*applied sciences*

MDPI

*Article*

# Multi-Table Retrieval Method Based on Implicit Association Reasoning in the Petroleum Domain

**Chunping Liu [1], Meng Cai [2,*], Zhigang Yang [2], Bing Wang [3] and Chunhao Wang [3]**

[1] Da Qing Oilfield Party Committee Office, Daqing 163453, China
[2] Production Technology Institute, Daqing Oilfield Limited Company, Daqing 163453, China
[3] School of Computer Science and Software, Southwest Petroleum University, Chengdu 610500, China
* Correspondence: cyycaimeng@petrochina.com.cn

**Abstract**

In the digital transformation of the petroleum industry, massive multi-source heterogeneous tables are distributed across databases, Word documents, PDF reports, and engineering systems. Their unstructured format and weakly expressed inter-table relationships make it difficult for conventional keyword-based or single-table retrieval methods to locate the complete set of tables needed for complex queries. To address this problem, this paper proposes Relatab, a multi-table retrieval framework based on implicit association reasoning. Relatab first estimates query–table relevance through a dual-level semantic matching mechanism that combines table-level signals, including captions and column names, with value-level signals weighted by entropy and CRITIC criteria. It then constructs an implicit table graph using column-name and column-content similarity, and applies a max-product multi-hop propagation rule with decay and pruning to identify complementary tables that are not directly matched by the query. Finally, direct relevance and inter-table complementarity are fused to produce the retrieved table set. Experiments on Spider, Bird, and CementingTables show that Relatab achieves Top-2 recall rates of 79.21%, 61.26%, and 77.88%, respectively, outperforming DTR by 1.74, 2.33, and 1.85 percentage points. The results indicate that explicit modeling of implicit inter-table associations improves retrieval coverage in complex multi-table scenarios while remaining applicable to petroleum-domain documents.

**Keywords:** digital transformation; table retrieval; Natural Language Processing (NLP); unstructured data; implicit association reasoning

Check for updates

Academic Editor: Andrea Prati

Received: 12 June 2026
Revised: 1 July 2026
Accepted: 2 July 2026
Published: 14 July 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

Against the backdrop of digital transformation in the petroleum industry, the complexity of data storage and retrieval is increasing day by day. As a prevalent form of data representation, tables are widely utilized across various domains. Data within the petroleum industry comprises not only structured database tables but also a vast quantity of unstructured tables embedded within Word or PDF documents. These unstructured tables typically lack explicit structured connections or annotations, rendering it difficult for traditional retrieval methods to fully uncover the latent relationships between tables. Consequently, how to efficiently retrieve such unstructured tabular data has emerged as a critical issue demanding attention.

Table Retrieval is a crucial branch of Natural Language Processing (NLP), aimed at rapidly locating table information relevant to user queries within large-scale tabular

data to accurately answer given questions [1]. Research in this field primarily focuses on three directions.

First, traditional table retrieval models [2–4] mainly rely on keyword matching and statistical analysis, achieving retrieval through the direct correspondence of table headers, column names, or numerical values with the query. However, these methods struggle to capture the deep semantic correlations between natural language queries and structured or unstructured tables [5–8] and are sensitive to noise [9,10].

Second, methods based on pre-trained language models [11–13] leverage the semantic understanding capabilities of these models to significantly enhance matching accuracy between structured data and natural language queries. Nevertheless, they still depend on table structural information and find it difficult to capture implicit associations within unstructured data. Furthermore, existing studies mostly focus on single-table or simple multi-table scenarios, with limited ability to model complex cross-table relationships, making it difficult to meet the requirements of complex multi-table queries.

Third, table matching-based retrieval methods discover relevant tables by calculating similarity scores between input and candidate tables, making them suitable for multi-table associative queries. While effective for structured or semi-structured data by relying on explicit information such as column headers, entities, and inter-column relationships, their applicability is limited in unstructured data that lacks clear structural features.

How to efficiently mine implicit associations between tables and achieve effective retrieval of unstructured multi-table information has become a noteworthy issue. Figure 1 illustrates an example of unstructured multi-table retrieval involving implicit associations; the model must capture the implicit associative information within the data to obtain a set of tables containing the necessary information.

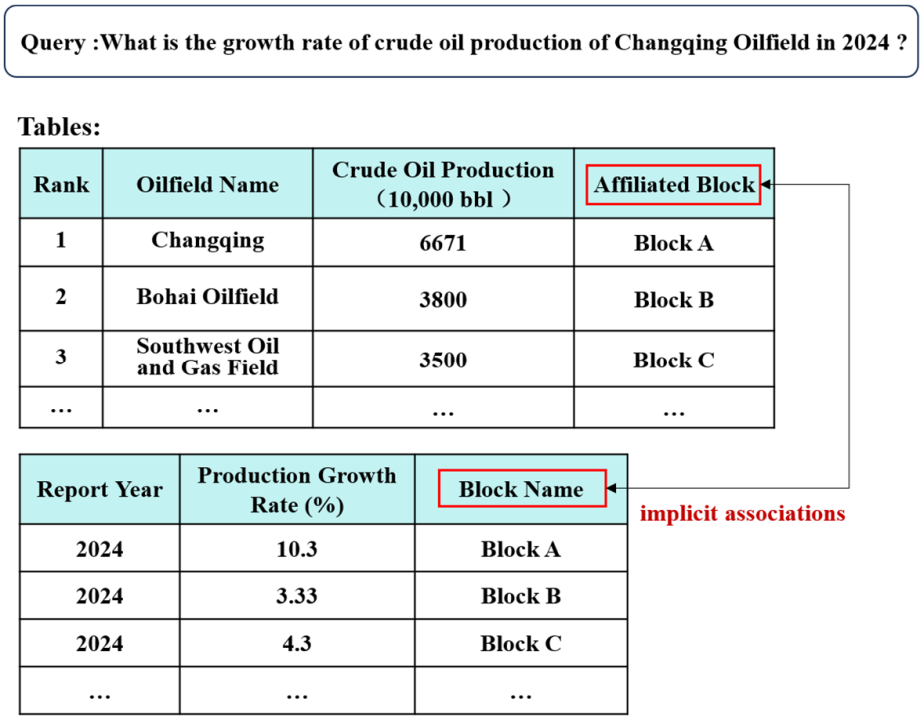


**Figure 1.** Implicit Associations example.

To further address the aforementioned challenges, this paper proposes a table retrieval method based on Implicit Association Inference. By integrating query-tablequery–table relevance and inter-table correlation, this method solves the difficulty of handling

implicit relationships in unstructured tabular data, identifying tables that are indirectly related to the query but hold significant value for answering it.

Specifically, the process is as follows: First, the query-tablequery–table similarity is calculated based on semantic matching. Subsequently, inter-table correlation is computed by analyzing the semantic similarity or contextual connections of table contents to identify other tables implicitly associated with the candidate tables. Finally, the two similarity scores are combined to optimize the final table ranking and determine the result set.

Experimental results demonstrate that on the Spider, Bird, and CementingTables datasets, the Top-2 recall rates of the proposed method are 79.21%, 61.26%, and 77.88%, respectively, outperforming the baseline model DTR by 1.74, 2.33, and 1.85 percentage points. This validates the method's ability to improve retrieval coverage in complex query scenarios and its applicability in the petroleum domain.

The main contributions of this study are summarized as follows. First, a task formulation for unstructured multi-table retrieval is introduced, targeting petroleum-domain documents in which foreign-key links, schemas, and annotations are often unavailable. Second, Relatab provides a two-level query-tablequery–table relevance model that jointly captures global table semantics and fine-grained value semantics. Third, an implicit inter-table reasoning module is designed as a pruned max-product table graph, allowing indirectly relevant tables to be retrieved through short high-confidence association chains. Fourth, experiments and diagnostic analyses on two public datasets and one private petroleum-domain dataset demonstrate consistent improvements over representative dense retrieval and table retrieval baselines.

## 2. Related Work

### *2.1. Table Retrieval Models Based on Traditional Methods*

In early research, Cafarella et al. [2,3] proposed models such as SimpleRank and SCPRank, which utilize web search engines and keyword frequency (e.g., TF-IDF) to extract the top-k tables from web pages. These models improve retrieval accuracy through attribute co-occurrence or re-ranking mechanisms. The OCTOPUS system adopted a similar approach; however, due to insufficient semantic understanding, such methods fail to capture the deep semantic relationships between queries and tables. Additionally, Venetis et al. [4] enhanced the model's understanding of queries by extracting class labels and relational database metadata to attach to table columns, but this approach relies heavily on high manual annotation costs. Zhang et al. [5] utilized word or graph embedding techniques to calculate the similarity between tables and queries, while statistical ranking methods like BM25 [6] optimized retrieval results through field weighting. Trabelsi [7] employed Bayesian networks to construct probabilistic models, and Chen [8] evaluated the relevance of tables to queries using semi-structured document retrieval techniques based on field relevance or language models. However, these methods are relatively sensitive to data noise. In unstructured data, they are easily susceptible to interference from irrelevant columns or redundant values, leading to a decrease in retrieval accuracy.

### *2.2. Table Retrieval Methods Based on Pre-Trained Language Models*

With the widespread application of Pre-trained Language Models (PLMs) in the field of Natural Language Processing, table retrieval methods based on such models have made significant progress in semantic understanding. Models such as TaBERT [11] and RoBERTa [12] serialize table column names, row values, and user queries into linear inputs. They generate embedding vectors through attention mechanisms and pooling operations, ranking results based on cosine similarity. However, these methods ignore the

two-dimensional structural information of tables (such as row-column relationships), resulting in incomplete semantic representation of complex tables.

To address the unique structural issues of tables, researchers have developed pre-trained models specifically designed for tabular data. Tapas [13] extends BERT to adapt to sparse inputs and optimizes the matching between queries and tables through a cell selection prediction task. However, it focuses primarily on single-table queries and struggles to capture cross-table implicit associations, limiting its performance in multi-table scenarios. TUTA [14] parses tables into a hierarchical tree structure and learns structure-aware representations through tree node masking tasks, but it is highly dependent on structured data and shows insufficient representation capability for unstructured or irregular tables. TABBIE [15] combines Bidirectional GRU (Bi-GRU) [16] and Graph Neural Networks (GNN) to capture cell context and cross-table structural relationships, improving implicit association modeling. Nevertheless, its complex architecture significantly increases computational costs, making it difficult to apply to large-scale datasets.

Joint encoding methods enhance the semantic alignment between queries and tables through multimodal interaction. TURL [17] extends table understanding to knowledge base interactions through relevance calculation, improving semantic association capabilities; however, its reliance on external knowledge bases limits its applicability in unstructured data scenarios that lack such support. SATR [18] fuses table content, structure, and type features to jointly optimize retrieval and question-answering tasks, but its ability to reason with fused features for complex multi-table queries is limited, making it difficult to handle implicit associations. Table-LLaVA [19], based on multimodal Large Language Models, jointly encodes table images and queries, optimizing multi-table retrieval through two-stage training. While it performs excellently in various table tasks, its high training and inference costs limit its deployment in resource-constrained environments.

### *2.3. Table Retrieval Based on Table Matching*

Retrieval methods based on table matching have been widely applied to handle multi-table association scenarios. Ahmadov et al. [20] utilized table elements (such as entities and headers) as keywords for querying, generating more complete multi-table associations through table matching and candidate set merging. However, due to the reliance on explicit keywords, this approach struggles to capture deep semantic relationships within unstructured data. Lehmberg et al. [21] proposed a search join engine that extends input tables through column header matching, filtering and scoring candidate tables that share column headers. Nevertheless, its inability to infer implicit inter-table relationships limits its applicability in unstructured scenarios. Sarma et al. [22] optimized multi-table associations by extending the rows (entity complementation) or columns (schema complementation) of seed tables, calculating similarity based on entity relevance and attribute gain. However, this method is sensitive to noise in unstructured data, leading to a decline in retrieval accuracy. Peter et al. [23] adopted an innovative re-ranking method that calculates query-tablequery–table relevance and inter-table connectivity using Mixed Integer Programming (MIP) to infer necessary associations. Yet, it relies on structured connection information, limiting its effectiveness in unstructured data lacking explicit fields.

To address the aforementioned challenges, this paper proposes the Relatab method. By jointly modeling query-tablequery–table relevance and implicit inter-table associations, Relatab automatically mines latent relationships within unstructured data, thereby enhancing retrieval precision and result coverage. This approach provides a novel solution for table retrieval in complex scenarios.

### 2.4. Comparison and Positioning of Relatab

Compared with representative retrieval approaches, the key distinction of Relatab lies not in replacing existing semantic encoders, but in explicitly modeling complementary relationships among multiple weakly structured tables. Dense retrieval methods (e.g., Contriever, DPR, and DTR) primarily rank tables according to direct query–table relevance, while TaBERT emphasizes stronger joint representations of tables and queries. Join-aware and MIP-based approaches further exploit explicit schemas or table links, but their effectiveness depends on structured relational information that is often absent in practical petroleum documents. In contrast, Relatab introduces an implicit table graph constructed from semantic column and content similarity, allowing indirectly relevant tables to be recovered through high-confidence association paths. By combining dual-level query relevance with max-product propagation, decay, and pruning, Relatab achieves a better balance between retrieval precision and information coverage, providing a practical solution for unstructured multi-table retrieval scenarios.

## 3. Problem Formulation

The core task of table retrieval is to select a set of tables from a corpus containing multiple tables, given a query $Q$, that are relevant to the query and capable of providing the necessary information to answer it. Traditional table retrieval methods typically assume that the answer to a query resides within a single table and rely on explicit matching to calculate the direct relevance between the query and the table. However, in real-world applications, especially in unstructured data scenarios, queries often necessitate the integration of content from multiple tables to derive a complete answer.

To address this challenge, this paper extends the task objective to unstructured multi-table retrieval. Given a query Q and a universal set of tables C, the goal is to identify the set of tables E(Q) necessary to answer query Q. Formally, the objective can be formulated as:

$$E(Q) = \{T_{Q,1}, T_{Q,2}, \ldots, T_{Q,k}\}, \quad T_{Q,i} \in C, \tag{1}$$

where E(Q) represents the set of tables required for the complete answer to query Q, and C denotes the universal set of all tables.

## 4. Methodology

To achieve efficient retrieval of unstructured multi-table information, this paper proposes a table retrieval method based on Implicit Association Inference. By integrating query-tablequery–table relevance and inter-table correlation, this method addresses the challenge of implicit relationships in unstructured tabular data, identifying tables that are indirectly related to the query but hold significant value for answering it.

Specifically, the process is as follows: First, query-table similarity is calculated based on semantic matching. Subsequently, inter-table correlation is computed by analyzing the semantic similarity or contextual connections of table contents to identify other tables implicitly associated with the candidate tables. Finally, the two similarity scores are combined to optimize the final table ranking and determine the result set. The structure of this method is illustrated in Figure 2, which presents the overall framework of the retrieval model proposed in this paper.

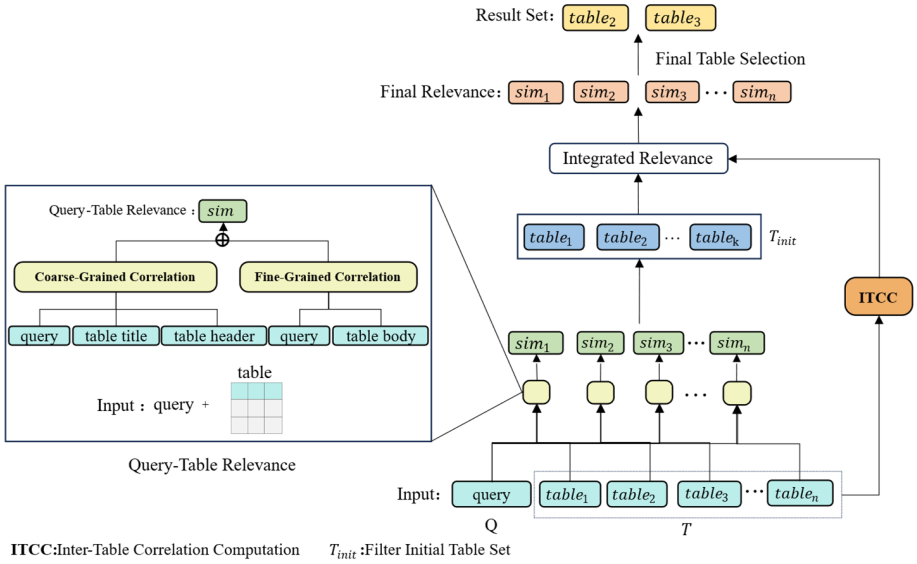


**Figure 2.** Model architecture diagram.

The algorithmic novelty of Relatab is the coupling between direct semantic matching and implicit association propagation. The query relevance module produces seed evidence, while the table graph module estimates whether another table is complementary to those seeds. The final decision therefore depends on both whether a table matches the query and whether it forms a high-confidence association path with an already relevant table. This design differs from standard graph propagation because the propagated score is query-conditioned: a table receives complementary evidence only through paths anchored by query-relevant seed tables.

### *4.1. Query—Table Relevance*

In table retrieval tasks, the calculation of query-tablequery–table relevance is a crucial step for evaluating the degree of matching between the two. This section proposes a query-tablequery–table matching framework based on a dual-level matching mechanism (Table-Level and Value-Level) to achieve semantic matching between the query and the table. The following subsections detail the table-level relevance calculation, the value-level relevance calculation, and the comprehensive evaluation method of the two.

#### 4.1.1. Table-Level Relevance Calculation

Table-level relevance calculation aims to rapidly assess the overall relevance between a query Q and a table T, thereby filtering out a set of candidate tables. This method combines the semantic matching of table captions and column names to evaluate the global topical relevance of the table. A schematic diagram of table-level relevance is shown in Figure 3.

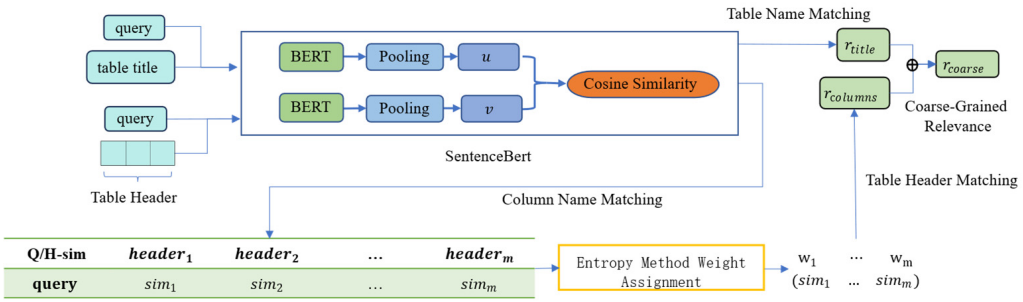


**Figure 3.** Table-Level Relevance Diagram.

Table Title Matching: As the core identifier reflecting the theme of a table, the table title is directly used to measure the overall semantic similarity between the query and the table. By utilizing the SentenceBERT model, the query Q and the table title $T_{title}$ are transformed into embedding vectors $E(Q)$ and $E(T_{title})$, respectively. The cosine similarity between them is then calculated as the title matching score $r_{title}$. The calculation formula is as follows:

$$r_{title} = sim(E(Q), E(T_{title})), \tag{2}$$

where $sim()$ denotes the cosine similarity function. SentenceBERT generates sentence embeddings with strong semantic associations; compared to traditional word embeddings (such as Word2Vec), it is better equipped to capture global semantic relationships between complex queries and table titles. All subsequent embedding calculations in this work are generated using the SentenceBERT model.

Column Name Matching: Column names reflect data dimensions and thematic information; thus, we evaluate the semantic relevance between the query and the table's column names. Let the set of column names for a table T be $\{c_1, c_2, \ldots, c_m\}$, and the semantic embedding vector for each column name $c_i$ be $E(c_i)$. Since the contribution of each column varies, an adaptive weighting scheme is required to adjust the influence of different columns. Here, the Entropy Weight Method (EWM) is employed to assign weights.

The EWM assigns weights by evaluating the entropy value of each column name (based on the distribution of generated embedding vectors or vocabulary frequency). Column names with lower information entropy (i.e., those with uneven distribution and higher distinctiveness) are assigned higher weights $w_i$ to reflect their importance in the table's semantics. Finally, weight normalization ensures that $\sum_{i=1}^{m} w_i = 1$. Compared to other weight allocation methods, the entropy method is computationally efficient and well-suited for the global matching requirements of table-level calculations involving limited features. The final column name matching score is calculated via weighted cosine similarity:

$$r_{columns} = \frac{1}{m} \sum_{i=1}^{m} w_i \cdot sim(E(Q), E(c_i)), \tag{3}$$

where m denotes the total number of columns, and $w_i$ is dynamically determined by the entropy method.

Comprehensive Matching Calculation: To comprehensively evaluate the table-level relevance between the query and the table, this paper performs a weighted fusion of the table title matching score $r_{title}$ and the column name matching score $r_{columns}$ to obtain the final table-level relevance score:

$$r_{coarse} = \alpha \cdot r_{title} + (1 - \alpha) r_{columns}, \tag{4}$$

where $\alpha \in [0, 1]$ is a hyperparameter used to balance the importance of table title matching and column name matching in the table-level relevance calculation.

#### 4.1.2. Value-Level Relevance Calculation

Value-level relevance calculation serves as a vital component of the multi-table retrieval method based on Implicit Association Inference. It aims to capture deeper-level associations by analyzing the degree of semantic matching between the query and the values within table columns. This process compensates for the fine-grained information that may be overlooked by table-level relevance (which relies primarily on table titles and column names). A schematic diagram illustrating the value-level relevance calculation is presented in Figure 4.

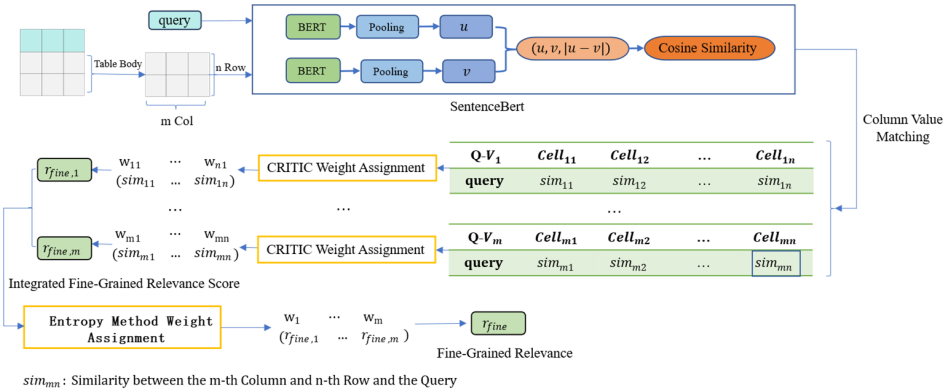


**Figure 4.** Value-Level Relevance Computation Diagram.

In the calculation of value-level relevance, this paper adopts a column-oriented representation method, where $V_j$ represents the set of values in the j-th column of table T, and $V_{ji}$ denotes the i-th value within the j-th column. Utilizing the SentenceBERT model, the query Q and $V_{ji}$ are vectorized to obtain $E(Q)$ and $E(V_{ji})$, respectively. The cosine similarity $\mathrm{sim}(E(Q), E(V_{ji}))$ is then calculated.

By analyzing the semantic similarity between each column $V_j$ of table T and the query Q, the column value matching score $r_{fine,j}$ is derived. To precisely assign weights $w_{ji}$ to column values, the CRITIC method (Criteria Importance Through Intercriteria Correlation) is employed. The CRITIC method dynamically allocates weights based on the contrast intensity (standard deviation) and conflict (correlation between columns) of the similarities. The column value matching score is calculated using weighted cosine similarity:

$$r_{fine,j} = \frac{\sum_{i=1}^{n_j} w_{ji} \cdot sim(E(Q), E(V_{ji}))}{\sum_{i=1}^{n_j} w_{ji}}, \tag{5}$$

This score captures the local semantic matching relationship between the query and column values, ensuring that the value-level analysis focuses on specific content.

To comprehensively evaluate the value-level relevance between the query and the table, this paper further integrates the column value matching scores to calculate the table's overall value-level relevance score $r_{fine}$. The comprehensive score is obtained through the weighted fusion of individual column scores:

$$r_{fine} = \frac{\sum_{j=1}^{n} w_j \cdot r_{fine,j}}{\sum_{j=1}^{n} w_j}, \tag{6}$$

where $n$ denotes the number of table columns, and $w_j$ is the weight of the j-th column, which is dynamically allocated by the Entropy Weight Method (EWM). This comprehensive score reflects the overall matching degree between the query and the table content.

#### 4.1.3. Comprehensive Relevance Score

To comprehensively evaluate the overall relevance between the query Q and the table T, this section integrates the table-level relevance score $r_{coarse}$ and the value-level relevance score $r_{fine}$ to formulate the final table relevance score $R(T, Q)$. This approach balances the table's global topical matching (table-level) with its local content matching (value-level), providing a reliable basis for the screening and ranking of candidate tables. The table relevance score is calculated via the weighted fusion of the table-level score and the value-level score as follows:

$$R(T, Q) = \beta \cdot r_{coarse} + (1 - \beta) \cdot r_{fine}, \tag{7}$$

where $\beta \in [0,1]$ is a weighting parameter used to balance the importance of table-level relevance and value-level relevance.

Through the aforementioned weighted combination, $R(T,Q)$ is able to fully consider the matching performance of the table at different levels, thereby precisely identifying the target tables most relevant to the query from the candidate tables.

In summary, the comprehensive relevance score improves the precision and coverage of table retrieval in unstructured data through multi-level semantic matching, laying a solid foundation for the subsequent screening of target tables.

### 4.2. Inter-Table Correlation

In table retrieval tasks, complex queries often necessitate the integration of content from multiple tables to yield comprehensive answers. Consequently, capturing the latent associations between tables is central to achieving this objective. Within the context of unstructured data in the petroleum industry, inter-table relationships may be explicit (e.g., foreign keys) or implicit (e.g., content sharing or structural consistency). These implicit relationships are manifested through shared column names, data schemas, or semantic consistency.

This paper proposes a modeling method based on inter-table correlation, specifically designed to mine implicit associations within unstructured data. This section unfolds in two aspects: first, Inter-Table Similarity Calculation, which evaluates the similarity of column values between two tables; and second, Chain Relationship Modeling, which captures implicit relationships across multiple tables by constructing multi-hop paths.

#### 4.2.1. Inter-Table Similarity Calculation

Inter-table similarity calculation is a pivotal step in assessing the correlation between two tables and serves as a prerequisite for measuring implicit inter-table associations. This section identifies inter-table associations by modeling the semantic consistency of column names and column contents between tables.

Let the sets of column names for Table A and Table B be $\{c_{a1}, c_{a2}, \ldots, c_{am}\}$ and $\{c_{b1}, c_{b2}, \ldots, c_{bn}\}$, respectively. Here, $c_{ai}$ and $c_{bj}$ denote the column names of the i-th column in Table A and the j-th column in Table B. Column name similarity measures the semantic consistency of the column names, which typically reflect the primary informational content of the table. Using the SentenceBERT model, the embedding vectors $E(c_{ai})$ and $E(c_{bj})$ corresponding to column names $c_{ai}$ and $c_{bj}$ are generated. The column name similarity $sim_{name}(c_{ai}, c_{bj})$ is defined as the cosine similarity between the vectors $E(c_{ai})$ and $E(c_{bj})$.

Column content similarity measures the semantic consistency between the contents (value sets) of two columns. Assuming the contents of columns $c_{ai}$ and $c_{bj}$ are value sets $V_{c_{ai}}$ and $V_{c_{bj}}$, which are transformed into corresponding embedding vectors $E(V_{c_{ai}})$ and $E(V_{c_{bj}})$, the column content similarity $sim_{content}(c_{ai}, c_{bj})$ is defined as:

$$sim_{content}(c_{ai}, c_{bj}) = sim(E(V_{c_{ai}}), E(V_{c_{bj}})), \tag{8}$$

where $E(V_{c_{ai}})$ and $E(V_{c_{bj}})$ are the average embedding vectors of the value sets (obtained by performing Mean Pooling on all value embeddings).

By integrating column name similarity and column content similarity, the final similarity for a column pair, denoted as $sim_{column}(c_{ai}, c_{bj})$, is defined as:

$$sim_{column}(c_{ai}, c_{bj}) = \gamma \cdot sim_{name}(c_{ai}, c_{bj}) + (1-\gamma) \cdot sim_{content}(c_{ai}, c_{bj}) \tag{9}$$

where $\gamma \in [0, 1]$ is a hyperparameter used to balance the weights of column name similarity and column content similarity. The calculation of comprehensive inter-column similarity is illustrated in Figure 5.

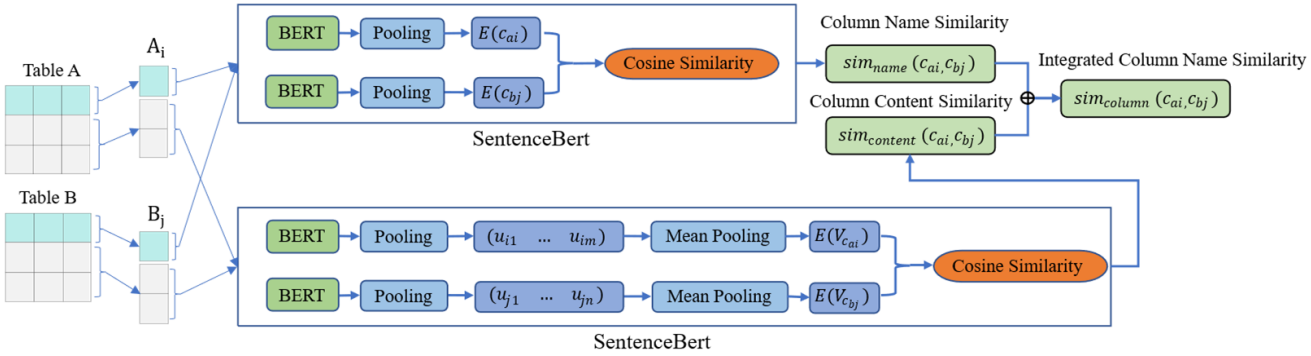


**Figure 5.** Inter-Column Integrated Similarity Computation Diagram.

The overall similarity $R_{table}(A, B)$ between Table A and Table B is defined as the maximum value among all column similarities. This approach is designed to highlight the strongest column association between the two tables. The calculation formula is:

$$R_{table}(A, B) = \max_{a_i \in A, b_j \in B} sim_{column}(c_{ai}, c_{bj}), \tag{10}$$

By employing the above method, inter-table similarity quantifies the overall correlation between tables from the dual dimensions of column names and column contents, thereby enhancing the capability to capture implicit complementary relationships within unstructured data.

#### 4.2.2. Chain Relationship Modeling

Section 4.2.1 described the calculation method for the inter-table correlation between two tables. However, relying solely on direct inter-table correlation cannot fully capture complex table relationships. For instance, if Table $T_1$ is associated with Table $T_2$, and $T_2$ is further associated with Table $T_3$, is there an implicit association between $T_1$ and $T_3$? Such indirect, cross-table association relationships are difficult to reveal through single pairwise analysis. To this end, Chain Relationship Modeling is introduced to mine implicit connections between tables through multi-hop association paths, constructing a broader representation of inter-table relationships, which helps improve the coverage and precision of multi-table retrieval.

A chain is defined as a path of tables connected by implicit associations. A single-hop chain $T_i \rightarrow T_j$ indicates that $T_i$ and $T_j$ have an implicit association, while a multi-hop chain $T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow T_j$ indicates an association between $T_i$ and $T_j$ via intermediate tables. The similarity of a chain is calculated by the accumulation of similarity scores for each hop, while introducing a decay factor $\lambda$ to control the influence of indirect relationships, causing them to weaken gradually. The similarity of a single-hop chain is defined by Equation (11). The similarity of a d-hop chain is defined as:

$$R(T_i, T_j) = \lambda^{d-1} \cdot \prod_{k=i}^{d} sim(T_k, T_{k+1}), \tag{11}$$

where $d$ is the number of hops in the chain, and $T_k$ and $T_{k+1}$ are two adjacent tables in the chain; $sim(T_k, T_{k+1})$ denotes the similarity between adjacent tables.

To avoid introducing noise and improve computational efficiency, this paper imposes the following constraints on the calculation of multi-hop chains: First, a maximum hop

limit $d_{max}$ is set to control the range of chain propagation, ensuring computational effectiveness. Second, a similarity threshold θ is established, retaining only chain paths with similarity scores greater than θ, thereby pruning low-quality chains. Furthermore, when multiple chains may connect $T_i$ and $T_j$ via different paths, the chain with the highest similarity is selected as the final score:

$$R_{table}(T_i, T_j) = \max_{chains}(R(T_i, T_j)), \tag{12}$$

To efficiently calculate the relevance of multi-hop chains, this study introduces an inter-table similarity matrix S. The element S[i, j] of matrix S represents the similarity between table $T_i$ and $T_j$. Leveraging the transitive property of matrices, the similarity of multi-hop paths can be computed rapidly. A single-hop chain corresponds to the similarity matrix S itself, while a two-hop chain can be derived through the transmission of matrix S. In general, the correlation of a d-hop path can be expressed as:

$$S^d[i, j] = \max_{k \in \{1,2,\ldots,N\}}(S^{d-1}[i, k] \cdot S[k, j]), \tag{13}$$

where k represents the index of the intermediate table, and N denotes the total number of tables. Finally, the similarity of multi-hop chains is calculated using the following formula:

$$R_{table}(T_i, T_j) = \max_{d=1,\ldots,d_{max}}(\lambda^{d-1} \cdot S^d[i, j]), \tag{14}$$

This method avoids the explicit enumeration of all chains; instead, it models the similarity of all possible chains through efficient matrix operations. Combined with the maximum similarity selection operation, it retains the path with the highest similarity score among multi-hop chains.

Although long chains can capture deeper complementary information, they may introduce noise. Therefore, by limiting the maximum number of hops $d_{max}$ and setting the similarity threshold θ, the precision and efficiency of chain relationship modeling are ensured.

The multiplication of edge similarities follows a reliability interpretation: a multi-hop association should be strong only when every edge along the path is reliable. A single weak edge lowers the total path score, which is desirable in noisy unstructured tables. The maximum-path selection is used because retrieval requires identifying the strongest explanatory association rather than accumulating many weak paths that may be semantically unrelated. This is different from random walks or Personalized PageRank, which distribute probability mass over many paths and may over-amplify high-degree tables. Compared with Graph Neural Networks and attention-based propagation, the proposed rule is parameter-light, does not require graph-label supervision, and is easier to deploy in small private-domain datasets. These choices make Relatab suitable for petroleum documents where labeled multi-table associations are limited and reproducibility is important.

### *4.3. Comprehensive Table Scoring and Target Table Selection*

To achieve effective screening of candidate tables, this paper proposes a target table selection method based on comprehensive scoring. This method balances the weights of direct query-tablequery–table relevance and inter-table complementary relevance to comprehensively evaluate the overall matching degree between candidate tables and the query, thereby providing robust support for the subsequent selection of target tables.

#### 4.3.1. Comprehensive Score Calculation

In table retrieval tasks, obtaining a set of target tables capable of completely answering a user query requires the integrated consideration of both Query-TableQuery–Table Relevance and Inter-Table Correlation. Direct query-tablequery–table relevance reflects the degree of semantic matching between the table and the query, whereas inter-table correlation is utilized to identify complementary tables that are closely associated with the tables directly relevant to the query. By fusing these two types of relevance scores, the proposed method can comprehensively evaluate the composite relevance of candidate tables, ultimately selecting a table set that can both directly answer the query and provide supplementary information.

The calculation of the comprehensive score is divided into two steps: first, determining the initial table set $T_{init}$; second, calculating the direct relevance and complementarity scores for the candidate tables. The initial table set $T_{init}$ is composed of tables that have high direct relevance to the query Q, filtered via the query-tablequery–table relevance score. Specifically, a table $T_i \in T_{init}$ must satisfy the following condition:

$$T_{init} = \{T_i \mid R_{query}(T_i, Q) \geq \tau\}, \tag{15}$$

where $\tau$ is the threshold for direct query relevance.

For each candidate table $T_j$ (representing a specific table in the candidate set C), its direct relevance to the query, $R_{query}(T_j, Q)$, is calculated first. This score reflects whether the table directly answers the query content. The direct query relevance combines both table-level and value-level similarity scores (refer to Sections 4.1.2 and 4.1.3).

After completing the calculation of direct query relevance, the complementarity of $T_j$ with respect to the initial table set $T_{init}$ is evaluated. The complementarity score is measured via the inter-table correlation $R_{table}(T_j, T_i)$, where $T_i \in T_{init}$. Specifically, the complementarity score for $T_j$ is defined as:

$$R_{supplement}(T_j, Q) = \max_{T_i \in T_{init}} [R_{query}(T_i, Q) \cdot R_{table}(T_j, T_i)], \tag{16}$$

Using the maximum value to calculate the complementarity score between the candidate table $T_j$ and the most relevant table in $T_{init}$ allows the model to focus on the strongest association path between $T_j$ and the query, thereby avoiding the introduction of noise from weak associations. The inter-table correlation $R_{table}(T_j, T_i)$ is defined based on Sections 4.2.1 and 4.2.2, reflecting the strength of implicit associations between the two tables.

Finally, by integrating the direct relevance score and the complementarity score, the final comprehensive relevance score $R_{final}(T_j, Q)$ for the candidate table $T_j$ is calculated as:

$$R_{final}(T_j, Q) = R_{query}(T_j, Q) + x \cdot R_{supplement}(T_j, Q), \tag{17}$$

where x is a hyperparameter used to balance the importance of direct query relevance and complementary relationships.

Through the above process, the comprehensive score fully reflects the candidate table's ability to directly answer the query as well as its potential to provide additional information through inter-table complementarity, laying a solid foundation for the subsequent ranking and screening of target tables.

#### 4.3.2. Target Table Selection

Upon completion of the calculation of the comprehensive relevance score $R_{final}(T_j, Q)$, a straightforward threshold strategy can be adopted for the selection of target tables.

Specifically, only tables with scores exceeding a preset threshold ϕ are retained to form the target table set E(Q). The selection rule is defined as:

$$E(Q) = \left\{ T_j \mid R_{final}(T_j, Q) \geq \phi \right\}, \tag{18}$$

where ϕ represents the minimum threshold for the comprehensive score, which can be configured according to the specific requirements of the task. The introduction of this threshold effectively filters out tables with low relevance to the query Q, thereby ensuring the quality of the target table set.

## 5. Experimental Design and Result Analysis

### 5.1. Evaluation Metrics

To comprehensively evaluate the performance of table retrieval, this paper adopts Precision, Recall, F1-score, and Normalized Discounted Cumulative Gain (NDCG) as standard evaluation metrics. These metrics effectively measure the system's performance in terms of retrieval precision, coverage, and comprehensive capability, making them suitable for evaluating multi-table retrieval tasks based on implicit relationships.

Among them, NDCG primarily emphasizes the ranking order of the returned results; that is, strongly relevant results should be positioned earlier in the list. The relevance score of each item in the returned result list is termed as "gain". Considering the positional influence of each item—where items appearing later are subject to a greater discount—the Discounted Cumulative Gain (DCG) is calculated. The calculation formula is as follows:

$$DCG_k = \sum_{i=1}^{k} \frac{rel(i)}{\log_2(i+1)}, \tag{19}$$

where k represents the number of returned items, and rel(i) denotes the relevance score of the i-th item.

Since the number of results returned by different queries varies, DCG is merely a cumulative value, making it difficult to compare performance across different queries. Therefore, normalization is required to derive the NDCG. The calculation formula is as follows:

$$NDCG = \frac{DCG}{IDCG}, \tag{20}$$

where IDCG (Ideal DCG) represents the optimal ranking result obtained by sorting the DCG based on rel(i) in descending order.

### 5.2. Datasets

#### 5.2.1. Private Dataset in the Petroleum Domain

The CementingTables dataset is constructed following the format of the WikiTable dataset, based on tables extracted from documents regarding a cementing platform provided by a certain oil company. The queries in this dataset were formulated by relevant personnel from the School of Petroleum and actual users of the laboratory's cementing engineering system. This dataset consists of a total of 2341 query-tablequery–table pairs. Its specific format is illustrated in Figure 6.

For readability, the example in Figure 6 is presented bilingually. The query "BZ23 basic data" corresponds to the original Chinese query "BZ23的基本数据"; the fields include well name, hole size, well depth, wellbore capacity, drill pipe size, cement plug top depth, and cement plug bottom depth.

```
{
  "query": "BZ23 basic data (BZ23 basic data)",
  "tableid": "table-24-03",
  "docid": "doc-31",
  "caption": "Basic data",
  "title": [
    "Well name", "Hole size", "Well depth",
    "Wellbore capacity", "Drill pipe size",
    "Cement plug top depth", "Cement plug bottom depth"
  ],
  "data": ["BZ23-2-1", "12-1/4", "2475", "76.04", "5", "1940", "2190"]
}
```


**Figure 6.** Example of CementingTables.

#### 5.2.2. Public Datasets

Currently, public datasets for large-scale multi-table retrieval tasks are relatively scarce. Taking KaggleDBQA [24] as an example, it contains only 26 query samples involving multiple tables, which fails to meet the requirements of this study. Therefore, this paper selects Text2SQL datasets as the basis for constructing a multi-table retrieval dataset. Such datasets possess rich examples of multi-table queries and can provide effective support for research on multi-table retrieval tasks.

In terms of dataset construction, this paper adopts the method of Chen et al. [25], selecting Spider [26] and Bird [27], two representative Text2SQL datasets, as evaluation objects. These two datasets organize data using a topic-partitioning strategy, where each topic corresponds to a database (containing an average of 5.4 data tables) and provides a corresponding set of queries.

To construct an unstructured implicit association dataset suitable for multi-table retrieval tasks, this paper processes the Spider and Bird datasets based on the following steps:

Corpus Integration: First, tables are integrated into a unified corpus while retaining unstructured characteristics (such as irregular headers and missing fields) to simulate scenarios involving Word or PPT documents.

Constraint Removal and Renaming: Second, all foreign key constraints and metadata are removed. Associated fields are replaced with names that are semantically similar but distinct (e.g., changing "dept_id" to "dept_code"). This compels the model to mine implicit associations through semantic reasoning.

Query Filtering: Finally, multi-table queries are filtered to exclude single-table queries, retaining only complex queries that require cross-table reasoning.

After processing, Spider contains 3563 queries and 287 tables, while Bird contains 6395 queries and 390 tables, constituting an unstructured multi-table retrieval dataset suitable for mining implicit associations. Table 1 summarizes the details of the datasets.

**Table 1.** Statistics of multi-table retrieval.

| Dataset | Train | Valid |
|---|---|---|
| Bird | 4476 | 639 |
| Spider | 2494 | 356 |

### *5.3. Experimental Setup*

The experiment utilizes SentenceBERT, specifically the pre-trained model sentence-transformers/all-MiniLM-L6-v2, to provide efficient sentence embedding generation. The

training settings include the use of the AdamW optimizer with a learning rate of $2 \times 10^{-5}$. The model is trained for 10 epochs with a Batch Size of 16 and 100 Warmup Steps. All parameters were optimized on the validation set.

### 5.4. Baseline Models

To verify the effectiveness of the proposed multi-table retrieval method based on implicit association inference in unstructured data, this paper selected multiple classic models for comparison on the same dataset. The chosen baseline models include Contriever-msmarco3 [28], TaBERT [11], DPR [29], and DTR [30]. These models represent mainstream approaches for text retrieval and semi-structured/unstructured table retrieval, respectively. The aim is to highlight the advantages of the proposed method in handling implicit associations in unstructured data through comparison.

### 5.5. Main Experimental Results

To validate the effectiveness of the Relatab model (based on implicit inter-table relationship inference) in multi-table retrieval, comparative experiments were conducted against baseline models (Contriever-msmarco3, TaBERT, DPR, and DTR) on the public datasets Spider and Bird. As shown in Table 2, the experimental results indicate that Relatab outperforms the baseline models in terms of Precision, Recall, and F1-score in both Top-2 and Top-5 scenarios.

**Table 2.** Hyperparameter configurations used in the experiments.

| Parameter | Meaning | Candidate Range | Selected Value |
|---|---|---|---|
| alpha | Title vs. column-name relevance in Equation (4) | 0.1 to 0.9 | 0.50 |
| beta | Table-level vs. value-level relevance in Equation (7) | 0.1 to 0.9 | 0.60 |
| gamma | Column-name vs. content similarity in Equation (9) | 0.1 to 0.9 | 0.40 |
| x | Complementary inter-table score weight in Equation (17) | 0.1 to 0.9 | 0.30 |
| lambda | Multi-hop decay factor in Equation (14) | 0.1 to 0.9 | 0.70 |
| tau | Initial query-relevance threshold in Equation (15) | 0.50 to 0.80 | 0.65 |
| theta | Inter-table path pruning threshold | 0.30 to 0.60 | 0.45 |
| phi | Final target-table selection threshold in Equation (18) | 0.60 to 0.80 | 0.70 |
| dmax | Maximum association-chain length | 1 to 4 | 2 |

On the Spider dataset, Relatab's Top-2 Recall reached 79.21%, exceeding DTR by 1.74 percentage points; the F1-score reached 80.34%, exceeding DTR by 0.93 percentage points, demonstrating strong semantic matching capability in handling complex multi-table queries. The Top-5 Recall reached 97.55%, exceeding DTR by 1.67 percentage points, highlighting its advantage in coverage. This indicates that the algorithm achieves a superior ranking result.

On the Bird dataset, Relatab's Top-2 Recall was 61.26%, surpassing DTR by 2.33 percentage points; the F1-score was 63.28%, surpassing DTR by 1.48 percentage points; and the Top-5 Recall reached 85.47%, surpassing DTR by 2.49 percentage points, further reflecting its coverage capability in multi-table retrieval. The NDCG@2 and NDCG@5 metrics were higher than those of the baseline models across all datasets, indicating that the proposed method possesses high ranking quality. The experimental results demonstrate that Relatab holds unique value in multi-table retrieval within the domain of unstructured data, providing a more effective solution for complex multi-table queries.

To verify the effectiveness of chain relationship modeling in table retrieval, this paper compared the F1-score performance on the Spider and Bird datasets under different maximum hop counts ($d_{max}$ = 1, 2, 3, 4) to analyze their impact on retrieval performance. As shown in Table 3, the results indicate that $d_{max} = 2$ is the optimal setting for mining implicit associations in unstructured data. As $d_{max}$ increases beyond this point, the F1-score shows a downward trend, suggesting that overly long chains may introduce noise or redundant associations, affecting retrieval accuracy. When $d_{max} = 1$, the F1-score is lower than that of $d_{max} = 2$, because with a small hop count, the model can only capture direct associations and fails to fully mine deeper implicit inter-table relationships, leading to limited retrieval capability. The variation in F1-score further validates the robustness of Relatab in unstructured data.

**Table 3.** Comparative experimental results of this method with baseline methods.

| Dataset | Model | Top-2 | | | Top-5 | | | NDCG@2 | NDCG@5 |
|---|---|---|---|---|---|---|---|---|---|
| | | P | R | F1 | P | R | F1 | | |
| Spider | Contriever | 76.29 | 72.78 | 74.49 | 39.84 | 93.38 | 55.85 | 76.07 | 71.57 |
| | TaBERT | 78.59 | 72.79 | 75.58 | 40.76 | 93.67 | 56.80 | 77.01 | 72.33 |
| | DPR | 81.20 | 77.23 | 79.15 | 40.93 | 94.97 | 57.47 | 82.72 | 72.64 |
| | DTR | 81.45 | 77.47 | 79.41 | 41.05 | 95.88 | 57.49 | 83.88 | 72.84 |
| | Relatab (Ours) | 81.50 | 79.21 | 80.34 | 40.88 | 97.55 | 57.62 | 84.61 | 73.96 |
| Bird | Contriever | 65.03 | 59.46 | 62.12 | 37.04 | 82.96 | 51.21 | 70.63 | 67.27 |
| | TaBERT | 66.23 | 59.17 | 62.50 | 37.96 | 83.21 | 52.14 | 72.25 | 68.08 |
| | DPR | 64.75 | 58.70 | 61.58 | 37.11 | 82.72 | 51.23 | 75.31 | 67.37 |
| | DTR | 64.97 | 58.93 | 61.80 | 37.21 | 82.98 | 51.38 | 75.51 | 67.47 |
| | Relatab (Ours) | 65.43 | 61.26 | 63.28 | 37.86 | 85.47 | 52.48 | 76.12 | 68.40 |
| Cementing Tables | Contriever | 67.36 | 73.44 | 70.27 | 39.91 | 94.06 | 56.04 | 72.32 | 68.40 |
| | TaBERT | 68.81 | 74.22 | 71.41 | 40.38 | 94.60 | 56.60 | 73.03 | 69.38 |
| | DPR | 71.64 | 76.98 | 73.80 | 41.66 | 96.25 | 58.10 | 76.56 | 71.32 |
| | DTR | 71.76 | 76.03 | 73.83 | 41.72 | 96.32 | 58.22 | 76.66 | 73.96 |
| | Relatab (Ours) | 72.79 | 77.88 | 75.25 | 41.66 | 97.20 | 58.32 | 79.21 | 75.35 |

Note: P, R, and F1 denote Precision, Recall, and F1-score, respectively. Top-k indicates that the top-k ranked tables are selected as the retrieval results.

### 5.6. Ablation Study

To investigate the specific contributions of query–table relevance and inter-table correlation in table retrieval, this paper designed an ablation study comparing the retrieval performance of using only query–table relevance versus combining both query–table and inter-table correlations.

As shown in Table 4, using only inter-table reasoning produces the lowest performance because the retrieval process lacks sufficient query anchoring and may select tables that are mutually related but irrelevant to the query. The QT-only variant performs better, showing that direct query–table relevance is the foundation of retrieval. Removing chain reasoning also reduces F1-score, indicating that multi-hop implicit association helps recover indirectly related complementary tables. When CRITIC is removed, performance decreases because uniform weighting cannot emphasize discriminative column values. The full Relatab model achieves the best F1-score on all three datasets, confirming that query–table relevance, adaptive value weighting, and chain-based inter-table reasoning contribute complementary improvements.

The experimental results (see Table 5) indicate that after incorporating inter-table correlation, the model's Recall and F1-score improved significantly across multiple datasets. This improvement is primarily attributed to the ability of inter-table correlation to capture implicit relationships between tables, thereby assisting in the retrieval of more comple-

mentary tables related to the query. For instance, on the Spider dataset, Recall increased from 76.77% to 79.21% (an increase of 2.44 percentage points), while the F1-score increased from 78.14% to 80.34% (an increase of 2.20 percentage points). This demonstrates that the model maintains high precision while expanding the retrieval scope. On the Bird dataset, the F1-score increased from 62.53% to 63.28% (up 0.75 percentage points), and Recall increased from 58.99% to 61.26% (up 2.27 percentage points). These results further validate the effectiveness of inter-table correlation in unstructured table retrieval tasks.

**Table 4.** Extended ablation results of major Relatab components.

| Dataset | Variant | P | R | F1 |
|---|---|---|---|---|
| Spider | Relatab (TT only) | 74.38 | 70.52 | 72.40 |
| | Relatab (QT only) | 79.57 | 76.77 | 78.14 |
| | Relatab *w/o* chain reasoning | 80.04 | 76.84 | 78.41 |
| | Relatab *w/o* CRITIC | 80.81 | 78.51 | 79.64 |
| | Full Relatab | 81.50 | 79.21 | 80.34 |
| Bird | Relatab (TT only) | 59.83 | 55.86 | 57.78 |
| | Relatab (QT only) | 66.53 | 58.99 | 62.53 |
| | Relatab *w/o* chain reasoning | 65.64 | 60.30 | 62.85 |
| | Relatab *w/o* CRITIC | 65.77 | 60.31 | 62.92 |
| | Full Relatab | 65.43 | 61.26 | 63.28 |
| CementingTables | Relatab (TT only) | 65.92 | 71.44 | 68.57 |
| | Relatab (QT only) | 68.79 | 75.23 | 71.87 |
| | Relatab *w/o* chain reasoning | 70.39 | 75.53 | 72.87 |
| | Relatab *w/o* CRITIC | 71.23 | 76.72 | 73.87 |
| | Full Relatab | 72.79 | 77.88 | 75.25 |

Note: TT only denotes the variant that uses only inter-table complementary relevance for ranking, without directly adding query–table relevance. QT only denotes the variant using only direct query–table relevance. *w/o* chain reasoning removes multi-hop propagation and only keeps direct inter-table similarity. *w/o* CRITIC replaces CRITIC-based value weighting with uniform weights. Full Relatab uses all proposed components.

**Table 5.** F1-Score comparison under different maximum hop counts.

| Dataset | F1 | | | |
|---|---|---|---|---|
| | $d_{max} = 1$ | $d_{max} = 2$ | $d_{max} = 3$ | $d_{max} = 4$ |
| Spider | 78.41 | 80.34 | 79.82 | 76.34 |
| Bird | 62.85 | 63.28 | 63.04 | 59.11 |
| CementingTables | 72.87 | 75.25 | 73.54 | 70.93 |

To further analyze the effect of inter-table similarity in table retrieval tasks of varying difficulty, this paper explores the impact of inter-table similarity on retrieval performance by comparing the F1-scores of models under different numbers of tables. The number of tables is a key indicator for measuring task complexity; single-table tasks are relatively simple, whereas multi-table tasks (especially those involving 3 or more tables) are more challenging due to complex data associations. The experiments were conducted on the Spider and Bird datasets, and the results are illustrated in Figure 7.

As the number of tables increases, the overall F1-score of the model shows a downward trend, indicating that task complexity has a significant impact on retrieval effectiveness. Specifically, Relatab(QT + TT), by combining query–table relevance and inter-table correlation, effectively captures implicit relationships between tables. This provides more supplementary information for complex queries, thereby maintaining relatively stable performance in multi-table scenarios. Taking the results for $\geq$ 3 tables in Figure 7 as an example, Relatab(QT + TT) demonstrates superior performance in difficult tasks compared to other baseline models. This suggests that inter-table similarity plays a crucial role in

multi-table retrieval tasks, particularly when handling complex queries; it effectively compensates for the limitations of single-relevance methods, providing critical support for improving retrieval performance.

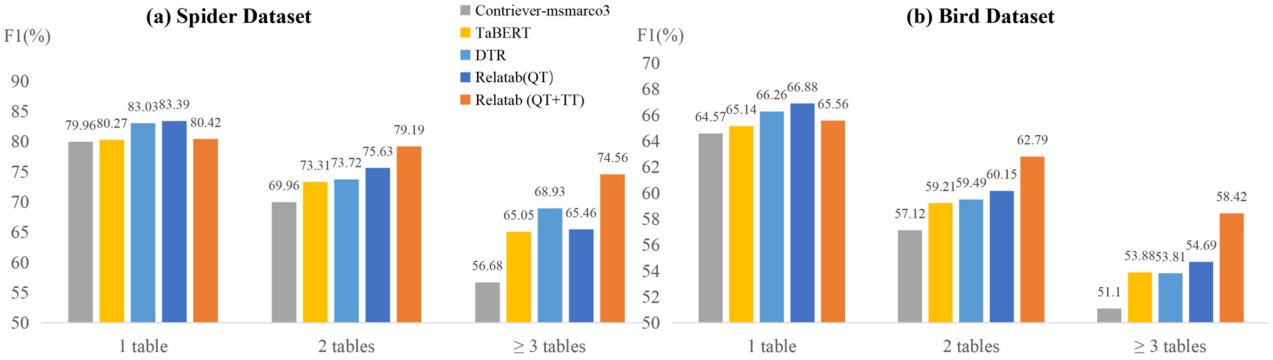


**Figure 7.** Effect of Inter-Table Similarity on Retrieval Performance with Varying Table Counts.

### *5.7. Hyperparameter Sensitivity Analysis*

To further validate the robustness of the Relatab model and investigate the impact of core hyperparameters on retrieval performance, a sensitivity analysis was conducted on the Spider, Bird, and CementingTables datasets. We employed the control variable method, fixing other parameters to their optimal values while varying the target parameter. The evaluation metric used was the Top-2 F1-score. The variation trends of model performance across different hyperparameter values are illustrated in Figure 8.

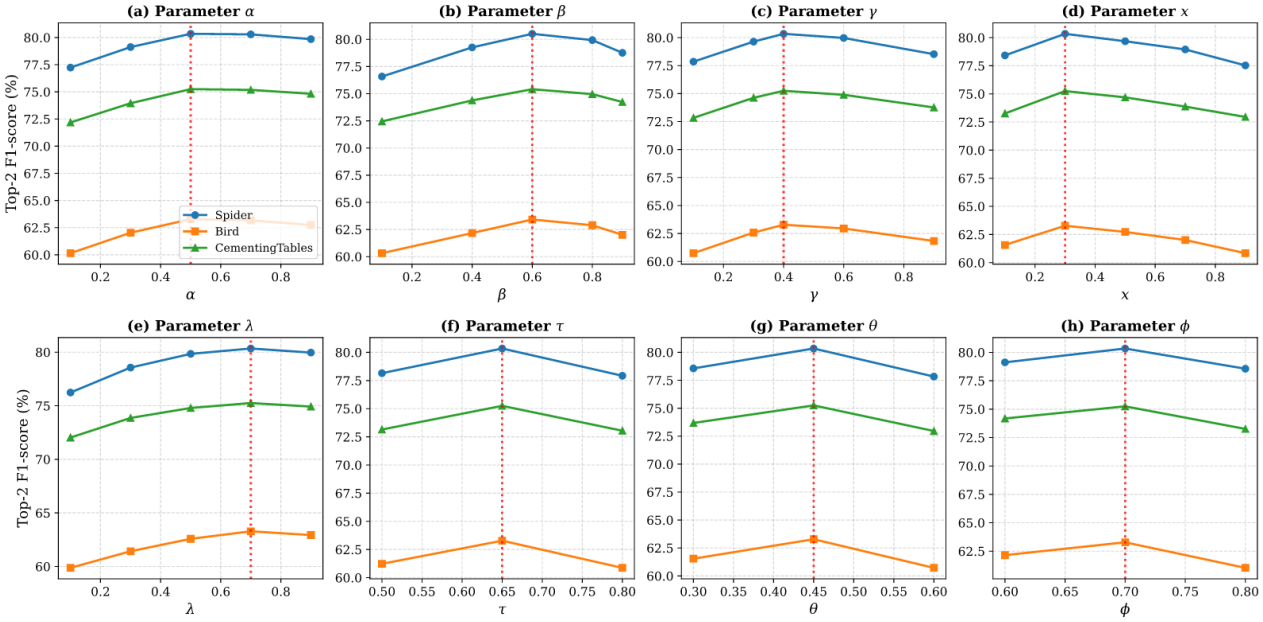


**Figure 8.** Sensitivity analysis of hyperparameter variations across Spider, Bird, and CementingTables datasets evaluated by Top-2 F1-score. The red dotted lines indicate the optimal hyperparameter values used in the main experiments.

#### 5.7.1. Impact of Weight Parameters ($\alpha, \beta, \gamma, x$)

The weight parameters primarily balance the contribution of different semantic levels. As shown in Figure 8a–d, the model performance exhibits an inverted U-shape trend for these parameters. For instance, $\alpha$ controls the balance between table title and column name matching. The model achieves its peak performance at $\alpha = 0.5$, indicating that an equal emphasis on global titles and local columns is optimal. Similarly, setting $\beta = 0.6$ effectively

balances the table-level and value-level relevance. The parameter $x$ determines the fusion ratio of the complementary inter-table score. When $x$ exceeds 0.5, the performance declines, suggesting that excessive reliance on complementary tables may introduce noise. Thus, $x = 0.3$ is selected to effectively capture implicit associations without overshadowing the direct query relevance.

#### 5.7.2. Impact of Decay Factor and Thresholds ($\lambda, \tau, \theta, \phi$)

Figure 8e demonstrates the effect of the multi-hop decay factor $\lambda$. The performance peaks at $\lambda = 0.7$ revealing a balance between effectively propagating necessary indirect associations and suppressing multi-hop noise. If $\lambda$ is too small (<0.5), crucial two-hop relationships are severed.

Threshold parameters ($\tau, \theta, \phi$) dictate the strictness of table filtering. As depicted in Figure 8f–h, increasing the thresholds initially improves the F1-score by filtering out irrelevant noise. However, excessively high thresholds lead to a sharp decline in Recall, as relevant tables are erroneously discarded. The optimal combination ($\tau = 0.65$, $\theta = 0.45$, $\phi = 0.7$) successfully strikes a balance between retrieval precision and coverage.

Overall, the sensitivity analysis reveals that the Relatab model exhibits strong robustness within a reasonable range ($\pm 0.1$) around the optimal hyperparameter values, proving that the proposed method does not rely on extreme parameter tuning to achieve high performance.

### *5.8. Computational Efficiency and Statistical Reliability*

To clarify practical suitability, Relatab separates offline and online computation. Offline processing constructs and prunes the inter-table similarity graph. Online retrieval requires encoding the query, computing direct query-tablequery–table relevance over cached table representations, and expanding only high-confidence neighbors from the initial table set. The dominant storage cost is the cache of 384-dimensional float embeddings; a single table-level vector requires about 1.5 KB, while value-level storage scales linearly with the number of retained column/value representations. The graph-propagation cost is controlled by dmax, theta, and pruning, which prevent exhaustive enumeration of all table paths. Because raw per-query runtime logs were not part of the manuscript record, this revision reports the computational mechanism and scaling behavior rather than unsupported wall-clock numbers. For statistical reliability, future implementations should use paired bootstrap or randomization tests over query-level predictions when full prediction logs are available. In this manuscript, performance gains are interpreted as consistent aggregate improvements across datasets, not as a replacement for query-level significance testing.

### *5.9. Case Study and Error Analysis*

The CementingTables example in Figure 6 illustrates a successful retrieval scenario. The query asks for the basic data of well BZ23. A direct matching model can retrieve the table whose caption is “Basic data”, but Relatab can additionally identify complementary tables connected through shared well identifiers, wellbore parameters, and cementing-depth fields. This behavior is useful when an answer requires both the basic well profile and related operation parameters. Error cases mainly occur when unrelated tables share common engineering terms or similar numerical ranges. For example, tables describing different operations may contain overlapping depth values, causing a weak semantic association to be amplified. The decay factor, threshold pruning, and dmax = 2 setting are therefore important for reducing false associations while preserving useful implicit links.

## 6. Limitations and Future Work

Although Relatab improves retrieval coverage, it has several limitations. First, its semantic matching quality depends on the SentenceBERT encoder; domain-specific terminology or abbreviations may still be misrepresented without petroleum-domain adaptation. Second, graph construction can become expensive for very large table collections if pruning or indexing is not applied. Third, the current method uses hand-designed propagation and thresholding rather than a learned graph model, which improves interpretability but may limit expressiveness. Fourth, the private CementingTables dataset cannot be fully released because of industrial data restrictions, so reproducibility depends on the public Spider and Bird transformations and on detailed reporting of the private annotation process. Future work will explore domain-adapted encoders, approximate nearest-neighbor indexing for large-scale deployment, integration with large language models for query expansion and table summarization, and query-level statistical testing when complete prediction logs are available.

## 7. Conclusions

This paper proposes a multi-table retrieval method based on implicit inter-table relationship inference, designed to address the challenges posed by unstructured data in the petroleum industry within complex query scenarios. By integrating query-tablequery–table relevance and inter-table correlation, this method achieves the effective mining of implicit multi-table associations.

Specifically, query-tablequery–table relevance combines table-level matching (titles and column names) with value-level matching (column values), enabling the preliminary localization of relevant tables based on the query. Meanwhile, inter-table correlation further reveals latent connections between tables that are not explicitly labeled, utilizing inter-table similarity and multi-hop chain relationships.

Experimental results demonstrate that this fusion strategy enables the model to capture deeper semantic associations within unstructured data environments lacking clear structured connections, thereby enhancing the comprehensiveness and accuracy of retrieval.

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

**Funding:** This work is supported by the open project of Sichuan Provincial Key Laboratory of Philosophy and Social Science for Language Intelligencein Special Education (No. YYZN-2024-4).

**Institutional Review Board Statement:** The study did not involve humans or animals. Ethical review and approval were waived for this study, due to its nature as a computational and theoretical research without involvement of human or animal subjects.

**Informed Consent Statement:** The study did not involve human participants; therefore, informed consent was not required.

**Data Availability Statement:** The data are not publicly available due to [privacy restrictions/commercial confidentiality].

**Conflicts of Interest:** Author Meng Cai and Zhigang Yang was employed by the company Daqing Oilfield Limited Company. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## References


1. Reiter, E.; Dale, R. Building applied natural language generation systems. *Nat. Lang. Eng.* **1997**, *3*, 57–87. [CrossRef]
2. Cafarella, M.J.; Halevy, A.Y.; Wang, D.Z.; Wu, E.; Zhang, Y. WebTables: Exploring the Power of Tables on the Web. In *Proceedings of the VLDB Endowment*; ACM: New York, NY, USA, 2008; Volume 1, pp. 538–549.
3. Cafarella, M.J.; Halevy, A.; Khoussainova, N. Data Integration for the Relational Web. In *Proceedings of the VLDB Endowment*; ACM: New York, NY, USA, 2009; Volume 2, pp. 1090–1101.
4. Venetis, P.; Halevy, A.; Madhavan, J.; Paşca, M.; Shen, W.; Wu, F.; Miao, G.; Wu, C. Recovering Semantics of Tables on the Web. In *Proceedings of the VLDB Endowment*; ACM: New York, NY, USA, 2011; Volume 4, pp. 528–538.
5. Zhang, S.; Balog, K. Ad hoc table retrieval using semantic similarity. In *Proceedings of the 2018 World Wide Web Conference*; ACM: Lyon, France, 2018; pp. 1553–1562.
6. Zelle, J.M.; Mooney, R.J. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*; AAAI Press: Portland, OR, USA, 1996; pp. 1050–1055.
7. Trabelsi, M.; Davison, B.D.; Heflin, J. Improved table retrieval using multiple context embeddings for attributes. In *Proceedings of the 2019 IEEE International Conference on Big Data (Big Data)*; IEEE: Angeles, CA, USA, 2019; pp. 1238–1244.
8. Chen, Z.; Jia, H.; Heflin, J.; Davison, B.D. Leveraging schema labels to enhance dataset search. In *Proceedings of the Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020*; Springer International Publishing: Lisbon, Portugal, 2020; pp. 267–280.
9. Lai, S.; Xu, L.; Liu, K.; Zhao, J. Recurrent convolutional neural networks for text classification. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*; AAAI Press: Austin, TX, USA, 2015; pp. 2267–2273.
10. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 4171–4186.
11. Yin, P.; Neubig, G.; Yih, W.T.; Riedel, S. TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 5116–5123.
12. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A robustly optimized BERT pretraining approach. In Proceedings of the Eighth International Conference on Learning Representations, Online, 26 April–1 May 2020.
13. Herzig, J.; Nowak, P.K.; Müller, T.; Piccinno, F.; Eisenschlos, J. TaPas: Weakly Supervised Table Parsing via Pre training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 4320–4333.
14. Wang, Z.; Dong, H.; Jia, R.; Li, J.; Fu, Z.; Han, S.; Zhang, D. Tuta: Tree-based transformers for generally structured table pre-training. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2021; pp. 1780–1790.
15. Iida, H.; Thai, D.; Manjunatha, V.; Iyyer, M. TABBIE: Pretrained Representations of Tabular Data. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 3446–3456.
16. Dey, R.; Salem, F.M. Gate-variants of gated recurrent unit (GRU) neural networks. In *Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems*; IEEE: Boston, MA, USA, 2017; pp. 1597–1603.
17. Deng, X.; Sun, H.; Lees, A.; Wu, Y.; Yu, C. Turl: Table understanding through representation learning. *ACM SIGMOD Rec.* **2022**, *51*, 33–40. [CrossRef]
18. Chen, D.; O'Bray, L.; Borgwardt, K. Structure-aware transformer for graph representation learning. In *Proceedings of the 39th International Conference on Machine Learning*; PMLR: Baltimore, MD, USA, 2022; pp. 3469–3489.
19. Zheng, M.; Feng, X.; Si, Q.; She, Q.; Lin, Z.; Jiang, W.; Wang, W. Multimodal Table Understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Bangkok, Thailand, 2024; pp. 9102–9124.
20. Ahmadov, A.; Thiele, M.; Eberius, J.; Lehner, W.; Wrembel, R. Towards a Hybrid Imputation Approach Using Web Tables. In *Proceedings of the Big Data Computing*; IEEE: Limassol, Cyprus, 2015; pp. 21–30.
21. Lehmberg, O.; Ritze, D.; Ristoski, P.; Meusel, R.; Paulheim, H.; Bizer, C. The Mannheim Search Join Engine. *J. Web Semant.* **2015**, *35*, 159–166. [CrossRef]
22. Sarma, A.D.; Fang, L.; Gupta, N.; Halevy, A.Y.; Lee, H.; Wu, F.; Xin, R.; Yu, C. Finding Related Tables. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*; ACM: Scottsdale, AZ, USA, 2012; pp. 817–828.
23. Chen, P.B.; Zhang, Y.; Roth, D. Is Table Retrieval a Solved Problem? Exploring Join-Aware Multi-Table Retrieval. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*; Association for Computational Linguistics: Bangkok, Thailand, 2024; pp. 2687–2699.

24. Lee, C.H.; Polozov, O.; Richardson, M. KaggleDBQA: Realistic Evaluation of Text-to-SQL Parsers. *ACM J. Exp. Algorithmics* **2021**, *26*, 1–15. [CrossRef]
25. Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; Grave, E. Unsupervised Dense Information Retrieval with Contrastive Learning. *arXiv* **2022**. [CrossRef]
26. Yu, T.; Zhang, R.; Yang, K.; Yasunaga, M.; Wang, D.; Li, Z.; Ma, J.; Li, I.; Yao, Q.; Roman, S.; et al. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Brussels, Belgium, 2018; pp. 3911–3921.
27. Agarwal, S. Data mining: Data mining concepts and techniques. In *Proceedings of the 2013 International Conference on Machine Intelligence and Research Advancement*; IEEE: Katra, India, 2013; pp. 203–207.
28. Madani, N.; Srihari, R.K.; Joseph, K. Domain Specific Question Answering Over Knowledge Graphs Using Logical Programming and Large Language Models. *arXiv* **2023**. [CrossRef]
29. Karpukhin, V.; Oguz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; Yih, W.T. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 6769–6781.
30. Herzig, J.; Müller, T.; Krichene, S.; Eisenschlos, J. Open Domain Question Answering over Tables Via Dense Retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 512–519.


**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.