data. The knowledge graph fact-checking baselines include **CKRL** [34], which performs confidence-aware knowledge representation learning; **RotatE** [35], which models relations as rotations in complex vector space; **AnyBURL** [36], which relies on interpretable rule learning; and **PaTyBRED-RF** [12], which combines relational-path features with a random forest classifier. For pretrained language model-based methods, **StAR-BERT** [16], which integrates graph structure with textual semantics, and **SimKGC-BERT** [17], which applies contrastive learning to knowledge graph completion, are selected. Finally, the large language model baselines include **Qwen3.7-Max**, **DeepSeek-V4-Pro**, and **GPT-5.5** [22–24].

All comparison methods are evaluated using the same training, development, and test splits. Methods requiring supervised training or task-specific fine-tuning are trained exclusively on the training set, whereas inference-only or unsupervised methods are executed according to their original procedures without access to test labels. The development set is used for hyperparameter selection and, when a method produces a continuous confidence score, for selecting the decision threshold according to the F1-score. The test set is used only for final evaluation. The information provided to each category of baseline follows its original formulation. Traditional truth-inference methods receive the candidate facts and their source assignments; knowledge graph fact-checking methods use the candidate triples together with the graph information required by their original implementations; pretrained model-based methods receive the triple or textual inputs required by the corresponding models. For the direct large language model baselines, the models are provided only with the original candidate facts and a unified task instruction, without access to the SFC-generated clusters, CEM credibility evidence, or the proposed cluster-level evidence-based prompting strategy. For methods that directly produce discrete predictions, the decision rule defined by the original method is retained. The complete prompt template for the direct LLM baseline is provided in Appendix A.3

The main hyperparameters are determined on the development set through grid search. During the semantics-aware clustering stage, the high-similarity merging threshold τ_s is selected from $\{0.75, 0.80, 0.85, 0.90, 0.93, 0.95\}$, and the Cross-Encoder matching threshold τ_c is selected from $\{0.20, 0.40, 0.50, 0.60, 0.70, 0.80\}$. The NLI entailment threshold τ_e is selected from $\{0.001, 0.0027, 0.01, 0.10, 0.30, 0.60\}$, and the weighting parameter λ is selected from $\{0.00, 0.25, 0.50, 0.75, 1.00\}$. In addition, the 0.50, 0.60, 0.70, 0.80, 0.85, 0.90, and 0.95 quantiles of the corresponding development-set scores are included as candidate threshold values. The final values of τ_s, τ_c, τ_e, and λ are set to 0.90, 0.40, 0.0027, and 0.50, respectively. The final configuration is selected according to the Error F1 on the development set and is then fixed for test-set evaluation; the test set is not used for hyperparameter or threshold selection. Detailed LLM inference settings and complete prompt templates are provided in Appendix A.

4.2. Experimental Results and Analysis

4.2.1. Overall Experimental Results

The main experimental results are presented in Table 3. As shown in the table, the proposed method achieves performance improvements across all three large language models. In particular, Ours (GPT-5.5) achieves the best error-detection performance in the table, with an Error Precision of 81.0% and an Error F1 of 75.7%. It also maintains strong true-fact preservation performance, achieving a True Precision of 98.5%, a True Recall of 99.1%, and a True F1 of 98.8%. Compared with the strongest direct LLM baseline, GPT-5.5, the proposed method improves Error Precision and Error F1 by 10.5 and 6.0 percentage

points, respectively, while True Recall increases from 98.5% to 99.1%. These results indicate that the proposed method improves erroneous-fact detection while maintaining high preservation performance for true facts.

Therefore, the following conclusions can be drawn:

1. Traditional truth-inference methods generally perform poorly. Among them, CRH achieves the highest F1-score in this category, reaching 18.6%. Although LTM and CATD obtain relatively high Recall, their Precision values are only 5.47% and 8.03%, respectively, accompanied by substantial decreases in True Recall. This indicates that these methods tend to incorrectly classify a large number of true facts as erroneous, and that relying solely on source reliability is insufficient for knowledge graph fusion scenarios.
2. Knowledge graph fact-checking methods and pretrained model-based methods perform substantially better than traditional truth-inference approaches. PaTyBRED-RF achieves an F1-score of 58.1%, while SimKGC-BERT further improves it to 63.5%, suggesting that relational paths, graph structure, and textual semantic information are beneficial for identifying erroneous facts in knowledge graphs.
3. Large language models demonstrate strong factual judgment capabilities. Among them, GPT-5.5 achieves an F1-score of 69.7%, outperforming DeepSeek-V4-Pro and Qwen3.7-Max by 12.3 and 5.6 percentage points, respectively, while maintaining a True Recall of 98.5%.

The proposed method further improves the Error F1 of all three large language models. Compared with their corresponding direct LLM baselines, Ours (DeepSeek-V4-Pro), Ours (Qwen3.7-Max), and Ours (GPT-5.5) improve Error F1 by 5.7, 1.1, and 6.0 percentage points, respectively. For Qwen3.7-Max, Error Recall increases substantially from 65.0% to 73.3%, whereas Error Precision decreases from 63.2% to 58.7% and True Recall decreases from 98.0% to 97.3%. This indicates that the Qwen-based variant shifts toward higher error-detection recall, identifying more erroneous facts at the cost of additional false positives and a slight reduction in true-fact preservation. Therefore, its F1 improvement reflects a precision–recall trade-off rather than a uniform improvement across all metrics. In contrast, Ours (GPT-5.5) improves Error Precision from 70.5% to 81.0% while maintaining strong True Recall.

4.2.2. Robustness Under Different Error Rates

To further examine the robustness of CETI-KGF under different noise levels, we conduct additional experiments with injected-error rates of 5%, 10%, and 15%. The original 5% setting is retained as the reference setting, while two additional settings with higher proportions of erroneous facts are introduced. Across the three settings, the same data construction procedure, error-injection strategies, and evaluation protocol are maintained, with only the injected-error rate being varied.

For this robustness analysis, we compare CETI-KGF with the direct GPT-5.5 baseline using the same underlying large language model. This controlled comparison is intended to examine whether the performance advantage introduced by semantic clustering, cluster-level credibility evidence, and structured cluster-level inference can be maintained when the proportion of erroneous candidate facts changes.

As shown in Figure 2, CETI-KGF consistently outperforms the direct GPT-5.5 baseline under all three injected-error settings. At error rates of 5%, 10%, and 15%, CETI-KGF achieves F1-scores of 75.7%, 75.2%, and 74.5%, respectively, compared with 69.7%, 70.3%, and 69.9% for direct GPT-5.5. The corresponding F1 improvements are 6.0, 4.9, and 4.6 percentage points.

Table 3. Comparison of Error-Class and True-Class Performance Across Different Models.

Category	Model	Error Precision (%)	Error Recall (%)	Error F1 (%)	True Precision (%)	True Recall (%)	True F1 (%)
Traditional Truth Inference	Voting	5.56	1.67	2.56	95.0	98.5	96.7
	TruthFinder	5.91	6.67	6.27	95.1	94.4	94.7
	LTM	5.47	73.3	10.2	95.9	33.2	49.4
	CATD	8.03	56.7	14.1	96.6	65.8	78.3
	CRH	15.0	24.4	18.6	95.9	92.7	94.3
Fact Checking	CKRL	7.72	32.8	12.5	95.7	79.4	86.8
	RotatE	29.8	62.2	40.3	97.9	92.3	95.0
	AnyBURL	32.4	70.0	44.3	98.3	92.3	95.2
	PaTyBRED-RF	68.4	50.6	58.1	97.4	98.8	98.1
Pretrained Models	StAR-BERT	41.2	38.9	40.0	96.8	97.1	96.9
	SimKGC-BERT	67.5	60.0	63.5	97.9	98.5	98.2
Large Language Models	DeepSeek-V4-Pro	49.2	68.9	57.4	98.3	96.3	97.3
	Qwen3.7-Max	63.2	65.0	64.1	98.2	98.0	98.1
	GPT-5.5	70.5	68.9	69.7	98.4	98.5	98.4
Proposed Method	Ours (DeepSeek-V4-Pro)	58.6	68.3	63.1	98.3	97.5	97.9
	Ours (Qwen3.7-Max)	58.7	**73.3**	65.2	**98.6**	97.3	97.9
	Ours (GPT-5.5)	**81.0**	71.1	**75.7**	98.5	**99.1**	**98.8**

The bold and underline formatting in the table cells are intentional: bold indicates the best (highest) performance metric, and underline indicates the second best metric in each row/column, which is a standard convention for experimental result tables.

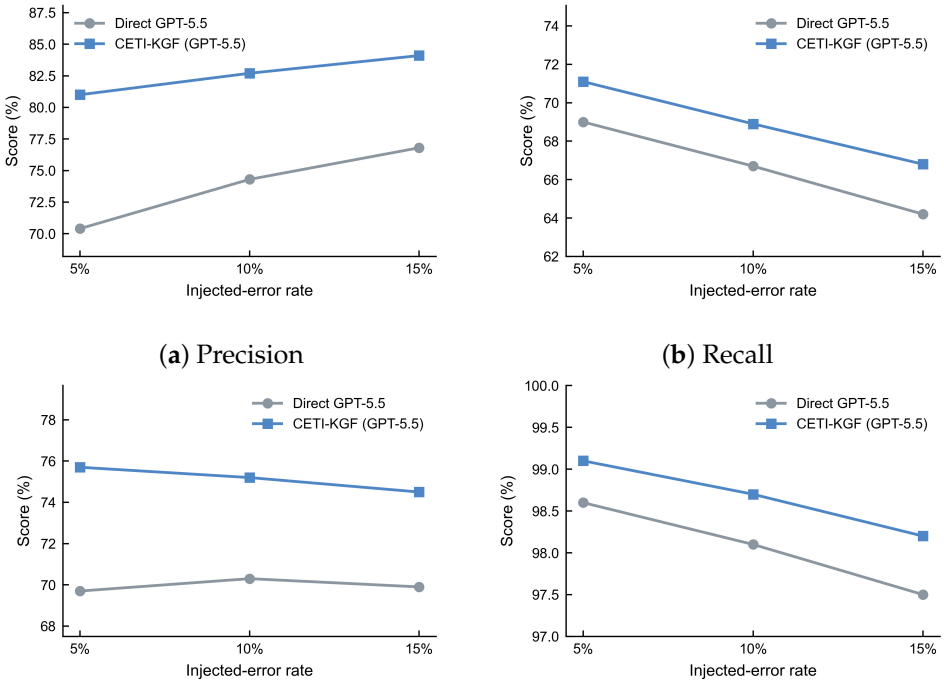


Figure 2. Robustness under Different Injected-Error Rates.

As the injected-error rate increases, the Precision of both methods increases, whereas Recall gradually decreases. Since erroneous facts are treated as the positive class in the current evaluation, the increase in their prevalence can partly contribute to higher Precision and should not be interpreted as an improvement in model capability. More importantly, CETI-KGF maintains relatively stable F1 performance and consistently higher True Recall than the direct GPT-5.5 baseline across all three settings. These results indicate that the performance advantage of CETI-KGF is preserved within the evaluated range of injected-error rates, although higher noise levels introduce additional difficulty in identifying all erroneous facts.

4.2.3. Performance Analysis Under Single-Truth and Multi-Truth Settings

To further examine the performance of CETI-KGF under different truth settings, we divide the 1113 test fact groups according to the number of distinct valid facts contained in each group. Among them, 356 groups contain only one distinct valid fact and are categorized as single-truth groups, whereas the remaining 757 groups contain multiple distinct valid facts and are categorized as multi-truth groups. Figure 3 reports the performance of Ours (GPT-5.5) separately under the two settings. As shown in Figure 3, the proposed method achieves an F1-score of 78.8% on single-truth groups and 74.4% on multi-truth groups. The performance on multi-truth groups is relatively lower, reflecting the greater difficulty of identifying multiple valid facts within the same conflict group. Nevertheless, the method maintains comparable performance in both settings, with True Recall remaining above 99.0%, indicating that the proposed framework can accommodate both single-truth and multi-truth cases in the current evaluation.

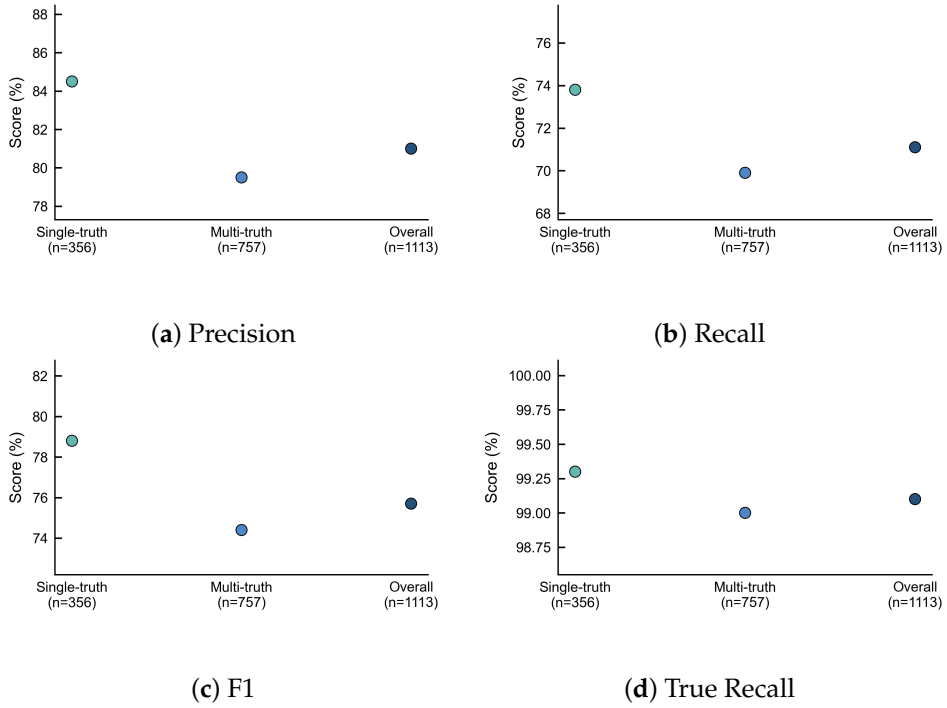


Figure 3. Performance on Single-Truth and Multi-Truth Groups.

Since relation-type inference provides prior guidance for the subsequent cluster-level truth inference, we further evaluate this stage independently. The ground-truth relation setting of each test group is determined according to the number of distinct valid facts: groups containing one valid fact are labeled as single-truth, whereas groups containing multiple valid facts are labeled as multi-truth. The inferred relation type is then compared with these reference labels.

As shown in Table 4, the relation-type inference stage correctly classifies 930 of the 1113 test groups, corresponding to an overall accuracy of 83.6%. The F1-scores for single-truth and multi-truth groups are 76.6% and 88.5%, respectively, with a macro F1-score of 82.6%. In addition, 30 groups are classified as uncertain, corresponding to an uncertain rate of 2.7%. These results indicate that relation-type inference provides useful prior guidance, while single-truth groups remain more difficult to identify than multi-truth groups.

Table 4. Relation-Type Inference Results.

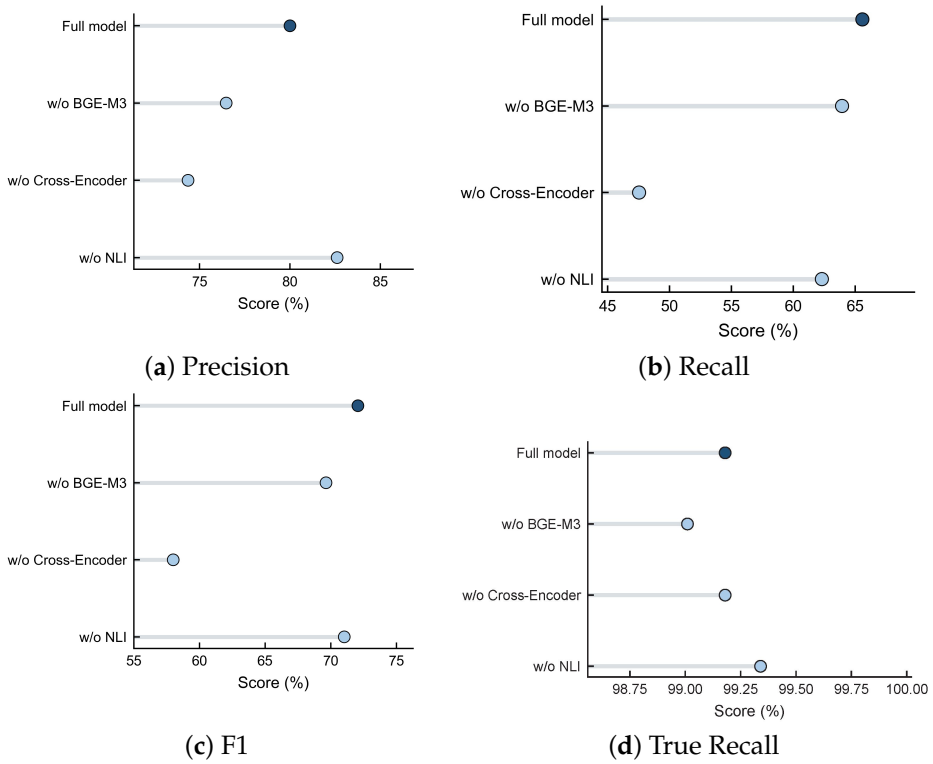
Setting	Groups	Correct	Precision (%)	Recall (%)	F1 (%)
Single-truth	356	270	77.4	75.8	76.6
Multi-truth	757	660	89.9	87.2	88.5
Overall (Macro)	1113	930	83.6	81.5	82.6

We further analyze the propagation of relation-type errors to the final truth-inference stage. For groups whose relation types are correctly inferred, the final truth-inference F1-score is 77.4%, whereas it decreases to 66.9% for groups with incorrect or uncertain relation-type predictions, a reduction of 10.5 percentage points. This result indicates that relation-type errors can affect downstream truth inference. Therefore, the inferred relation type is used as prior guidance rather than an absolute constraint in CETI-KGF, and the final selection of true clusters is jointly determined by the candidate-cluster information and cluster-level credibility evidence.

4.2.4. Fine-Grained Ablation Study of Semantics-Aware Fact Clustering

To further investigate the roles of different semantic models in the semantics-aware fact clustering module, a fine-grained ablation study is conducted. Specifically, BGE-M3, the Cross-Encoder, and NLI are removed individually, and fact merging is performed solely according to the similarity or entailment scores produced by the remaining models. This yields three variants: w/o BGE-M3, w/o Cross-Encoder, and w/o NLI. The experiment uses the same deterministic, stratified one-third subset as the direct clustering evaluation, comprising 371 test groups, 1279 candidate facts, and 61 injected erroneous facts. Precision, Recall, F1, and True Recall are micro-aggregated over all facts in this fixed subset.

As shown in Figure 4, the full model obtains a Precision of 80.00%, a Recall of 65.57%, and an F1-score of 72.07% on this subset. Removing BGE-M3 decreases the F1-score to 69.64%. Removing the Cross-Encoder produces the largest reduction, with Recall and F1 decreasing to 47.54% and 58.00%, respectively, which indicates that fine-grained pairwise matching is important for preventing erroneous cluster assignments. Removing NLI yields an F1-score of 71.03%, indicating that bidirectional entailment provides complementary evidence. These values are calculated directly from integer confusion counts for the 61 injected erroneous facts in the subset. For reproducibility, the corresponding (TP, FP, FN, TN) counts are $(40, 10, 21, 1208)$ for the full model, $(39, 12, 22, 1206)$ for w/o BGE-M3, $(29, 10, 32, 1208)$ for w/o Cross-Encoder, and $(38, 8, 23, 1210)$ for w/o NLI.



(**a**) Precision (**b**) Recall

(**c**) F1 (**d**) True Recall

Figure 4. Fine-Grained Ablation Results of Semantics-Aware Fact Clustering on the Fixed One-Third Test Subset.

4.2.5. Efficiency and Scalability Analysis

We further analyze the computational efficiency of CETI-KGF. Table 5 compares CETI-KGF with the corresponding direct GPT-5.5 baseline on the 1113 test conflict groups. Direct GPT-5.5 requires an average of 0.86 s per group, whereas CETI-KGF requires 2.03 s per group, corresponding to approximately 2.36× the recorded end-to-end runtime. Relation-type inference is cached by relation rather than repeated for every group. If N_g and N_r

denote the numbers of test groups and distinct test relations, respectively, the two inference stages require

$$N_{\mathrm{LLM}} = N_g + N_r, \qquad \overline{N}_{\mathrm{LLM}} = 1 + \frac{N_r}{N_g}. \tag{30}$$

For the current test set, $N_g = 1113$ and $N_r = 70$, giving 1183 requests, or approximately 1.063 requests per group, before the optional post-processing step for representative output. The additional runtime mainly arises from semantic scoring, evidence construction, and the amortized relation-type reasoning stage.

Let N denote the number of candidate facts in a conflict group. Since each candidate may be compared with members of the currently constructed clusters, the worst-case computational complexity of SFC is $O(N^2)$, with $O(N)$ storage for the cluster assignments. In practice, the more expensive Cross-Encoder and NLI models are invoked only when the first-stage BGE-M3 criterion cannot determine a merging decision, thereby reducing the number of expensive semantic comparisons. The local semantic models were executed on a server equipped with a six-core Intel Xeon Gold 6142 CPU, 60.9 GB of RAM, and an NVIDIA GeForce RTX 3090 GPU with 24 GB of memory, while GPT-5.5 was accessed through its API.

Table 5. Efficiency Comparison with the Direct LLM Baseline.

Method	Avg. Time (s/Group)	Total Time (min)	Relative Runtime
Direct GPT-5.5	0.86	15.9	1.00×
CETI-KGF (GPT-5.5)	2.03	37.7	2.36×

Although CETI-KGF introduces additional computational overhead, conflict groups are mutually independent and can be processed in parallel. Nevertheless, the current experiments do not constitute a large-scale industrial deployment. For substantially larger knowledge graphs, batched semantic encoding, caching, and parallel conflict-group processing can be further adopted to improve efficiency.

4.2.6. Direct Evaluation of Semantics-Aware Fact Clustering

To directly evaluate the clustering quality of SFC, we select a deterministic one-third subset of the test groups, stratified by whether a group contains an injected error and using seed 20260821. This protocol yields 371 conflict groups and 1279 candidate facts. We then construct the reference cluster partition of each sampled group using the canonical tail identifiers and entity-alignment mappings retained during dataset construction. Candidate facts mapped to the same canonical tail entity are assigned to the same reference cluster, while those mapped to different canonical tail entities are assigned to different clusters. This procedure yields 1079 reference clusters, with an average of 2.91 reference clusters per group, whereas the predicted partition contains 1081 clusters. The reference partition is therefore a reproducible identity- and alignment-based benchmark rather than an independently human-annotated semantic gold standard. Pairwise metrics are adopted because they directly reflect incorrect merging and splitting decisions between candidate facts. As shown in Table 6, SFC achieves a pairwise Precision of 54.66%, a pairwise Recall of 64.50%, and a pairwise F1-score of 59.17%. At the fact level, the B-cubed Precision, Recall, and F1-score are 94.40%, 94.45%, and 94.42%, respectively, and 257 of the 371 groups are clustered exactly, corresponding to an exact-group rate of 69.27%. The difference between the pairwise and B-cubed results reflects the large number of singleton reference clusters

and the concentration of pairwise errors in a small number of larger multi-value groups. Further inspection shows that incorrect merges mainly occur when candidate facts have highly similar surface forms or overlapping semantics but express different factual values. Incorrect splits mainly occur when equivalent facts differ substantially in cross-lingual expression, semantic granularity, aliases, or information completeness. No predicted cluster contains both an injected erroneous fact and an original fact in this subset. Therefore, the observed clustering errors mainly affect the separation of semantically related original facts and may weaken subsequent evidence aggregation, although they do not directly mix injected errors with original facts in the current evaluation.

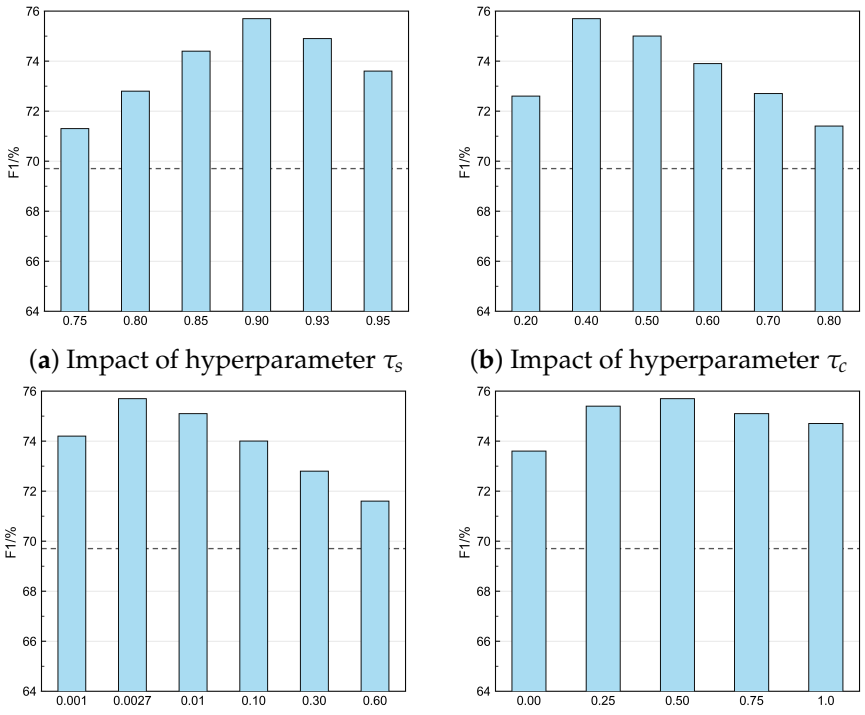
Table 6. Performance metrics of SFC on the direct evaluation set.

Method	Groups	Reference Clusters	P_{pair} (%)	R_{pair} (%)	$F1_{\text{pair}}$ (%)	$F1_{B^3}$ (%)
SFC	371	1079	54.66	64.50	59.17	94.42

4.2.7. Hyperparameter Analysis

This section investigates the effects of four key hyperparameters in the proposed framework, with the corresponding results reported in Figure 5.

For the high-similarity merging threshold τ_s, as its value increases from 0.75 to 0.90, the F1-score improves from 71.3% to 75.7%. When the threshold is further increased to 0.95, the F1-score decreases to 73.6%. This is because a smaller τ_s relaxes the fact-merging criterion, causing some semantically similar but non-equivalent facts to be incorrectly grouped into the same cluster. In contrast, an excessively large threshold may separate semantically equivalent facts into different clusters, resulting in fragmented intra-cluster evidence. Therefore, $\tau_s = 0.90$ provides a better trade-off between erroneous merging and over-clustering.



(**a**) Impact of hyperparameter τ_s (**b**) Impact of hyperparameter τ_c

(**c**) Impact of hyperparameter τ_e (**d**) Impact of hyperparameter λ

Figure 5. Hyperparameter sensitivity analysis results.

For the Cross-Encoder matching threshold τ_c, the model achieves the highest F1-score of 75.7% when $\tau_c = 0.40$. When the threshold is lower than this value, the overly permissive matching criterion introduces unreliable facts. As the threshold increases further, some semantically equivalent facts with substantial differences in surface form fail to satisfy the matching criterion, causing the F1-score to gradually decline. For the NLI entailment threshold τ_e, the best performance on the development set is obtained at $\tau_e = 0.0027$. Although this value is relatively small when considered independently, the NLI score is used only as an auxiliary semantic constraint rather than as the sole basis for clustering. A candidate can be merged only when the Cross-Encoder score also satisfies $\tau_c = 0.40$ and both directional entailment scores satisfy τ_e. A lower threshold may introduce facts with only weak semantic relevance, whereas a higher threshold may filter out valid entailment relations and lead to increasingly fragmented clustering results. The weighting parameter λ achieves the best performance at 0.50, indicating that both node degree and motif information contribute to the construction of credibility evidence.

Overall, all four hyperparameters exhibit relatively stable performance trends within reasonable ranges, and the F1-scores under all tested settings remain higher than that of the GPT-5.5 baseline. Based on the experimental results, τ_s, τ_c, τ_e, and λ are finally set to 0.90, 0.40, 0.0027, and 0.50, respectively.

4.2.8. Statistical Uncertainty and Significance Analysis

To assess the statistical uncertainty of the reported improvements, we perform a paired bootstrap analysis on the test set. Since candidate facts within the same conflict group are not independent, bootstrap resampling is conducted at the conflict-group level rather than at the individual-fact level. In each bootstrap iteration, 1113 conflict groups are sampled with replacement from the original test set, and Precision, Recall, F1-score, and True Recall are recalculated using all candidate facts contained in the sampled groups. This procedure is repeated 1000 times, and the 2.5th and 97.5th percentiles of the resulting distributions are used to construct 95% confidence intervals. The paired F1 difference between CETI-KGF and the corresponding direct LLM baseline is also calculated in each bootstrap iteration to assess the statistical reliability of the observed improvement.

Figure 6 reports the bootstrap confidence intervals for the direct GPT-5.5 baseline and Ours (GPT-5.5). Ours (GPT-5.5) achieves an F1-score of 75.7% with a 95% confidence interval of [70.8%, 80.1%], whereas the direct GPT-5.5 baseline achieves 69.7% with a confidence interval of [64.7%, 74.4%]. The paired bootstrap difference in F1 is 6.0 percentage points, with a 95% confidence interval of [2.1, 9.8] percentage points. Since the confidence interval of the paired difference does not include zero, the observed F1 improvement remains positive across the bootstrap resamples.

Because the LLM inference temperature was fixed at 0 and each test instance was evaluated once, the present analysis focuses on uncertainty induced by variation in the test sample rather than repeated stochastic decoding. The bootstrap results indicate that the performance advantage of CETI-KGF over direct GPT-5.5 is stable under resampling of the current test set.

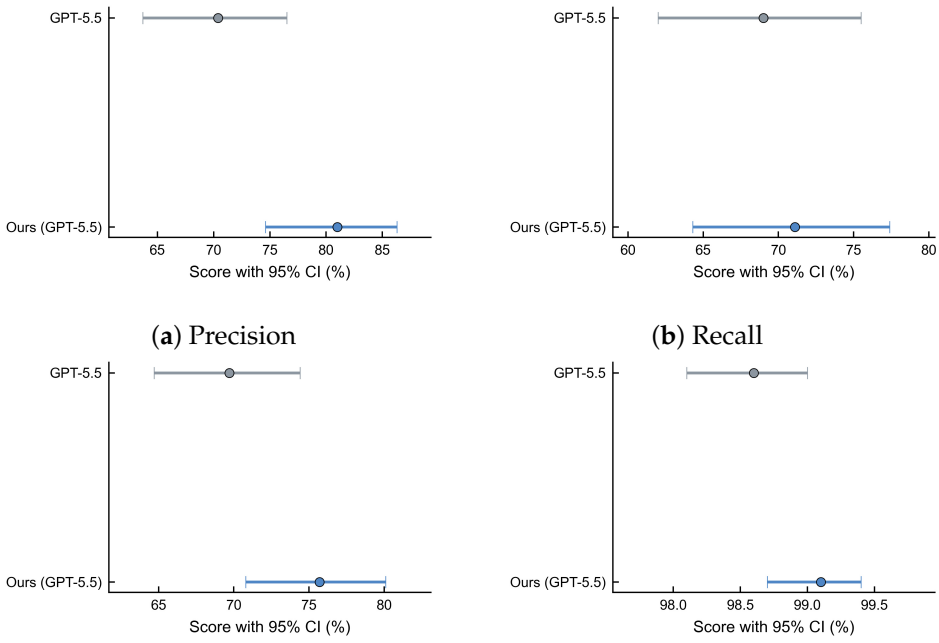


Figure 6. Bootstrap 95% Confidence Intervals on the Test Set.

4.2.9. Case Study

To illustrate the interpretability of the proposed method, we present a representative example in Tables 7 and 8. For the aligned entity Campestre do Menino Deus, five facts from different knowledge graphs are associated with the relation northwest. One example is (Campestre do Menino Deus, northwest, Our Lady of Perpetual Help), where the tail entity Our Lady of Perpetual Help represents one candidate fact provided by the triple. The fifth fact is an artificially injected erroneous fact.

First, the clustering module partitions facts from different sources into three clusters. Our Lady of Perpetual Help and Nossa Senhora do Perpétuo Socorro are the English and Portuguese expressions of the same location, respectively, and are therefore assigned to K_1. Santo Antao (Santa Maria) and Santo Antão, Santa Maria are semantically similar and differ only in formatting, so they are grouped into K_2. The remaining fact, Juscelino Kubitschek, Santa Maria, represents an independent fact and is therefore assigned to a separate cluster, K_3. After clustering, the large language model performs truth inference at the cluster level based on the evidence within each cluster. Since K_1 and K_2 are supported by stronger evidence, they are labeled as true, whereas K_3 is labeled as false. The resulting cluster-level labels are then propagated to every fact within the corresponding clusters.

In addition, after cluster-level truth inference, the inferred labels are mapped back to the corresponding member facts. When a concrete output representation is required, one representative fact can be selected from each cluster determined to be true. This step is used only for output construction and is not independently evaluated in the current experiments.

Table 7. Case Study Workflow (a).

Input Facts	Cluster	Cluster Members
Our Lady of Perpetual Help Nossa Senhora do Perpétuo Socorro	K_1	Our Lady of Perpetual Help Nossa Senhora do Perpétuo Socorro
Santo Antao (Santa Maria) Santo Antão, Santa Maria	K_2	Santo Antao (Santa Maria) Santo Antão, Santa Maria
Juscelino Kubitschek, Santa Maria	K_3	Juscelino Kubitschek, Santa Maria

Table 8. Case Study Workflow (b).

Model	Cluster	Fact	Representative Output
CETI-KGF (GPT-5.5)	K_1 ✓	Our Lady of Perpetual Help ✓ Nossa Senhora do Perpétuo Socorro ✓	Our Lady of Perpetual Help ✓ Nossa Senhora do Perpétuo Socorro ×
	K_2 ✓	Santo Antao (Santa Maria) ✓ Santo Antão, Santa Maria ✓	Santo Antao (Santa Maria) ✓ Santo Antão, Santa Maria ×
	K_3 ×	Juscelino Kubitschek, Santa Maria ×	Juscelino Kubitschek, Santa Maria ×

The symbols "✓" and "×" are used in the comparison table to indicate whether each method supports a given feature/capability: "✓" means supported, and "×" means not supported.

5. Limitations

The current evaluation has several limitations regarding dataset construction and ground-truth validity. The original facts extracted from DBP15K are treated as benchmark-positive samples because they are present in the source knowledge graphs; however, their real-world veracity has not been independently verified. The erroneous facts used as negative samples are synthetically injected under controlled rules. Although filtering procedures and manual verification of the injected test errors are employed to reduce ambiguous labels, this does not establish the real-world truthfulness of all original DBP15K facts.

Therefore, the current experiments should be interpreted primarily as a controlled evaluation of whether CETI-KGF can distinguish synthetically injected erroneous facts from facts originally present in DBP15K, rather than as a direct evaluation of truth inference over naturally occurring real-world KG-fusion conflicts. The current study is also restricted to the FR-EN language pair and does not include an additional language pair or a realistic heterogeneous fusion dataset. Validation on manually verified conflict sets, additional language pairs, and naturally occurring fusion conflicts remains necessary to further assess the generalizability of the proposed method. Future work will construct or adopt such datasets and evaluate CETI-KGF under more realistic knowledge graph fusion settings.

6. Conclusions

To address factual conflicts that arise in multi-source knowledge graphs after entity alignment, this paper proposes CETI-KGF, a cluster-level evidence-enhanced truth-inference method for knowledge graph fusion. Semantics-aware clustering first organizes weak-conflict fact variants into semantic groups. Cluster-level evidence is then constructed from source support, intra-cluster support strength, and local contextual consistency, and is used to guide a large language model in performing truth inference across candidate clusters. The inferred cluster-level results are subsequently mapped back to the corresponding facts. On the constructed DBP15K-based benchmark, CETI-KGF (GPT-5.5) achieves the highest Error Precision and Error F1 among the compared methods, reaching 81.0% and 75.7%, respectively.

In summary, the proposed method provides a cluster-level truth-inference framework for knowledge graph fusion by organizing semantically related facts and aggregating their supporting evidence before LLM-based reasoning. The current experiments primarily evaluate the correctness of truth inference under the controlled DBP15K-based setting. Future work will further investigate more realistic multi-source fusion scenarios and

extend the evaluation to downstream fused-KG quality, including representative selection, redundancy resolution, consistency, and completeness.

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

Funding: This research was funded by the Shandong Provincial Key Research and Development Program, grant number 2025CXPT104, and the Fundamental Research Funds for the Central University under grant No.26CX05052A.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The DBP15K dataset analyzed in this study is publicly available at https://github.com/nju-websoft/JAPE/raw/master/data/dbp15k.tar.gz (accessed on 23 September 2026).

Acknowledgments: The authors thank the anonymous reviewers for their constructive comments.

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

Abbreviations

The following abbreviations are used in this manuscript:

KG	Knowledge graph
KGF	Knowledge graph fusion
EA	Entity alignment
TI	Truth inference
LLM	Large language model
SFC	Semantic-aware fact clustering
CEM	Cluster-level credibility evidence modeling
CEP	Cluster-level evidence-based prompt generation
NLI	Natural language inference

Appendix A. LLM Inference Configuration and Prompt Templates

Appendix A.1. LLM Inference Settings

All LLMs were accessed through their respective application programming interfaces. The model identifiers used in the comparison experiments were DeepSeek-V4-Pro, Qwen3.7-Max (qwen3.7-max-2026-06-08), and GPT-5.5. The temperature was set to 0 and reasoning mode was disabled for the direct LLM baselines and the corresponding CETI-KGF variants. Each conflict group was evaluated once. For the GPT-5.5-based CETI-KGF experiments, the maximum completion length was set to 8192 tokens to accommodate the structured JSON output. The top_p and random seed parameters were not explicitly specified, and the default values of the corresponding APIs were used. Failed requests caused by network errors or invalid output formats were retried, but only the first valid response was retained; no majority voting or self-consistency aggregation was applied.

Appendix A.2. Prompt Design

CETI-KGF employs a two-stage structured prompting strategy. In the first stage, the LLM determines whether a relation follows a single-truth, multi-truth, or uncertain setting based on its semantic meaning and graph-level cardinality statistics. The inferred relation

type is cached and reused for all conflict groups associated with the same relation. In the second stage, the LLM receives the entity, relation, inferred relation type, and cluster-level evidence, and then selects the candidate clusters considered true. The exact prompt templates are provided below.

Appendix A.2.1. Stage 1: Relation-Type Inference Prompt

System Prompt

```
You classify knowledge-graph relation cardinality for fusion. Use the
supplied graph statistics and relation meaning.
Return JSON only with keys relation_type (single, multi, or uncertain),
confidence (0 to 1), and rationale (one short sentence).
```

User Prompt

```
Determine the relation type from this evidence:

{
  "relation": "{relation}",
  "raw_triple_count": {raw_triple_count},
  "distinct_head_count": {distinct_head_count},
  "avg_tails_per_head": {avg_tails_per_head},
  "p95_tails_per_head": {p95_tails_per_head},
  "fraction_heads_with_multiple_tails":
    {fraction_heads_with_multiple_tails},
  "avg_heads_per_tail": {avg_heads_per_tail},
  "statistical_hint": "{statistical_hint}"
}

Keep the rationale under 20 words.
```

Required Output

```
{
  "relation_type": "single | multi | uncertain",
  "confidence": 0.0,
  "rationale": "..."
}
```

Appendix A.2.2. Stage 2: Cluster-Level Truth-Inference Prompt

System Prompt

```
You perform multi-truth inference over semantic fact clusters in a
fused knowledge graph. Select every cluster that can be true for the
stated entity and relation. Evidence values are support signals, not
labels. Return JSON only with keys selected_cluster_ids, confidence,
and rationale.
```

User Prompt

```
For entity "{entity}" and relation "{relation}", the relation type is
"{relation_type}". Select all true semantic clusters. A single relation
normally permits one selected cluster; a multi relation can permit
multiple clusters; an uncertain relation requires cautious semantic
judgment.
```

```
Treat the task as multi-truth by default. The inferred relation type is
guidance rather than an absolute constraint. Every conflict group
contains at least one true cluster. Evidence scores are normalized
support signals rather than ground-truth labels. Never return an empty
selected_cluster_ids list. If the available factual knowledge is
inconclusive, select at least the best-supported candidate cluster.
Return the exact cluster IDs provided in the input and keep the
rationale under 20 words.

Input:
{
  "entity": "{entity}",
  "relation": "{relation}",
  "relation_type": "{relation_type}",
  "relation_type_confidence": {relation_type_confidence},
  "candidate_clusters": [
    {
      "cluster_id": "{cluster_id}",
      "representative_tail": "{representative_tail}",
      "member_tails": ["{tail_1}", "{tail_2}", "..."],
      "source_support": {source_support},
      "cluster_size": {cluster_size},
      "degree_score": {degree_score},
      "motif_score": {motif_score},
      "local_consistency_score": {local_consistency_score}
    }
  ]
}
```

Required Output

```
{
  "selected_cluster_ids": ["{cluster_id_1}", "{cluster_id_2}"],
  "confidence": 0.0,
  "rationale": "..."
}
```

Appendix A.3. Direct LLM Baseline Prompt

System Prompt

```
You perform truth inference for candidate facts in knowledge graph fusion.
Given an entity, a relation, and a set of raw candidate facts from multiple
source knowledge graphs, select all candidate facts that can be true.
Use the provided candidate facts and your own parametric knowledge only.
Return JSON only with keys selected_fact_ids, confidence, and rationale.
```

User Prompt

```
For the following entity and relation, determine which candidate facts are true.

Entity: {entity}
Relation: {relation}

Candidate facts:
[
  {"fact_id": "f1", "source": "KG1", "tail": "{tail_1}"},
```

```
  {"fact_id": "f2", "source": "KG2", "tail": "{tail_2}"},
  {"fact_id": "f3", "source": "KG1", "tail": "{tail_3}"},
  ...
]

Task requirements:
1. Select all fact IDs that correspond to factually true candidate facts.
2. More than one fact may be true.
3. Do not use any precomputed semantic clusters or evidence scores.
4. Return the exact fact IDs provided in the input.
5. Keep the rationale under 20 words.

Required JSON format:
{
  "selected_fact_ids": ["f1", "f3"],
  "confidence": 0.0,
  "rationale": "..."
}
```

References

1. Pan, S.; Luo, L.; Wang, Y.; Chen, C.; Wang, J.; Wu, X. Unifying Large Language Models and Knowledge Graphs: A Roadmap. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 3580–3599. [CrossRef]
2. Cong, S.; Su, G.; Liu, L. Survey on Knowledge Graph Entity Alignment: From Traditional Methods to Frontier Technologies. *Comput. Eng. Appl.* **2026**, *62*, 47–67.
3. Marchesin, S.; Silvello, G. Efficient and Reliable Estimation of Knowledge Graph Accuracy. *Proc. VLDB Endow.* **2024**, *17*, 2392–2404. [CrossRef]
4. Li, Y.; Gao, J.; Meng, C.; Li, Q.; Su, L.; Zhao, B.; Fan, W.; Han, J. A Survey on Truth Discovery. *ACM SIGKDD Explor. Newsl.* **2016**, *17*, 1–16. [CrossRef]
5. Yin, X.; Han, J.; Yu, P.S. Truth Discovery with Multiple Conflicting Information Providers on the Web. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2007; pp. 1048–1052. [CrossRef]
6. Dong, X.L.; Berti-Equille, L.; Srivastava, D. Integrating Conflicting Data: The Role of Source Dependence. *Proc. VLDB Endow.* **2009**, *2*, 550–561. [CrossRef]
7. Zhao, B.; Rubinstein, B.I.P.; Gemmell, J.; Han, J. A Bayesian Approach to Discovering Truth from Conflicting Sources for Data Integration. *Proc. VLDB Endow.* **2012**, *5*, 550–561. [CrossRef]
8. Li, Q.; Li, Y.; Gao, J.; Zhao, B.; Fan, W.; Han, J. Resolving Conflicts in Heterogeneous Data by Truth Discovery and Source Reliability Estimation. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*; ACM: New York, NY, USA, 2014; pp. 1187–1198. [CrossRef]
9. Li, Q.; Li, Y.; Gao, J.; Zhao, B.; Fan, W.; Han, J. A Confidence-Aware Approach for Truth Discovery on Long-Tail Data. *Proc. VLDB Endow.* **2014**, *8*, 425–436. [CrossRef]
10. Lao, N.; Cohen, W.W. Relational Retrieval Using a Combination of Path-Constrained Random Walks. *Mach. Learn.* **2010**, *81*, 53–67. [CrossRef]
11. Shiralkar, P.; Flammini, A.; Menczer, F.; Ciampaglia, G.L. Finding Streams in Knowledge Graphs to Support Fact Checking. In *Proceedings of the 2017 IEEE International Conference on Data Mining*; IEEE: Piscataway, NJ, USA, 2017; pp. 859–864. [CrossRef]
12. Melo, A.; Paulheim, H. Detection of Relation Assertion Errors in Knowledge Graphs. In *Proceedings of the Knowledge Capture Conference*; ACM: New York, NY, USA, 2017; pp. 1–8. [CrossRef]
13. Syed, Z.H.; Roder, M.; Ngonga Ngomo, A.C. Unsupervised Discovery of Corroborative Paths for Fact Validation. In *Proceedings of the Semantic Web–ISWC 2019*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 630–646. Available online: https://link.springer.com/chapter/10.1007/978-3-030-30793-6_36 (accessed on 3 November 2026).
14. Tan, F.A.; Desai, J.; Sengamedu, S.H. Enhancing Fact Verification with Causal Knowledge Graphs and Transformer-Based Retrieval for Deductive Reasoning. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*; Association for Computational Linguistics: Kerrville, TX, USA, 2024; pp. 151–169.
15. Yao, L.; Mao, C.; Luo, Y. KG-BERT: BERT for Knowledge Graph Completion. *arXiv* **2019**, arXiv:1909.03193. [CrossRef]

16. Wang, B.; Shen, T.; Long, G.; Zhou, T.; Wang, Y.; Chang, Y. Structure-Augmented Text Representation Learning for Efficient Knowledge Graph Completion. In *Proceedings of the Web Conference 2021*; ACM: New York, NY, USA, 2021; pp. 1737–1748. [CrossRef]
17. Wang, L.; Zhao, W.; Wei, Z.; Liu, J. SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-Trained Language Models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*; Association for Computational Linguistics: Kerrville, TX, USA, 2022; pp. 4281–4294. [CrossRef]
18. Baek, J.; Aji, A.F.; Saffari, A. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering. *arXiv* **2023**, arXiv:2306.04136. [CrossRef]
19. Jiang, J.; Zhou, K.; Dong, Z.; Ye, K.; Zhao, W.X.; Wen, J.R. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Kerrville, TX, USA, 2023; pp. 9237–9251. [CrossRef]
20. Wu, X.; Huang, M.; Bu, C. BEKO: Bidirectional Enhancement of Large Language Models and Knowledge Graphs. *Chin. J. Comput.* **2025**, *48*, 1572–1588.
21. Lu, Y.; Wu, W.; Zhao, X.; Peng, R.; Wang, J. KARMA: Leveraging Multi-Agent LLMs for Automated Knowledge Graph Enrichment. In Proceedings of the Advances in Neural Information Processing Systems, Sydney, Australia, 6–12 December 2026; Volume 38.
22. OpenAI. Models. Available online: https://platform.openai.com/docs/models (accessed on 23 September 2026).
23. Alibaba Cloud. Models: Alibaba Cloud Model Studio. Available online: https://www.alibabacloud.com/help/en/model-studio/getting-started/models (accessed on 23 September 2026).
24. DeepSeek-AI. DeepSeek API Docs. Available online: https://api-docs.deepseek.com/ (accessed on 23 September 2026).
25. Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; Liu, Z. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings through Self-Knowledge Distillation. *arXiv* **2024**, arXiv:2402.03216. [CrossRef]
26. Humeau, S.; Shuster, K.; Lachaux, M.A.; Weston, J. Poly-Encoders: Architectures and Pre-Training Strategies for Fast and Accurate Multi-Sentence Scoring. In Proceedings of the International Conference on Learning Representations, Virtual, 26–30 April 2020.
27. Sentence-Transformers. Cross-Encoder/Stsb-Roberta-Base. Available online: https://huggingface.co/cross-encoder/stsb-roberta-base (accessed on 23 September 2026).
28. Williams, A.; Nangia, N.; Bowman, S.R. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: Kerrville, TX, USA, 2018; pp. 1112–1122. [CrossRef]
29. Sun, Z.; Hu, W.; Li, C. Cross-Lingual Entity Alignment via Joint Attribute-Preserving Embedding. In *Proceedings of the Semantic Web–ISWC 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 628–644. [CrossRef]
30. Sun, Z.; Hu, W.; Li, C. JAPE: DBP15K Dataset. Available online: https://github.com/nju-websoft/JAPE/blob/master/data/dbp15k.tar.gz (accessed on 23 September 2026).
31. Zhang, Q.; Dong, J.; Duan, K.; Huang, X.; Liu, Y.; Xu, L. Contrastive Knowledge Graph Error Detection. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*; ACM: New York, NY, USA, 2022; pp. 2590–2599. [CrossRef]
32. Liu, X.; Liu, Y.; Hu, W. Knowledge Graph Error Detection with Contrastive Confidence Adaption. *Proc. AAAI Conf. Artif. Intell.* **2024**, *38*, 8824–8831. [CrossRef]
33. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-Relational Data. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–8 December 2013; Volume 26, pp. 2787–2795.
34. Xie, R.; Liu, Z.; Lin, F.; Lin, L. Does William Shakespeare Really Write Hamlet? Knowledge Representation Learning with Confidence. *Proc. AAAI Conf. Artif. Intell.* **2018**, *32*, 4954–4961. [CrossRef]
35. Sun, Z.; Deng, Z.H.; Nie, J.Y.; Tang, J. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
36. Meilicke, C.; Chekol, M.W.; Ruffinelli, D.; Stuckenschmidt, H. Anytime Bottom-Up Rule Learning for Knowledge Graph Completion. In Proceedings of the 28th International Joint Conference on Artificial Intelligence, Macao, China, 10–16 August 2019; pp. 3137–3143. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.