# Unveiling Implicit Table Knowledge with *Question-Then-Pinpoint* Reasoner for Insightful Table Summarization

**Kwangwook Seo Jinyoung Yeo Dongha Lee**[†]
Yonsei University
{tommy2130,jinyeo,donalee}@yonsei.ac.kr

## Abstract

Implicit knowledge hidden within the explicit table cells, such as data insights, is the key to generating a high-quality table summary. However, unveiling such implicit knowledge is a non-trivial task. Due to the complex nature of structured tables, it is challenging even for large language models (LLMs) to mine the implicit knowledge in an insightful and faithful manner. To address this challenge, we propose a novel table reasoning framework ***Question-then-Pinpoint***. Our work focuses on building a plug-and-play table reasoner that can self-question the insightful knowledge and answer it by faithfully pinpointing evidence on the table to provide explainable guidance for the summarizer. To train a reliable reasoner, we collect table knowledge by guiding a teacher LLM to follow the coarse-to-fine reasoning paths and refine it through two quality enhancement strategies to selectively distill the high-quality knowledge to the reasoner. Extensive experiments on two table summarization datasets, including our newly proposed **InsTaSumm**, validate the general effectiveness of our framework.[1]

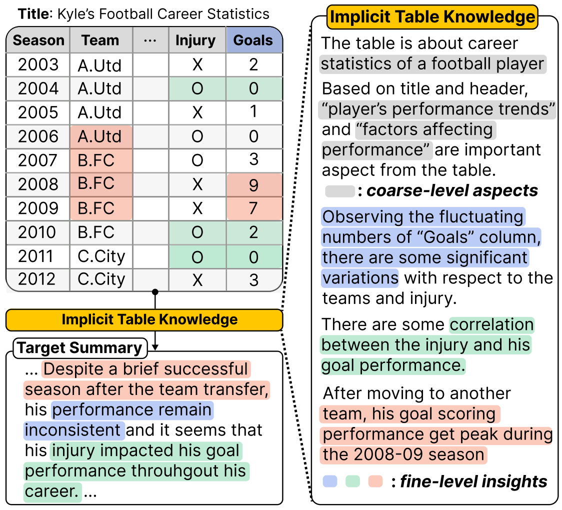


Figure 1: An example of implicit table knowledge that should be unveiled from explicitly stated table cells to generate the target summary.

## 1 Introduction

Table data has emerged as pivotal repositories of knowledge in facilitating data analysis, offering concise and structured representation of information. As comprehending the complex table can be time-consuming for human, text generation systems that can accurately summarize a provided table have the potential to greatly enhance the process of obtaining data insights.

In the task of table summarization (Lebret et al., 2016; Suadaa et al., 2021; Moosavi et al., 2021), a straightforward solution is to use neural model as an end-to-end summary generator. However, the model struggles to capture all necessary information in an end-to-end approach. The problem lies in that unlike table question answering tasks (Pasupat and Liang, 2015; Zhong et al., 2018) where explicit guidance (*i.e.*, input query) to search the answer is given, the table summarization task lacks direct control on what aspect of information should be searched from the table. Therefore, it is challenging for the model to decide a favorable choice of implicit evidence required for summarization only from the table input.

A line of research address this *uncontrollability* problem by injecting knowledge as a mediator to guide what kind of table contents should be stated in the text description. Most works adopt symbolic operations as guidance for sampling knowledge from the table to enhance the logical reasoning ability of the model. These symbolic operations include executable programs which mimics SQL (Liu et al., 2022a,b; Zhao et al., 2023a) or Logical Types (Zhao et al., 2023c; Perlitz et al., 2023) which categorize information seeking queries in several predefined types, serving as a control for knowledge collection. Some works (Su et al., 2021;

[†] Corresponding author

[1]Our code and dataset are available at https://github.com/tommyEzreal/QtP.



*Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12337–12362
November 12-16, 2024 ©2024 Association for Computational Linguistics

Guo et al., 2024a,b) use table as query for retrieving relevant knowledge from external source (*e.g.*, KB) to supplement the insufficient table knowledge of the language model. However, these approaches suffer from limited knowledge in terms of both diversity and coverage. Since symbolic operations heavily rely on rule-based sampling or predefined types, they remain insufficient for delivering comprehensive information within individual tables. Additionally, relying on external knowledge sources requires a rigorous assumption about the completeness of the knowledge base, which can lead to limited coverage of the required knowledge.

In this paper, we aim to unveil the implicit table knowledge by using the reasoning ability of the large language models (LLMs). As shown in Figure 1, the implicit knowledge required for generating the target summary should encompass multiple aspects of information that are scattered across table cells. We argue that by facilitating LLMs to directly mining these knowledge from the table with multiple reasoning paths, we can represent more diverse knowledge that can be used as informative insights to support the table summarization.

Despite the potential effectivness of this approach, there are two major challenges in using LLM as a knowledge miner. **(1) Low insightfulness**: As for the complex nature of structured table, the in-depth knowledge related to the target summary are not explicitly stated in the table cells. However LLMs tend to focus more on explicit textual cues (Chae et al., 2023) which often leads to surface-level realization and making it challenging to capture the insightful knowledge. **(2) Low faithfulness**: Since information that needs to be selected from the table to derive the reliable knowldege are scattered and hidden among irrelevant information, these often act as distracting factor or noise (Patnaik et al., 2024), leading to hallucination.

To address the aformentioned challenges, we introduce a novel table reasoning framework, *insightfully* ***Question then*** *faithfully* ***Pinpoint***. The key idea of our framework is to build a plug-and-play table reasoner that can self-quesiton-and-answer the insightful knowledge by faithfully pinpointing the evidence on the table to guide the table summarizer. To train the reasoner, we collect training data from teacher LLM which follows coarse-to-fine reasoning paths to mine in-depth knowledge required for summarization. During this step, we refine the data using two quality enhancement strategies to selectively distill the high-quality knowledge that is helpful to train a reliable reasoner.

For effective demonstration of our framework, we conduct experiments on two table summarization datasets, including our newly proposed dataset **INSTASUMM**. Since our table reasoner can be pluged-and-played to different variants of table summaization models, we validate that our framework can be applied to both fine-tuned and zero-shot summarization models with significant improvement. In addition, we evaluate our framework under out-of-domain setting, showing its robustness in diverse real-world scenarios.

## 2 Preliminaries

**Input Table Serialization** Following the recent works that employ language model on table-related tasks (Chen, 2023; Zhao et al., 2023b,c), we serialize the table input into a flattened sequence. Specifically, we use a vertical bar (|) to separate headers and cells in different columns, and a newline along with the row index to separate rows. This approach enables the direct input of structured tables into the language model. For example, the input table $t$ is serialized as follows:

*col : <header 1> | <header 2> | . . . row 1 : <cell value> | <cell value> | . . . row 2: <cell value> | <cell value> | . . .*

**Problem Formulation** Existing studies on table-to-text generation (Liu et al., 2022a; Zhao et al., 2022b) have mainly focused on pretraining the model with synthetic reasoning examples, and further fine-tuning them on downstream tasks in an end-to-end manner. Despite their progress, thsese models often struggle with generalizing to unseen domains (Chen, 2023) and consider intermediate knowledge only as a latent factor in the generation process, which poses issues of explainability.

Inspired by recent efforts (Ye et al., 2023; Cao et al., 2023; Kim et al., 2024; Ko et al., 2024) that incorporate auxiliary reasoning agents to empower the model's ability for downstream tasks, we extend the end-to-end table summarization setting by incorporating a table reasoner as an additional component in the generation process. This addition enables the externalization of implicit knowledge, which serve as explainable guidance for table summarization. Formally, given the serialized input table $t$, our reasoner focuses on generating table insights $\mathcal{I}$ as additional input knowledge to help

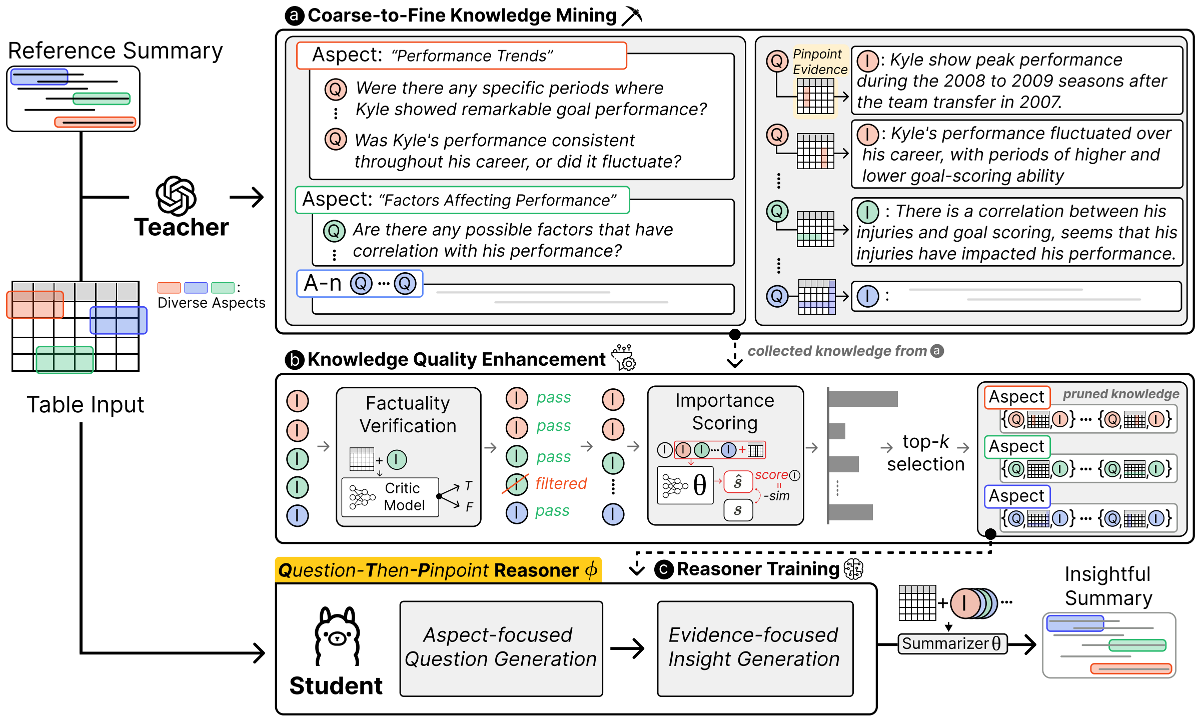


Figure 2: Overview of our framework. We (a) leverage LLM to collect diverse aspects of knowledge from the table and reference summary. For the collected knowledge, we (b) apply two quality enhancement strategy to construct a high-quality dataset for (c) training a reliable table reasoner $\phi$. During the inference phase (bottom right), the output insight $\mathcal{I}$ from the reasoner is provided to the summarizer $\theta$ as an additional input to guide the summarization.

summarizer $\theta$ in predicting the summary $s$:

$$s \sim P_\theta(\cdot|t, \mathcal{I}) \tag{1}$$

## 3 Proposed Framework

In this section, we propose a novel table reasoning framework *Question-Then-Pinpoint*, which focuses on building a table reasoner that can provide faithful insights supportive for summarization. The overall framework is illustrated in Figure 2.

### 3.1 Coarse-to-Fine Knowledge Mining

The goal of this step is to augment existing end-to-end table summarization training corpora $D = \{(t, s)\}$ with implicit table knowledge. To mine the knowledge that is helpful to infer the target summary, we leverage the capability of LLM to rationalize the target summary from the given table. As simply prompting LLM to generate intermediate reasoning paths often leads to surface-level knowledge, we provide several steps as checkpoints that guide the coarse-to-fine reasoning path for generating in-depth knowledge from the table.

Specifically, given the table $t$ and reference summary $s$, the teacher model $LLM_{teacher}$ first extracts coarse-level aspects $\mathcal{A} = \{a_n\}_{n=1}^N$, where $a_n$ represents one of the abstract topics across diverse aspects in the table. Then, conditioned on $\mathcal{A}$, $\mathcal{Q} = \{\mathcal{Q}_n\}_{n=1}^N$ is generated, where $\mathcal{Q}_n$ is a set of fine-level questions for each $a_n$ to query the information that should be captured from $t$.

$$\mathcal{A}, \mathcal{Q} = LLM_{teacher}(t, s) \tag{2}$$

After generating fine-level questions, we generate corresponding insights as answers for each question, along with relevant cell evidence to search for insights from the table. Specifically, we prompt LLM to answer the given questions $\mathcal{Q}$ to generate corresponding insights $\mathcal{I} = \{\mathcal{I}_n\}_{n=1}^N$. These insights are obtained by pinpointing the cell evidence $\mathcal{E} = \{\mathcal{E}_n\}_{n=1}^N$, where $\mathcal{E}$ is a set of relevant cell information that provides explicit evidence from $t$ to answer $\mathcal{Q}$. By focusing on explicit evidence, the model can faithfully capture implicit insights while avoiding distractions from irrelevant information.

$$\mathcal{E}, \mathcal{I} = LLM_{teacher}(t, s, \mathcal{Q}) \tag{3}$$

### 3.2 Knowledge Quality Enhancement

Symbolic knowledge distillation requires a strong teacher model to maximize the quality of the generated knowledge (West et al., 2022). However, some knowledge generated by LLMs may not align with the source table, resulting in unfaithful or redundant information that may harm the helpfulness of knowledge in summary generation. Therefore, we propose two knowledge quality enhancement strategies to filter the low-quality knowledge generated by the teacher model and selectively distill the high-quality knowledge to the student reasoner.

**Factuality Verification** Despite the clear pinpointing of evidence, some of the generated knowledge might still be unfaithful and not aligned with the source table. To effectively filter out this unfaithful knowledge, we adopt a critic model **C** to classify the counterfactual knowledge. Specifically, we use TAPEX (Liu et al., 2022b) trained on TabFACT (Chen et al., 2020b) dataset as the critic model to verify the generated insights. Given the source table $t$ and insight set $\mathcal{I}$, the model **C** do the binary classification on $i(\in \mathcal{I})$ to annotate each as consistent from $t$ or not. We then filter out those counterfactual insights from the training dataset.

**Importance Scoring** In the task of table summarization, as the provided input data should be mapped into the target in an encapsulated form, simply providing additional inputs may harm the summary output. That is, even if the additional knowledge is faithful, some of them might not be helpful to generate the target summary.

**Algorithm 1** Importance Scoring

**Input** Original insight set $\mathcal{I}$, Similarity measure $sim$, Table summarizer $\theta$, Source table $t$, Reference summary $s$

```
function ImportanceScoring(𝓘, sim, θ, t, s)
    scores ← {}
    for each i ∈ 𝓘 do
        𝓘̃ ← 𝓘 \ {i}
        ŝ ← θ(𝓘̃, t)
        score ← −sim(ŝ, s)
        scores[i] ← score
    end for            ▷ Loop ends after processing all i in 𝓘
    return scores
end function
```

To address this challenge, we examine the helpfulness of generated knowledge by scoring the importance of each insight (Algorithm 1). We assume that by iteratively evaluating the impact of removing each insight when inferring the target summary, we can check whether each insight is actually helpful in leading the model to output the target summary. Specifically, we first make subset $\tilde{\mathcal{I}}$ by removing $i(\in \mathcal{I})$ from the original set and infer table summary $\hat{s}$ from the source table $t$ conditioned on $\tilde{\mathcal{I}}$. We then measure the semantic similarity between the generated summary and the reference summary to compute the importance score for each ablated $i$ with respect to the negative of the similarity.

**Input Table** $t$**:** List of The Real Housewives of New Jersey episodes - Season 9 (2018–19)
[col] : No. overall | No. in season | Title | Original air date | U.S. viewers (millions) [row 1] ...

**Table Knowledge**: $(q_1, e_1, i_1), (q_2, e_2, i_2), ... \in (\mathcal{Q}, \mathcal{E}, \mathcal{I})$
• **Aspects** $\mathcal{A}$: $a_1$:Episode Highlights, $a_2$:Viewership Trends
• **Questions** $\mathcal{Q}$:
$q_1$:What are some of the standout moments or highlights from the episodes with the highest viewership?
$q_2$:Are there any noticeable patterns or fluctuations in viewership numbers across different episodes? ...
• **Evidences** $\mathcal{E}$:
$e_1$:col(Title, U.S.viewers), row(13)
$e_2$:col(No.in season, U.S.viewers), row(3,8,13) ...
• **Insights** $\mathcal{I}$:
$i_1$:The standout moment is Episode 13, titled "Camels, Cabo & Catfights" attracted highest viewership of 1.40 million.
$i_2$:There are fluctuations in viewership numbers across different episodes, with upward trends from certain episode. ...

**Reference Sumamry** $s$: ... Viewership numbers fluctuate throughout the season with some variations. Notably, there is a noticeable upward trend in viewership towards later episodes, culminating in Episode 13, titled "Camels, Cabo & Catfights," which attracted the highest viewership of 1.40 million. ... These patterns reflect audience engagement and preferences throughout the season, indicating particular episodes that resonated more strongly with viewers.

Table 1: An example of training set $D'$.

After repeating above process until all insights are scored, the top-$k$ insights are selected from each $\mathcal{I}_n(\subset \mathcal{I})$ for constructing the pruned training set. In the end, we construct an augmented training set $D' = \{(t, s, (\mathcal{A}, \mathcal{Q}, \mathcal{E}, \mathcal{I}))\}$ by mining and pruning the table knowledge $(\mathcal{A}, \mathcal{Q}, \mathcal{E}, \mathcal{I})$. We provide an example of processed dataset in Table 1.

### 3.3 QTP Reasoner Training

Using the annoated dataset $D'$, we train ***Question-then-Pinpoint*** **Reasoner** with two different objectives. We employ a single student model[2] for both question generation and insight generation, which are jointly trained on two instruction-tuning tasks.

**Aspect-focused Question Generation Task** The question generation task aims to generate fine-level questions to seek implicit knowledge from the table. Formally, given the source table $t$, our reasoner model $\phi$ is optimized to generate the sequence of

[2]In this work, we choose Llama2-7b (Touvron et al., 2023) as the backbone model for reasoner

$(\mathcal{A}, \mathcal{Q})$ pair by using the causal language modeling objective:

$$\mathcal{L}_{\text{QG}}(t, \mathcal{A}, \mathcal{Q}) = -\log p_\phi(\mathcal{A}, \mathcal{Q}|t) \quad (4)$$

**Evidence-focused Insight Generation Task** The insight generation task aims to predict insights by answering the given question, focusing on cell evidence. With the sequential prediction of both cell evidence $e$ and the corresponding $i$, the model can learn to capture faithful insights based on the pinpointed evidence. Given a question $q$ and source table $t$, the insight generation module aims to generate insight $i$ by answering the given $q$:

$$\mathcal{L}_{\text{IG}}(t, \mathcal{Q}, \mathcal{E}, \mathcal{I}) = -\sum_{(q,e,i)\in(\mathcal{Q},\mathcal{E},\mathcal{I})} \log p_\phi(e, i|t, q) \quad (5)$$

The final objective function of QTP reasoner is the combination of question and insight generation:

$$\mathcal{L}_{\text{Reasoner}} = \mathop{\mathbb{E}}_{(t,s,(\mathcal{A},\mathcal{Q},\mathcal{E},\mathcal{I}))\sim D'} [\mathcal{L}_{\text{QG}}(t, \mathcal{A}, \mathcal{Q}) + \mathcal{L}_{\text{IG}}(t, \mathcal{Q}, \mathcal{E}, \mathcal{I})] \quad (6)$$

# 4 Experiments

We conduct extensive experiments on summarization to demonstrate how the insights from QTP Reasoner provide useful guidance to the summarizer in generating high-quality table summaries.

## 4.1 Datasets

**In-domain** We first evaluate the performance on the test set held-out from the dataset for training QTP Reasoner. Since existing open-domain table-to-text generation datasets mainly focus on sentence-level generation (Chen et al., 2020a; Parikh et al., 2020) or are limited to specific domain (Liang et al., 2009; Wiseman et al., 2017; Suadaa et al., 2021), we require a more comprehensive testbed for evaluating our framework. Therefore, we build a refined version of an existing dataset named **INSTASUMM**, which focuses on generating **Ins**ightful **Ta**ble **Summ**ary solely from the input table in a paragraph format.

We adopt QTSumm (Zhao et al., 2023b) as a source dataset to construct INSTASUMM. QTSumm is a query-focused table summarization dataset, collected by human-annotated multiple queries and summaries for a single table input. As QTSumm considers informativeness when curating queries and covers diverse aspects with multiple query-summary pairs for each table, it consists of rich and in-depth information in the annotated descriptions compared to general table-to-text datasets. Hence, we construct INSTASUMM to comprise a paragraph-form summary for each individual table by aggregating diverse query-focused summaries from QTSumm. Instead of simply concatenating, we aggregate them into a single summary by prompting GPT-4 to verbalize it in a more fluent form. We provide detailed statistics of INSTASUMM in Appendix A.3.

**Out-of-domain** To further evaluate the generalizability of our framework, we choose **SciGEN** (Moosavi et al., 2021) as an out-of-domain dataset. SciGEN is a domain-specific table-to-text dataset, which is collected from scientific articles. It requires intensive reasoning to generate the long-form description from the given table. We use the test split of the medium setting for the experiment.

## 4.2 Evaluation Metrics

To evaluate the table summarization performance from multiple perspectives, we employ various automated evaluation metrics at four different levels. **(1) Surface-level**: We adopt SacreBLEU, ROUGE, METEOR, BERTScore, and A3CU (Zhao et al., 2023b; Post, 2018; Liu et al., 2023b) to evaluate both lexical overlap and contextual similarity between the reference and inferred summary. **(2) Faithfulness-level**: Following the previous works (Liu et al., 2022a; Zhao et al., 2023b,c), we use TAPAS-Acc and GPT4-Acc to evaluate the factual correctness of the generated summary. **(3) Insightfulness-level**: We use G-EVAL (Liu et al., 2023a) approach to evaluate the analytical depth of each summary. Specifically, we prompt GPT-4 to evaluate the insightfulness of the generated summary for the given table and summary pair in 1 to 5 Likert scale and report the average score. **(4) Pairwise quality comparison**: Following Dubois et al. (2024), we conduct a pairwise comparison where we present a source table and two summaries made by different models and ask GPT-4 to choose one based on diverse criteria. We adopt three criteria: which table summary is more *natural*, *comprehensive*, and *informative*. We provide the details of each metric and all the prompts used in the evaluation in Appendix A.6.

## 4.3 Table Summarizer

We consider both fine-tuned and zero-shot table summarization models for assessing the helpfulness of our reasoner in diverse scenarios. **(1) Fine-tuned Summarizer:** We consider two foundation models, **ReasTAP** (Zhao et al., 2022b) and **Llama-**

| Type | Methods | Surface-level | | | | | Faithfulness-level | | Insightfulness-level |
|---|---|---|---|---|---|---|---|---|---|
| | | S-BLEU | ROUGE-L | METEOR | BERTScore | A3CU | TAPAS-Acc | GPT4-Acc | G-EVAL |
| **INSTASUMM** (*In-domain*) | | | | | | | | | |
| ***fine-tuned*** *summarizer* | ReasTAP (Zhao et al., 2022b) | 9.34 | **33.70** | 33.45 | 87.67 | 29.61 | 68.18 | 69.55 | 2.79 |
| | + CoT Reasoner | 9.34 | 32.24 | 33.88 | 87.74 | 27.36 | 69.31 | 65.49 | 2.87 |
| | + Plan-and-Solve Reasoner | 9.03 | 31.91 | 34.62 | 87.60 | 30.51 | 67.04 | 70.82 | 2.43 |
| | + Logical Type Reasoner | 9.48 | 32.70 | 33.99 | 87.85 | 27.70 | 70.45 | 72.43 | 2.50 |
| | + SQL Reasoner | 9.11 | 32.38 | 33.30 | 87.67 | 27.35 | 70.90 | **73.72** | 2.74 |
| | **+ QTP Reasoner (ours)** | **10.83** | 32.68 | **36.35** | **88.25** | **31.25** | **75.68** | 73.04 | **3.05** |
| ***fine-tuned*** *summarizer* | Llama-2-7b (Touvron et al., 2023) | 16.47 | 29.70 | 36.05 | 89.23 | 26.72 | 78.86 | 74.61 | 2.95 |
| | + CoT Reasoner | 16.43 | 28.92 | 34.20 | 88.93 | 27.93 | 77.27 | 78.65 | 3.03 |
| | + Plan-and-Solve Reasoner | 17.39 | 29.98 | 35.18 | 88.78 | 26.49 | 78.40 | 77.89 | 2.87 |
| | + Logical Type Reasoner | 16.02 | 28.28 | 36.01 | 88.54 | 23.72 | 81.81 | 76.29 | 2.80 |
| | + SQL Reasoner | 17.70 | 28.35 | 33.24 | 88.62 | 24.57 | 79.54 | 80.48 | 2.79 |
| | **+ QTP Reasoner (ours)** | **19.48** | **31.79** | **40.29** | **89.76** | **30.28** | **85.90** | **84.83** | **3.34** |
| ***zero-shot*** *summarizer* | Mistral-7b (Jiang et al., 2023a) | 7.31 | 21.41 | 29.42 | 85.91 | 18.25 | 68.40 | 64.46 | 1.98 |
| | + CoT Reasoner | 7.60 | 20.26 | 26.26 | 86.46 | 15.41 | 70.90 | 73.24 | 2.24 |
| | + Plan-and-Solve Reasoner | 8.02 | **22.63** | 29.94 | 86.61 | 16.89 | 70.54 | 69.31 | 2.85 |
| | + Logical Type Reasoner | 7.33 | 20.53 | 28.03 | 86.41 | 16.53 | 71.36 | 73.63 | 2.75 |
| | + SQL Reasoner | 7.85 | 20.52 | 27.19 | 86.49 | 17.36 | 72.04 | 74.80 | 2.36 |
| | **+ QTP Reasoner (ours)** | **8.53** | 21.49 | **34.95** | **87.44** | **21.02** | **82.72** | **78.04** | **3.18** |
| ***zero-shot*** *summarizer* | GPT-3.5-turbo (OpenAI, 2023) | 7.64 | 23.45 | 28.43 | 87.12 | **23.96** | 72.49 | 76.74 | 1.94 |
| | + CoT Reasoner | 8.69 | 23.90 | 27.06 | 87.62 | 20.90 | 65.90 | 74.40 | 2.10 |
| | + Plan-and-Solve Reasoner | 9.77 | 24.10 | 29.04 | 87.77 | 21.87 | 65.45 | 71.11 | 2.74 |
| | + Logical Type Reasoner | 8.28 | 23.38 | 26.39 | 87.39 | 19.53 | 67.95 | 71.66 | 2.04 |
| | + SQL Reasoner | 8.92 | 24.22 | 26.69 | 87.61 | 22.50 | 70.68 | 79.35 | 2.32 |
| | **+ QTP Reasoner (ours)** | **11.38** | **25.04** | **34.17** | **88.07** | 23.48 | **88.63** | **85.33** | **3.52** |
| | + Self-Gen Knowledge | 9.31 | 24.08 | 29.35 | 87.53 | 21.23 | 86.59 | 84.26 | 3.40 |
| | + ORACLE Knowledge | 13.56 | 26.46 | 36.14 | 88.63 | 26.48 | 89.09 | 87.41 | 3.34 |

Table 2: In-domain summarization results on INSTASUMM testset.

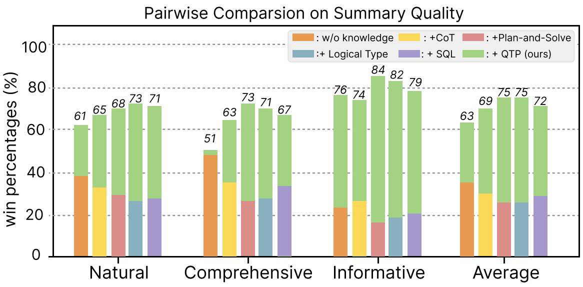


Figure 3: Pairwise **summary quality** comparison results on INSTASUMM using GPT-3.5 as backbone summarizer. We report the win percentage of QTP Reasoner.

**2-7b-chat** (Touvron et al., 2023). **(2) Zero-shot Summarizer:** For zero-shot evaluation, we consider two large-scale models, **GPT-3.5-turbo** (OpenAI, 2023), **Mistral-7b** (Jiang et al., 2023a). We provide details on each model in Appendix A.4.

Specifically, for both scenarios, we provide the knowledge generated by QTP Reasoner as an additional input. For fine-tuned summarizers, we augment the input during both the training and inference phases, while for zero-shot summarizers, we provide the knowledge during the inference.

### 4.4 Baselines

To evaluate how knowledge affects the performance of summarization, we compare QTP with the following baselines. **(1) Without knowledge:** We first consider the end-to-end baseline, where the summarizer directly predicts the target summary without externalization of implicit knowledge. **(2) Generate knowledge with step-by-step reasoning:** We consider another baseline that uses generic LLM reasoning to generate knowledge for augmenting the summarizer. Specifically, we implement two knowledge models, including CoT Reasoner (Yang et al., 2024; Kojima et al., 2022) and Plan-and-Solve (P&S) Reasoner (Wang et al., 2023), that generate implicit knowledge based on step-by-step reasoning. **(3) Generate knowledge with symbolic reasoning:** We then consider task-specific reasoners as baselines that guide knowledge generation with logical table operations. For Logical Type (LT) Reasoner, we adopt 9 predefined operation types[3] as control for knowledge generation, following Zhao et al. (2023c); Perlitz et al. (2023). For SQL Reasoner, we use SQL queries as guidance for generation, following the concept of Liu et al. (2022a,b); Ye et al. (2023); Cheng et al. (2022).

For a fair comparison, we train all baseline reasoners with the same backbone model as QTP Rea-

[3]Following Zhao et al. (2023c), we adopt Aggregation, Negation, Superlative, Count, Comparative, Ordinal, Unique, All, and Surface

| SciGEN (*Out-of-domain*) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Type** | **Methods** | **Surface-level** | | | | | **Faithfulness-level** | | **Insightfulness-level** |
| | | S-BLEU | ROUGE-L | METEOR | BERTScore | A3CU | TAPAS-Acc | GPT4-Acc | G-EVAL |
| ***zero-shot* *summarizer*** | Mistral-7b (Jiang et al., 2023a) | 1.89 | 14.09 | 20.09 | 82.42 | 6.91 | 71.19 | 69.80 | 1.73 |
| | + CoT Reasoner | 1.87 | 13.90 | 19.49 | 83.24 | 6.06 | 68.88 | 67.44 | 2.56 |
| | + Plan-and-Solve Reasoner | 2.08 | **14.13** | 22.03 | 83.33 | 8.03 | 69.36 | 69.53 | 3.08 |
| | + Logical Type Reasoner | 1.95 | 13.74 | 19.72 | 83.12 | 6.20 | 70.32 | 71.36 | 2.10 |
| | + SQL Reasoner | 1.90 | 13.21 | 19.50 | 83.07 | 7.49 | 72.73 | 74.85 | 2.03 |
| | **+ QTP Reasoner (ours)** | **2.22** | 13.67 | **23.51** | **83.82** | **9.19** | **75.59** | **75.49** | **3.35** |
| ***zero-shot* *summarizer*** | GPT-3.5-turbo (OpenAI, 2023) | 2.53 | 15.53 | 19.71 | 83.64 | 7.45 | 81.40 | 76.48 | 2.05 |
| | + CoT Reasoner | 2.41 | 15.39 | 18.38 | 84.11 | 7.07 | 75.91 | 78.44 | 3.08 |
| | + Plan-and-Solve Reasoner | **3.42** | 15.33 | 22.15 | 83.98 | 8.77 | 70.71 | 71.20 | 3.24 |
| | + Logical Type Reasoner | 2.49 | 15.20 | 18.67 | 83.89 | 6.86 | 79.96 | 81.34 | 2.54 |
| | + SQL Reasoner | 2.52 | 15.41 | 18.21 | 83.95 | 7.01 | 82.85 | 82.58 | 2.38 |
| | **+ QTP Reasoner (ours)** | 3.16 | **15.67** | **23.30** | **84.53** | **9.37** | **91.71** | **86.45** | **3.94** |

Table 3: Out-of-domain summarization results on SciGEN testset.

soner, by distilling the reasoning ability of LLM. We provide more details about implementations in Appendix A.5.

## 4.5 Main Results

**Comparison with end-to-end approach** We first compare our approach with different variants of end-to-end summary generation. Table 2 shows that summary conditioned on knowledge from QTP reasoner significantly improves the performance of both fine-tuned and zero-shot summarizers. From Figure 3, we find that summaries conditioned on knowledge from QTP Reasoner tend to be more natural, comprehensive, and informative. These suggest that augmenting the summarizer with the QTP Reasoner is beneficial for capturing related knowledge to produce a better-quality summary. In addition, the consistent improvements on different backbone summarizers demonstrates the general effectiveness of our approach.

**Comparison with other reasoner baselines** We compare QTP Reasoner with other knowledge-augmented baselines that generate knowledge with two different types of reasoning, *i.e.,* step-by-step reasoning (CoT, Plan-and-Solve) and symbolic reasoning (Logical Type, SQL). From Table 2 and Figure 3, we observe that incorporating baseline knowledge models into table summarization yields only marginal improvements, and in some metrics, it even underperforms compared to the end-to-end model. Specifically, we find that while symbolic reasoning enhances the factual correctness of the summary, it falls short of enhancing insightfulness. We also observe that although the Plan-and-Solve Reasoner achieves comparable performance on surface and insightfulness metrics to QTP Reasoner with multi-step reasoning, it still suffers from low faithfulness. In contrast, our model remains robust against this insightful-faithful trade-off by grounding the coarse-to-fine knowledge mining process in explicitly pinpointed evidence.

| QTP vs. | CoT | P&S | LT | SQL |
|---|---|---|---|---|
| Diverse | 59%* | 57%* | 52% | 69%* |
| Insightful | 73%* | 68%* | 76%* | 84%* |
| Faithful | 61%* | 66%* | 58%* | 51% |

Table 4: Human evaluation on **knowledge qulaity**. We report QTP's win percentages. (*: p-value $< 0.05$)

**Comparison with oracle and self-generated knowledge** To further demonstrate the efficacy of our approach, we compare the QTP Reasoner with two additional baselines. First, we augment the summarizer with ORACLE knowledge, which is obtained from the held-out test split in InsTaSumm, serving as the upper bound for summarization performance. Additionally, we introduce another setting where the teacher LLM directly generates knowledge to augment the summarizer without distillation to the student reasoner, which we refer to as Self-Generated Knowledge. From the results in Table 2, we find that QTP Reasoner-augmented summarizer shows comparable performance to the summarizer augmented with oracle knowledge, even without referencing the ground-truth summary. Moreover, augmenting the summarizer directly with knowledge generated by the teacher LLM does not result in better summary quality compared to QTP Reasoner. These results indicate that LLM-generated knowledge is not always helpful for summarization, highlighting the need for a selective distillation mechanism with a quality enhancement strategy to ensure the quality of the generated knowledge.

**Generalizability of QTP in out-of-domain scenario** In Table 3, we observe that our approach outperforms all baselines in out-of-domain scenarios, where the test domain is unseen during the training phase. This is attributed to QTP Reasoner's generalization ability, which stems from its flexibility of self-questioning the required knowledge from the unseen tables. While LLMs show remarkable generalization ability in diverse tasks, we find that they can still benefit from QTP Reasoner in capturing implicit knowledge that provides robust guidance for unseen domains.

### 4.6 Analysis

**QTP produces better-quality knowledge** To assess QTP Reasoner's ability to generate implicit knowledge from tables, we conduct a human evaluation on knowledge quality, focusing on three dimensions: *diversity*, *insightfulness*, and *faithfulness*. We randomly sample 100 reasoner inferences on the test set of INSTASUMM and ask 3 different human judges to compare the knowledge from QTP paired with baselines. We provide evaluation details in Appendix A.6. The results are shown in Table 4. We can see that while baselines achieve comparable performance in the diversity of knowledge against QTP, they usually struggle to generate insightful and faithful knowledge. This suggests that QTP generates better-quality knowledge that provides more in-depth and accurate analysis.

**Question guides the deeper analysis of the table, while evidence pinpoints faithful cues** To analyze the role of question and cell evidence, we perform ablation studies on knowledge generation. Specifically, we remove all questions from the dataset and train the model to generate insights along with cell evidence. Next, we ablate cell evidence and train the reasoner to directly predict insights for each question. The results are shown in Table 5. We observe that without questions, the summary quality drops significantly, especially in surface-level and insightfulness-level metrics. When ablating cell evidence, the performance decreases especially in faithfulness-level metrics. From these results, we posit that question plays a crucial role in providing specific guidance to the model for searching in-depth knowledge from the table, while the role of evidence is narrowing down the search space with clear pinpointing to avoid distractions from irrelevant cell values.

| Training | ROUGE-L | BERTScore | TAPAS-Acc | G-EVAL |
|---|---|---|---|---|
| QTP (full) | **25.04** | 88.07 | **88.63** | **3.52** |
| w/o question $\mathcal{Q}$ | 21.35 | 86.14 | 85.22 | 3.16 |
| w/o evidence $\mathcal{E}$ | 24.94 | **88.15** | 81.81 | 3.43 |
| w/o Fact Verif. | 24.15 | 87.93 | 86.36 | 3.38 |
| w/o Impt. Scoring | 23.59 | 87.12 | 87.50 | 3.20 |

Table 5: Ablation results on INSTASUMM using GPT-3.5-turbo as backbone summarizer.

**Knowledge Quality Enhancement leads to better-quality summary** To investigate the effect of knowledge quality enhancement strategies, we construct two different training data by omitting each strategy and training different versions of the reasoner. In Table 5, we find that factuality verification impacts more on faithfulness while importance scoring impacts on the surface and insightfulness metrics. These results suggest that a quality enhancement strategy for selecting key knowledge that factually aligns with the table is essential for training a reliable knowledge model.

**Case Study** We present an example summary from QTP paired with other baselines in Table 10. The table shows that QTP provides a more comprehensive analysis and offers detailed information compared to the baselines. While the baselines merely list the facts from the table, QTP-generated summary is well-structured with a logical flow that transitions seamlessly from a general overview to specific details, making it easier to follow the narrative. We present more examples in Appendix A.7.

## 5 Related Work

**Reasoning Over Table** Enhancing the reasoning ability of models has been explored in diverse table-related tasks, including TableQA, TableFV, and Table-to-Text. Existing works (Liu et al., 2022a; Zhao et al., 2022b) have mainly focused on pretraining the model with auxiliary reasoning tasks with large-scale corpora. Recently, some works (Zhao et al., 2023c; Ye et al., 2023; Yang et al., 2024) have shown the ability of LLMs in diverse table-based tasks through step-by-step reasoning. However, LLMs still suffer from unreliable predictions, leading to unfaithful or low-quality generation. Inspired by the recent works on knowledge distillation (West et al., 2022; Hsieh et al., 2023), we focus on building a reliable table reasoner, which selectively distills high-quality knowledge from the teacher LLM and helps the downstream task model as a plug-and-play module.

**Table-to-Text Generation** Recently, some works have adopted the knowledge-augmented approach in a table-to-text generation where the model is provided supplementary knowledge for the target text. Most of this knowledge is collected based on logical table operations (Zhao et al., 2023a,b; Perlitz et al., 2023), such as Logical Form and Logical Types, or retrieved from the external knowledge source (Guo et al., 2024a,b) to supplement the insufficient domain knowledge of the language model. Nevertheless, these can suffer from being constrained to certain logical types or limited coverage of external knowledge sources. Therefore, we propose QTP which uses LLM as a knowledge generator to represent more diverse and complex knowledge by directly mining the knowledge from the table.

## 6 Conclusion

In this paper, we propose *Question-then-Pinpoint*, a novel table reasoning framework that builds a plug-and-play table reasoner providing faithful insights supportive for table summarization. To achieve this, we mine the implicit table knowledge via coarse-to-fine reasoning paths and train the reasoner to self-question-and-answer the required knowledge by pinpointing the explicit evidence. We conduct extensive experiments on two different datasets including our newly proposed INSTASUMM, and demonstrate the general effectiveness of our framework compared to the baselines.

## Limitations

Despite the remarkable performance of our approach, several limitations remain, suggesting areas for future improvement. First limitation is the maximum sequence length limit occurred from input serialization. While we reduce noise in the table by explicitly pinpointing relevant evidence for each insight, our approach still requires to serialize the whole input table. However, real-world tables can be much longer than existing benchmark tables, exceeding the input length capacity of the language model. To address this, we plan to explore methods to reconstruct larger input tables into a more concise form that encapsulates compact information.

Second, our method and dataset currently do not explicitly handle multiple tables or hierarchical tables (*i.e.,*header or cells exhibits a multi-level structure) as input, which contain more complex structures than those used in our experiments. Considering these is important in terms of the applicability of our system in real-life scenarios where financial experts, such as analysts or investors, often refer to multiple hierarchical tables to obtain insightful conclusions (Zhao et al., 2022a). Therefore, extending our framework to effectively understand and process these hierarchical tables would be an important future direction.

Lastly, the reliable automated evaluation for the generated summary still remains as a challenge. Current table-related tasks often suffer from unreliable automated evaluation metrics that do not align well with human evaluations (Liu et al., 2022a, 2023b). While we employ certain automatic evaluation metrics (*e.g.,* faithfulness-level, insightfulness -level) that assess the quality of summaries beyond mere surface-level matching to references, these metrics may contain inherent biases (Dubois et al., 2024) that affect their ability to accurately measure the true faithfulness or insightfulness of the generated outputs. Recent studies on the fine-grained evaluating LMs shows remarkable progress across diverse tasks (Jiang et al., 2023b; Zhu et al., 2023; Kim et al., 2023) that better aligns with the human judgement with the customized evaluation criteria tailored for each tasks. These trends inspired us to develop a robust, table structure-aware evaluator that can assess the fine-grained quality of the generated outputs from diverse perspectives , aligning closely with human judgments in the future work.

## Ethical Consideration

Texts generation output of LLMs may include harmful, biased, or offensive content. However, we assert that in our research, this risk is largely minimized. The source tables and reference summaries in INSTASUMM are collected from QT-Summ (Zhao et al., 2023b), which is a publicly available dataset and have annotated by humans. We also check the generated knowledge with manual elimination of toxic, offensive or biased uses of lanaguage. For human evaluation, we hire three different judges from Amazon Mechanical Turk and guarantee fair compensation for each judge. We pay $0.15 for each unit task. The presented INSTASUMM dataset does not contain personal information that could lead to the identification of individuals or groups.

## Acknowledgement

This work was supported by the IITP grants funded by the Korea government (MSIT) (No. RS-2020-II201361; RS-2024-00457882, AI Research Hub Project), and the NRF grant funded by the Korea government (MSIT) (No. RS-2023-00244689).

## References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Yihan Cao, Shuyi Chen, Ryan Liu, Zhiruo Wang, and Daniel Fried. 2023. Api-assisted code generation for question answering on varied table structures. *ArXiv*, abs/2310.14687.

Hyungjoo Chae, Yongho Song, Kai Tzu-iunn Ong, Taeyoon Kwon, Minjin Kim, Youngjae Yu, Dongha Lee, Dongyeop Kang, and Jinyoung Yeo. 2023. Dialogue chain-of-thought distillation for commonsense-aware conversational agents. *arXiv preprint arXiv:2310.09343*.

Wenhu Chen. 2023. Large language models are few(1)-shot table reasoners. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1120–1130, Dubrovnik, Croatia. Association for Computational Linguistics.

Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, and William Yang Wang. 2020a. Logical natural language generation from open-domain tables. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7929–7942, Online. Association for Computational Linguistics.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020b. Tabfact: A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations*.

Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, Rahul Nadkarni, Yushi Hu, Caiming Xiong, Dragomir Radev, Mari Ostendorf, Luke Zettlemoyer, et al. 2022. Binding language models in symbolic languages. *arXiv preprint arXiv:2210.02875*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators.

Zhixin Guo, Minyxuan Yan, Jiexing Qi, Jianping Zhou, Ziwei He, Guanjie Zheng, and Xinbing Wang. 2024a. Adapting knowledge for few-shot table-to-text generation.

Zhixin Guo, Jianping Zhou, Jiexing Qi, Mingxuan Yan, Ziwei He, Guanjie Zheng, Zhouhan Lin, Xinbing Wang, and Chenghu Zhou. 2024b. Towards controlled table-to-text generation with scientific reasoning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9951–9955. IEEE.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023a. Mistral 7b.

Dongfu Jiang, Yishan Li, Ge Zhang, Wenhao Huang, Bill Yuchen Lin, and Wenhu Chen. 2023b. Tigerscore: Towards building explainable metric for all text generation tasks. *ArXiv*, abs/2310.00752.

Seoyeon Kim, Kwangwook Seo, Hyungjoo Chae, Jinyoung Yeo, and Dongha Lee. 2024. VerifiNER: Verification-augmented NER via knowledge-grounded reasoning with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2441–2461, Bangkok, Thailand. Association for Computational Linguistics.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, S. Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2023. Prometheus: Inducing fine-grained evaluation capability in language models. *ArXiv*, abs/2310.08491.

Sungho Ko, Hyunjin Cho, Hyungjoo Chae, Jinyoung Yeo, and Dongha Lee. 2024. Evidence-focused fact summarization for knowledge-augmented zero-shot question answering.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in*

*Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, page 611–626, New York, NY, USA. Association for Computing Machinery.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213, Austin, Texas. Association for Computational Linguistics.

Percy Liang, Michael Jordan, and Dan Klein. 2009. Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 91–99, Suntec, Singapore. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 150–157.

Ao Liu, Haoyu Dong, Naoaki Okazaki, Shi Han, and Dongmei Zhang. 2022a. PLOG: Table-to-logic pretraining for logical table-to-text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5531–5546, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022b. TAPEX: Table pre-training via learning a neural SQL executor. In *International Conference on Learning Representations*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023a. G-eval: Nlg evaluation using gpt-4 with better human alignment.

Yixin Liu, Alexander Fabbri, Yilun Zhao, Pengfei Liu, Shafiq Joty, Chien-Sheng Wu, Caiming Xiong, and Dragomir Radev. 2023b. Towards interpretable and efficient automatic reference-based summarization evaluation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16360–16368, Singapore. Association for Computational Linguistics.

Nafise Sadat Moosavi, Andreas Rücklé, Dan Roth, and Iryna Gurevych. 2021. Scigen: a dataset for reasoning-aware text generation from scientific tables. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

OpenAI. 2023. Chatgpt. `https://openai.com/blog/chatgpt`.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480, Beijing, China. Association for Computational Linguistics.

Sohan Patnaik, Heril Changwal, Milan Aggarwal, Sumit Bhatia, Yaman Kumar, and Balaji Krishnamurthy. 2024. CABINET: Content relevance-based noise reduction for table question answering. In *The Twelfth International Conference on Learning Representations*.

Yotam Perlitz, Liat Ein-Dor, Dafna Sheinwald, Noam Slonim, and Michal Shmueli-Scheuer. 2023. Diversity enhanced table-to-text generation via type control.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Yixuan Su, Zaiqiao Meng, Simon Baker, and Nigel Collier. 2021. Few-shot table-to-text generation with prototype memory. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 910–917, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lya Hulliyyatus Suadaa, Hidetaka Kamigaito, Kotaro Funakoshi, Manabu Okumura, and Hiroya Takamura. 2021. Towards table-to-text generation with numerical reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1451–1465, Online. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4602–4625, Seattle, United States. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Bohao Yang, Chen Tang, Kun Zhao, Chenghao Xiao, and Chenghua Lin. 2024. Effective distillation of table-based reasoning ability from LLMs. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5538–5550, Torino, Italia. ELRA and ICCL.

Yunhu Ye, Binyuan Hui, Min Yang, Binhua Li, Fei Huang, and Yongbin Li. 2023. Large language models are versatile decomposers: Decompose evidence and questions for table-based reasoning.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yilun Zhao, Yunxiang Li, Chenying Li, and Rui Zhang. 2022a. Multihiertt: Numerical reasoning over multi hierarchical tabular and textual data. In *Annual Meeting of the Association for Computational Linguistics*.

Yilun Zhao, Linyong Nan, Zhenting Qi, Rui Zhang, and Dragomir Radev. 2022b. ReasTAP: Injecting table reasoning skills during pre-training via synthetic reasoning examples. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9006–9018, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yilun Zhao, Zhenting Qi, Linyong Nan, Lorenzo Jaime Flores, and Dragomir Radev. 2023a. LoFT: Enhancing faithfulness and diversity for table-to-text generation via logic form control. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 554–561, Dubrovnik, Croatia. Association for Computational Linguistics.

Yilun Zhao, Zhenting Qi, Linyong Nan, Boyu Mi, Yixin Liu, Weijin Zou, Simeng Han, Xiangru Tang, Yumo Xu, Arman Cohan, et al. 2023b. Qtsumm: A new benchmark for query-focused table summarization. *arXiv preprint arXiv:2305.14303*.

Yilun Zhao, Haowei Zhang, Shengyun Si, Linyong Nan, Xiangru Tang, and Arman Cohan. 2023c. Investigating table-to-text generation capabilities of large language models in real-world information seeking scenarios. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 160–175, Singapore. Association for Computational Linguistics.

Victor Zhong, Caiming Xiong, and Richard Socher. 2018. Seq2SQL: Generating structured queries from natural language using reinforcement learning.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2023. Judgelm: Fine-tuned large language models are scalable judges. *ArXiv*, abs/2310.17631.

# A Experimental Details

## A.1 QtP Reasoner Training Data Generation

**Coarse-to-Fine Knowledge Mining** We prompt `GPT-3.5-turbo-0125` to generate the implicit knowledge for a given source table and the target summary with a 1-shot demonstration. We empirically confirmed that increasing the number of demonstrations does not necessarily guarantee higher-quality training data. The reason for this is that providing too many examples can lead to the generation of responses that closely follow the reasoning paths presented in those examples, especially when dealing with tables from similar domains. This tendency reduces the diversity of questions needed to gather a variety of insights. Therefore, we selected demonstrations that are not overly specific to any particular reasoning path and focused more on instruction that closely align with our desired output format. The prompt used for knowledge generation is shown in Table 13 and 14. To collect sufficient knowledge candidates, we generate five questions for each aspect during the initial generation.

**Knowledge Qulaity Enhancement** After all sets of candidate knowledge are generated, we apply two quality enhancement strategies to filter out the low-quality knowledge. We adopt TAPEX (Liu et al., 2022b) trained on TABFACT (Chen et al., 2020b) to verify the incorrect insights. TAPEX enhances the pre-training of the BART model with a vast corpus of synthetic SQL query execution data, improving its table comprehension and reasoning abilities. We use the huggingface checkpoint[4] of the TAPEX-large version for the experiment. Around 9% of insights are filtered out during the Factuality Verification process.

Subsequently, we apply another strategy called importance scoring (Algorithm 1) for the processed examples. We adopt Llama-2-7b-chat (Touvron et al., 2023) as summarizer $\theta$ to iteratively evaluate the impact of each ablated insight. In detail, we perform zero-shot inference conditioned on each ablated set of insights $\tilde{\mathcal{I}}$, since the model without prior knowledge of the table in the training data reacts more sensitively to each ablation. For similarity measure, we employ SBERT[5] to compute the similarity between the generated summary and reference summary.

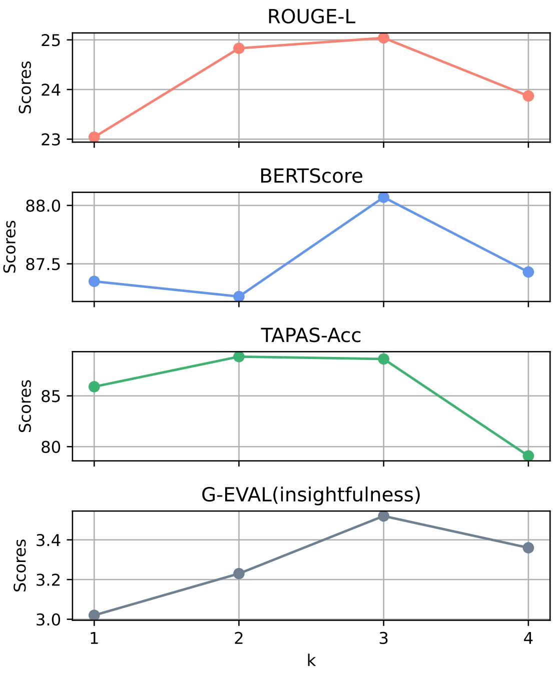


Figure 4: Summarization results with different $k$

After the scoring process, we select top-k insights from each aspect. We set $k$ to 3 in our experiments. To understand the effect of $k$, we perform an ablation study using different numbers of $k$ to prune the insights. Specifically, we adjust the number of insights in the training set in the range of 1 to 4 and train different reasoners for each $k$ to find the optimal $k$. The results are shown in figure 4.

In the end, we construct the pruned training set $D'$ which is an extended version of end-to-end table summarization training corpora $D$. We provide the statistics of the generated dataset in Table 6, and the filtered-out examples in Table 12.

| #Table | #Aspect $\mathcal{A}$ | #Question, #Evidence, #Insight $(\mathcal{Q}, \mathcal{E}, \mathcal{I})$ |
|---|---|---|
| 2,054 | 9,207 | 27,621 |

Table 6: Statistics of $D'$ used to train the QTP Reasoner.

## A.2 QTP Reasoner Training Details

We train QTP Reasoner with $D'$ on top of Llama-2-7b-chat (Touvron et al., 2023) model with two different instruction tunning tasks. Specifically, we randomly shuffle the instances from the aspect-focused question generation task and evidence-focused insight generation task, then jointly train them for a single model. To efficiently finetune

[4] https://huggingface.co/microsoft/tapex-large-finetuned-tabfact

[5] https://huggingface.co/sentence-transformers/stsb-xlm-r-multilingual

the model, we adopt 4-bit quantized QLoRA and set the parameters as $r = 64$, $\alpha = 16$. We use a constant learning rate schedule set at 2e-4, and train with the batch size of 4 on a single NVIDIA RTX A6000 GPU. The inference of the model is conducted using `vLLM` framework[6] (Kwon et al., 2023)

### A.3 Dataset Details

**INSTASUMM Construction** To gain a better tesbed for evaluating the insightful summarization performance, we build a refined version of an existing dataset, namely INSTASUMM. We adopt QTSumm (Zhao et al., 2023b) as a source dataset to construct INSTASUMM. We first collect all the tables and query-focused summaries in the train and test split of QTSumm, then aggregate it to a complete form of paragraph by prompting GPT-4 to verbalize it into a more fluent form. Table 7 shows the statistics of INSTASUMM.

**SciGEN** We choose SciGEN as an out-of-domain dataset to evaluate the generalization performance of QTP and other reasoner baselines. We use the test split of the medium setting for the experiments.

**Dataset Statistics** We provide dataset statistics of INSTASUMM and SciGEN in Table 7. We report the number of table instances (#Table) , average token length for each target summary (Avg.sum_len), and the average length of columns and rows in a single table (Avg.tab_len).

| Dataset | Domain | Split | #Table | Avg.sum_len | Avg.tab_len |
|---|---|---|---|---|---|
| INSTASUMM | open | train<br>test | 2,054<br>440 | 161.9 | #col: 6.6<br>#row: 11.71 |
| SciGEN | scientific | train<br>test | 13,607<br>1,038 | 115.3 | #col: 6.0<br>#row: 7.64 |

Table 7: Dataset Statistics of INSTASUMM and SciGEN.

### A.4 Table Summarizer

We consider two different table summarizers (*i.e.*, fine-tuned summarizer and zero-shot summarizer) for our experiments. For both scenarios, we provide the knowledge generated by QTP Reasoner as an additional input, concatenated to the serialized input table. For fine-tuned summarizers, we augment the input during both the training and inference phases, while for zero-shot summarizers, we augment the input only during the inference phase.

**Fine-tuned Summarizer** We adopt two different open-source models, ReasTAP and Llama-2-7b-chat.

- **ReasTAP**: ReasTAP (Zhao et al., 2022b) is a BART-based table-to-text model, pre-trained with synthetic table question and answering corpus. we use the official implementation of ReasTAP-large version from official GitHub[7] repository
- **Llama-2-7b-chat**: Llama-2-7b-chat[8] (Touvron et al., 2023) is specifically optimized for conversational contexts with the instruction tuning on the top of Llama-2.

**Zero-shot Summarizer** For zero-shot evaluation, we employ both open-source and closed-source LLM for the experiments. We adopt two large-scale models, GPT-3.5-turbo and Mistral-7b.

- **GPT-3.5-turbo**: GPT-3.5-turbo (OpenAI, 2023) is an instruction-tuned chat LLM with 175B parameters. It stands as a prominent closed-source model renowned for its generalization ability in diverse NLP tasks. We use `GPT-3.5-turbo-0125` version API provided by OPENAI.
- **Mistral-7b**: Mistral[9] (Jiang et al., 2023a) is an opensource LLM that outperforms Llama-2-13B in diverse evaluated NLP benchmarks.

### A.5 Baseline Resaoner Models

To evaluate how knowledge affects the performance of summarization, we compare QTP Reasoner with other knowledge-augmented baselines that generate knowledge with two different types of reasoning *i.e.,* step-by-step reasoning(CoT, Plan-and-Solve) and symbolic reasoning(Logical Type, SQL).

For a fair comparison, we implement all baseline knowledge reasoners with the same backbone model as QTP Reasoner. All reasoners are a student model trained on distilled knowledge which is generated by the teacher-LLM(GPT-3.5-turbo). We prompted LLM to generate training data for each reasoner from the reference summary and the input table with a 1-shot demonstration. Same from Section 3, the teacher model generates implicit

[6]All open-source LLM inference in our experiments are conducted using the `vLLM`.

[7]https://github.com/Yale-LILY/ReasTAP

[8]https://huggingface.co/meta-llama/Llama-2-7b-hf

[9]https://huggingface.co/mistralai/Mistral-7B-v0.1

knowledge from the table with different variants of reasoning strategy.

**CoT Reasoner** We first adopt Chain-of-Thought (Kojima et al., 2022) as the step-by-step reasoning strategy to generate the knowledge from the table. Specifically, the LLM is evoked to generate the reasoning step for each implicit knowledge with *"Let's think step by step"* prompt before the knowledge generation.

**Plan-and-Solve Reasoner** We then adopt Plan-and-Solve (Wang et al., 2023) for the variation of step-by-step reasoning, where the high-level plan is first generated to solve the knowledge generation task, and the final knowledge is generated according to the plan with step-by-step reasoning.

**Logical Type Reasoner** We adopt Logical Type Reasoner for the symbolic reasoning-based knowledge model baseline. Logical Types (Perlitz et al., 2023; Zhao et al., 2023c) are widely used schemes in recent table-to-text literature that categorizes several logical table operations to search the information on the table cells.

To apply the logical type in the process of knowledge generation, we adopt 9 predefined logical types (Negation, Superlative, Count, Comparative, Ordinal, Unique, All, and Surface) following Zhao et al. (2023c), to sample the knowledge from the table by using each type as a control for each knowledge generation. As simply providing all 9 types of knowledge from the table could be not helpful in generating the required knowledge for each reference summary, we first let the LLM choose the logical type that should be used to generates the reference summary. Then for each selected types, the model sequentially generate the corresponding table knowledge.

**SQL Resaoner** We adopt another symbolic reasoning baseline called SQL Reasoner, which uses SQL query for the intermediate control of the knowledge generation. Recent works (Cheng et al., 2022; Liu et al., 2022b; Ye et al., 2023; Liu et al., 2022a; Zhao et al., 2023a) in diverse table-based tasks have demonstrated that adopting the executable programs such as SQL or Logical Form shows remarkable performance improvement in table-related tasks. This is attributed to the complex nature of the structured table data, where logical programs can serve as faithful control for the information searching from the structured cells.

Therefore, we follow the concept of (Ye et al., 2023; Liu et al., 2022a; Zhao et al., 2023a) and let the LLM generate the implicit knowledge along with the SQL query for each knowledge. With the sequential prediction of each SQL query and knowledge, the model can be controlled to generate more faithful knoweldge (Liu et al., 2022a) that contains table-related logical reasoning in the generated description.

### A.6 Evaluation Details

We evaluate the performance of QTP and the baselines with both automatic evaluation and human evaluation. With automatic evaluation, we assess the quality of the generated summary, while with human evaluation, we assess the quality of generated knowledge from QTP Reasoner and the baseline reasoner models.

**Automatic Evaluation** We evaluate the performance of summarization with four different perspectives: (1) Surface-level, (2) Faithfulness-level, (3) Insightfulness-level, and (4) Pairwise quality comparison.

- **BLEU**: BLEU (Papineni et al., 2002) calculates the geometric mean of the precision over the n-grams of the output text. We utilized SacreBLEU (Post, 2018) to ensure consistent and reproducible BLEU scores.
- **ROUGE**: ROUGE (Lin and Hovy, 2003) evaluates word overlap between the candidate and reference summaries. We provided the F1 score for ROUGE-L, which considers the longest common subsequences.
- **METEOR**: METEOR (Banerjee and Lavie, 2005) focuses on a generalized concept of unigram matching between machine-generated translations and human reference translations.
- **BERTScore**: BERTScore (Zhang* et al., 2020) measures the similarity between the reference and the generated summary by using contextual word embeddings.
- **A3CU**: A3CU (Liu et al., 2023b) is an interpretable summarization evaluation system that aligns well with human judgments. It directly computes the similarity between texts without extracting atomic content units (ACUs) and uses the F1 score for evaluation.
- **TAPAS-Acc**: TAPAS-Acc (Liu et al., 2022a) is a reference-free metric that leverages TAPAS (Herzig et al., 2020) fine-tuned on the

TabFact (Chen et al., 2020b) dataset to assess the faithfulness of the generated content.

- **GPT4-Acc**: Following Zhao et al. (2023c), we assess the faithfulness of generated using the GPT-4 as the backbone. It shows a better correlation with human judgments than the TAPAS-Acc.
- **Insightfulness(G-EVAL)**: For insightfulness evaluation in Table 2, we adopt the G-EVAL approach to assess the insightfulness of each summary. We use a 5-point Likert scale to score each summary and report the average score. We provide an evaluation prompt in Table 18.
- **Pairwise Comparison**: For pairwise comparison in Figure 3, we use GPT-4 to evaluate the summary quality in diverse criteria. Specifically, GPT-4 is prompted to choose the better quality summary for each criterion among two candidates. We provide the prompt used in the evaluation in Table 20, 21 and 22. We adopt the following three criteria.
  - ***Natural***: the extent how naturally the information is conveyed, reflecting an easy and relaxed use of language that is clear and correct.
  - ***Comprehensive***: the extent to which the summary covers all the essential and important information presented in the source table.
  - ***Informative***: the extent to which the summary provides clear, accurate, and relevant information derived from the source table, contributing effectively to the understanding of the table.

**Human Evaluation** We assess the quality of generated knowledge with the human evaluation by comparing the output of QTP Reasoner and those from other reasoner baselines via Amazon Mechanical Turk (AMT). We show the interface for the evaluation in Figure 5. We ask three human judges to compare the quality of knowledge based on the following three criteria:

- ***Diverse*:** Which knowledge presents more diverse information from the table?
- ***Insightful*:** Which knowledge provides more in-depth analysis of the table?
- ***Faithful*:** Which knowledge is more accurate according to the table?

### A.7 Case Study

We select representative examples of QTP and present in Table 8 to 12

### A.8 Prompts

We show prompts used in our experiments in Table 13 to 22.

| **Example prediction of QTP from InsTaSumm** | |
|---|---|
| Input Table | title : 1990 - 91 Manchester United F.C. Season<br>col : Date \| Opponents \| H / A \| Result F - A \| Attendance<br>row 1 : 1 August 1990 \| Bury \| A \| 0 - 0 \| 7162<br>row 2 : 3 August 1990 \| Cork City \| A \| 0 - 0 \| 8000<br>row 3 : 5 August 1990 \| Waterford United \| A \| 4 - 0 \| 4750<br>row 4 : 8 August 1990 \| Derry City \| A \| 1 - 1 \| 9710<br>row 5 : 11 August 1990 \| Irish League \| N \| 3 - 0 \| 10037<br>row 6 : 13 August 1990 \| Bohemians \| A \| 3 - 0 \| 13878<br>row 7 : 15 August 1990 \| Rangers \| A \| 1 - 0 \| 31818<br>row 8 : 20 November 1990 \| Celtic \| H \| 1 - 3 \| 41658 |
| Generated Knowledge | **Aspect-a**: Stadium Attendance Trends<br>**Question-a$_1$**: How did the attendance numbers at matches held at Old Trafford evolve throughout the season? **Evidence-a$_1$**: The relevant columns and rows for the Question is col(Date, Attendance), row(5, 8). **Insight-a$_1$**: The attendance numbers at matches held at Old Trafford peaked at 41,658 for the game against Celtic on November 20, 1990. There were also matches with lower attendance figures, such as the match against the Irish League on August 11, 1990.<br>**Question-a$_2$**: What were the factors that influenced the fluctuations in attendance at home matches? **Evidence-a$_2$**: The relevant columns and rows for the Question is col(Date, Opponents, Attendance), row(8) **Insight-a$_2$**: The fluctuations in attendance at home matches could be influenced by various factors such as the opponents, the significance of the match, and the overall team performance.<br>**Question-a$_3$**: How did the attendance at home matches reflect the fan sentiment or support for Manchester United during that period? **Evidence-a$_3$**: The relevant columns and rows for the Question is col(Date, Opponents, Attendance), row(8) **Insight-a$_3$**: The attendance at home matches reflected varying levels of fan sentiment or support, with the highest attendance recorded in the game against Celtic, which ended in a 1-3 loss for Manchester United.<br><br>**Aspect-b**: Support from Fans<br>**Question-b$_1$**: Were there any events or milestones during the 1990-91 season that influenced attendance numbers? **Evidence-b$_1$**: The relevant columns and rows for the Question is col(Date, Opponents, Attendance), row(8). **Insight-b$_1$**: The 20 November 1990 match against Celtic at Old Trafford was a significant turnaround point for United, marking their first loss in the league and garnering the highest attendance for a single game during the season.<br>**Question-b$_2$**: How did Manchester United's fan base and support impact their overall performance and results during the season? **Evidence-b$_2$**: The relevant columns and rows for the Question is col(Opponents, Attendance), row(8). **Insight-b$_2$**: Manchester United's fan base and support seemed to positively impact their overall performance and results during the season, as evidenced by the high attendance numbers and successful matches, such as the one against Celtic.<br>**Question-b$_3$**: Were there any correlations between Manchester United's match outcomes and the attendance numbers at Old Trafford? **Evidence-b$_3$**: The relevant columns and rows for the Question is col(Date, Opponents, Result F - A, Attendance), row(8) **Insight-b$_3$**: Manchester United faced defeat against Celtic but still managed to secure a high attendance of 41,658, indicating strong fan support regardless of match outcomes.<br><br>**Aspect-c**: Results and Performance<br>**Question-c$_1$**: What were the key highlights of Manchester United's performance against Celtic in terms of results and attendance? **Evidence-c$_1$**: The relevant columns and rows for the Question is col(Date, Opponents, Attendance), row(8) **Insight-c$_1$**: Manchester United lost 1-3 to Celtic on November 20, 1990, with an attendance of 41,658.<br>**Question-c$_2$**: Were there any specific factors that influenced the attendance trends at Manchester United's home games? **Evidence-c$_2$**: The relevant columns and rows for the Question is col(Opponents, Attendance), row(6, 8). **Insight-c$_2$**: The high attendance at home games could be attributed to significant opponents like Bohemians and Celtic.<br>**Question-c$_3$**: Did Manchester United face any challenges in maintaining consistent attendance numbers throughout the season? **Evidence-c$_3$**: The relevant columns and rows for the Question is col(Date, Attendance), row(1,2,3,4,5,6,7,8). **Insight-c$_3$**: Manchester United faced fluctuations in attendance numbers throughout the season, with varying figures for different matches. |
| Reference Summary | The table detailing the 1990-91 Manchester United F.C. season provides a snapshot of key match details and attendance figures during that period. Notably, Manchester United faced Celtic on November 20, 1990, resulting in a 1-3 victory for Celtic, with an attendance of 41,658. Additionally, Manchester United played a match against an Irish League team on August 11, 1990, securing a 3-0 win with an attendance of 10,037. These records offer insights into the team's performance and fan engagement during the specified season, showcasing both victories and attendance numbers for the highlighted matches. |
| Generated Summary | During the 1990-91 season, Manchester United had a strong performance in their away games, securing notable victories against clubs like Waterford United, Celtic, and Bohemians. The high attendance at these away matches reflected the strong support from their fan base. However, their home game against Celtic on November 20, 1990, marked a significant turnaround point as it resulted in their first loss in the league, despite attracting the highest attendance of 41,658 for a single game during the season. The fluctuating attendance at home matches seemed to be influenced by factors such as the opponents, match significance, and overall team performance. Despite facing defeat against Celtic, Manchester United continued to receive strong fan support, as evidenced by the high attendance numbers. |

Table 8: Example of generated knowledge on InsTaSumm testset using GPT-3.5 as backbone summarizer.

| **Example prediction of QTP from SciGEN** | |
|---|---|
| Input Table | title(caption) : Table 4: All experiment results of transfer learning and abbreviation expansion (top-2 scores marked as bold). MedNLI (expanded) denotes MedNLI with abbreviation expansion.<br>col : Dataset ǀ BERT dev ǀ BERT test ǀ BioBERT dev ǀ BioBERT test<br>row 1 : MedNLI ǀ 79.56 ǀ 77.49 ǀ 82.15 ǀ 79.04<br>row 2 : MNLI (M) ǀ 83.52 ǀ - ǀ 81.23 ǀ -<br>...<br>row 8 : MedNLI (expanded) ǀ 79.13 ǀ 77.07 ǀ [BOLD] 83.87 ǀ 79.95<br>row 9 : S $\rightarrow$ M $\rightarrow$ MedNLI (expanded) ǀ [BOLD] 82.15 ǀ [BOLD] 79.95 ǀ 83.08 ǀ [BOLD] 81.85 |
| Generated Knowledge | **Aspect-a**: Interdataset Generalizability<br>**Question-a$_1$**:Are there any instances where models from one dataset were able to transfer well to another dataset, indicating strong interdataset generalizability? **Evidence-a$_1$**: The relevant columns and rows for the Question is col(BERT dev), row(1, 2). **Insight-a$_1$**: BERT models from the MedNLI dataset were able to transfer well to the MNLI (M) dataset, with an increase in performance on the MNLI (M) dataset.<br>**Question-a$_2$**: What factors contribute to the differences in model performance based on the dataset they were fine-tuned on? **Evidence-a$_2$**: The relevant columns and rows for the Question is col(BERT dev, BERT test, BioBERT dev, BioBERT test), row(1, 3, 4, 5, 6, 7, 8, 9). **Insight-a$_2$**: The choice of dataset plays a crucial role in determining the performance of the model.<br>**Question-a$_3$**: What implications do the varying scores have on the practical applications of models trained on different NLI datasets? **Evidence-a$_3$**: The relevant columns and rows for the Question is col(BERT dev, BERT test, BioBERT dev, BioBERT test), row(1, 3, 4, 5, 6, 7, 8, 9). **Insight-a$_3$**: The varying scores suggest that models trained on different NLI datasets have distinct strengths and potential applications in various NLP tasks requiring NLI knowledge.<br><br>**Aspect-b**: Transfer Learning Effectiveness<br>**Question-b$_1$**: How effective is transfer learning across different NLI datasets based on the provided scores? **Evidence-b$_1$**: The relevant columns and rows for the Question is col(BioBERT dev, BioBERT test), row(1, 3, 4, 5, 6, 7, 9). **Insight-b$_1$**: Transfer learning from BioBERT achieves the highest scores on both MedNLI and SNLI datasets, with slightly lower scores on MNLI (M).<br>**Question-b$_2$**: What challenges or advantages exist in transferring models pre-trained on MedNLI to other NLI datasets? **Evidence-b$_2$**: The relevant columns and rows for the Question is col(all), row(1,2,3,4,5,6,7, 8). **Insight-b$_2$**: Transferring models pre-trained on MedNLI to other NLI datasets presents challenges in generalization and adaptation due todataset-specific recurrent patterns. However, advantages include shared vocabulary and concepts, facilitating initial model alignment.<br>**Question-b$_3$**: How does the top-2 performance metric reflect the overall feasibility and impact of transfer learning in the NLI domain? **Evidence-b$_3$**: The relevant columns and rows for the Question is col(BERT dev, BERT test, BioBERT dev, BioBERT test), row(4, 7, 9). **Insight-b$_3$**: The top-2 performance metric reflects the success and capability of transfer learning in the NLI domain, with models trained on different datasets achieving high scores across various tasks.<br><br>**Aspect-c**: Cross-Dataset Transferability<br>**Question-c$_1$**: How does the top-2 performance metric evaluate the performance of models across different datasets? **Evidence-c$_1$**: col(BERT dev, BERT test, BioBERT dev, BioBERT test), row(7, 9) **Insight-c$_1$**: The top-2 performance metric evaluates the performance of models across different datasets by highlighting the top-scoring examples in each expansion scenario, indicating the highest-performing models for each dataset combination.<br>**Question-c$_2$**: What factors contribute to the differences in performance of models across datasets, as depicted in the table? **Evidence-c$_2$**: The relevant columns and rows for the Question is col(BERT dev, BERT test, BioBERT dev, BioBERT test), row(1, 3, 4, 5, 6, 7, 8, 9) **Insight-c$_2$**: The differences in performance of models across datasets could be attributed to variations in data and the specific challenges each dataset presents.<br>**Question-c$_3$**: How does transferring models from one dataset to another impact the overall performance and applicability of the models in the NLI domain? **Evidence-c$_3$**: The relevant columns and rows for the Question is col(BERT dev, BERT test, BioBERT dev, BioBERT test), row(4, 7, 9). **Insight-c$_3$**: Transferring models from one dataset to another generally results in an improvement in performance, indicating the potential benefits of transfer learning in the NLI domain. |
| Reference Summary | We conduct transfer learning on four different combinations of MedNLI, SNLI, and MNLI as it shown in the table 4 (line 4 to 7) and also add the results of general domain tasks (MNLI, SNLI) for comparison. BERT performs better on tasks in the general domain while BioBERT performs better on MedNLI which is in the clinical domain. positive transfer occurs on MedNLI. even though BioBERT is finetuned on general domain tasks before MedNLI, transfer learning shows better results than that fine-tuned on MedNLI directly. ... the accuracy of MNLI and SNLI on BioBERT is lower than the accuracy on BERT. The best combination is SNLI $\rightarrow$ MNLI $\rightarrow$ MedNLI on BioBERT. MedNLI (expanded) shows better performance than MedNLI on BioBERT while MedNLI works better on BERT (see table 4)... |
| Generated Summary | The table presents the experiment results of transfer learning and abbreviation expansion, with top-2 scores marked in bold. The performance of models on MedNLI (expanded) shows a slight decrease compared to the original MedNLI, with a decrease in test scores from 79.95 to 79.04. The BERT base model, bi-directional learning, ... Transfer learning from BioBERT achieves the highest scores on both MedNLI and SNLI datasets, with slightly lower scores on MNLI (M). Models trained on different NLI datasets have distinct strengths and potential applications in various NLP tasks requiring NLI knowledge. Transferring models from one dataset to another generally results in an improvement in performance, indicating the potential benefits of transfer learning in the NLI domain. |

Table 9: Example of generated knowledge on SciGEN testset using GPT-3.5 as backbone summarizer.

| **Example of Summary prediction from InsTaSumm** | |
|---|---|
| Input Table | title : 1990 - 91 Manchester United F.C. Season<br>col : Date \| Opponents \| H / A \| Result F - A \| Attendance<br>row 1 : 1 August 1990 \| Bury \| A \| 0 - 0 \| 7162<br>row 2 : 3 August 1990 \| Cork City \| A \| 0 - 0 \| 8000<br>row 3 : 5 August 1990 \| Waterford United \| A \| 4 - 0 \| 4750<br>row 4 : 8 August 1990 \| Derry City \| A \| 1 - 1 \| 9710<br>row 5 : 11 August 1990 \| Irish League \| N \| 3 - 0 \| 10037<br>row 6 : 13 August 1990 \| Bohemians \| A \| 3 - 0 \| 13878<br>row 7 : 15 August 1990 \| Rangers \| A \| 1 - 0 \| 31818<br>row 8 : 20 November 1990 \| Celtic \| H \| 1 - 3 \| 41658 |
| Without Knowledge | The table includes the date of the match, the opponents, whether the match was home or away, the result in terms of goals scored and conceded, and the attendance for each match. The season started with a series of away matches, with varying results and attendance numbers. The table also includes a home match against Celtic, which resulted in a 1-3 loss with a high attendance of 41658. |
| + CoT Reasoner | In the 1990 season, the team played a series of matches, with the highest attendance of 41,658 spectators recorded at the match against Celtic on 15 August 1990 at Celtic Park. The match against Rangers on 15 August 1990 at a Football Association (FA) shared ground had an attendance of 31,818 spectators. Notably, all the away matches resulted in draws, with the team playing against Bury, Cork City, Waterford United, Derry City, Irish League, Bohemians, and Rangers, with results of 0-0, 0-0, 4-0, 1-1, 3-0, 3-0, and 1-0 respectively. |
| + Plan-and-Solve Reasoner | During the 1990/91 season, Linfield F.C. had a mixed performance with three losses. The team played a series of matches, with a notable range in attendance numbers, from 13,378 to 42,500. The most significant attendance was recorded at 138,780 for their game against Rangers on 15 August 1990. The team had a variety of results, including draws and wins, with the most notable being a 5-0 victory against Irish League on 11 August 1990. |
| + Logical Type Reasoner | During the 1990-1991 season, a total of 8 games were played by the team. The matches were evenly split between home and away games, with the highest attendance recorded at 31818 during the away game against Rangers. The team had a strong start with a 4-0 victory against Waterford United and a 3-0 win against Bohemians. However, they faced a 1-3 defeat at home against Celtic, who outscored them with a score difference of 2-3. Overall, the team consistently attracted a high attendance, with no game having less than 10,000 spectators. |
| + SQL Reasoner | In August 1990, the football team played a series of matches, including away games against Bury, Cork City, Waterford United, Derry City, Bohemians, and Rangers, as well as a neutral game against the Irish League. The team achieved a 4-0 victory against Waterford United and a 3-0 victory against both the Irish League and Bohemians. The match against Rangers on August 15, 1990, had the highest attendance of 31,818 at Lansdowne Road Stadium. The team also played a home game against Celtic in November 1990, resulting in a 1-3 loss with an attendance of 41,658. |
| **+ QTP Reasoner** | During the 1990-91 season, Manchester United had a strong performance in their away games, securing notable victories against clubs like Waterford United, Celtic, and Bohemians. The high attendance at these away matches reflected the strong support from their fan base. However, their home game against Celtic on November 20, 1990, marked a significant turnaround point as it resulted in their first loss in the league, despite attracting the highest attendance of 41,658 for a single game during the season. The fluctuating attendance at home matches seemed to be influenced by factors such as the opponents, match significance, and overall team performance. Despite facing defeat against Celtic, Manchester United continued to receive strong fan support, as evidenced by the high attendance numbers. |

Table 10: Example of generated Summary on InsTaSumm testset using GPT-3.5 as backbone summarizer paired with different baselines.

<table>
<tr><th colspan="2">Example of Knowledge prediction from INSTASUMM</th></tr>
<tr><td>Input Table</td><td>title : 1990 - 91 Manchester United F.C. Season<br>col : Date | Opponents | H / A | Result F - A | Attendance<br>row 1 : 1 August 1990 | Bury | A | 0 - 0 | 7162<br>row 2 : 3 August 1990 | Cork City | A | 0 - 0 | 8000<br>row 3 : 5 August 1990 | Waterford United | A | 4 - 0 | 4750<br>row 4 : 8 August 1990 | Derry City | A | 1 - 1 | 9710<br>row 5 : 11 August 1990 | Irish League | N | 3 - 0 | 10037<br>row 6 : 13 August 1990 | Bohemians | A | 3 - 0 | 13878<br>row 7 : 15 August 1990 | Rangers | A | 1 - 0 | 31818<br>row 8 : 20 November 1990 | Celtic | H | 1 - 3 | 41658</td></tr>
<tr><td>CoT Reasoner</td><td>• The match against Celtic on 20 November 1990 at Celtic Park had the highest attendance of 41,658 spectators.<br>• The match against Rangers on 15 August 1990 at a Football Association (FA) shared ground had an attendance of 31,818 spectators.<br>• All the away matches ended in draws. The match against Cork City on 3 August 1990 ended in a draw with an attendance of 8,000 spectators.<br>• The game against Waterford United on 5 August 1990 saw an attendance of 4,750, with a final score of 4-0.<br>• On 8 August 1990, the match against Derry City ended in a 1-1 draw, attended by 9,710 fans.</td></tr>
<tr><td>Plan-and-Solve Reasoner</td><td>• Linfield F.C. had five losses during the 1990/91 season. There was a significant range in attendance numbers, from 13,878 to 47,500.<br>• The most notable attendance was 138,780 for their game against Rangers on 15 August 1990.<br>• The match against the Irish League on 11 August 1990, held at a neutral venue, ended with a score of 3-0 and had 10,037 spectators.<br>• A match against Bohemians on 13 August 1990, witnessed by 13,878 fans, concluded with a 3-0 result.<br>• The game against Bury on 1 August 1990 ended in a 0-0 draw, attracting 7,162 spectators.</td></tr>
<tr><td>Logical Type Reasoner</td><td>• The total number of games played during the 1990-1991 season was 8. There were no games where the attendance was less than 10,000.<br>• The match against Rangers had the highest attendance of 318,180. Celtic outscored the opponents with a score difference of 2-3.<br>• The match against Celtic on 20 November 1990, which was held at home, ended in a 1-3 result, with the highest attendance of 41,658 spectators.<br>• On August 1, 1990, the match against Bury saw an attendance of 7,162 spectators.<br>• The game against Bohemians on August 13, 1990, witnessed by 13,878 fans, recorded a significant attendance.</td></tr>
<tr><td>SQL Reasoner</td><td>• The match with the lowest attendance at Lansdowne Road Stadium was against Rangers on August 15, 1990.<br>• The match against Rangers on August 15, 1990 had the highest attendance of 318,180. The match against Celtic on November 20, 1990, held at Celtic Park, had the highest attendance of 41,350 spectators.<br>• On August 15, 1990, the game against Rangers, played at Lansdowne Road Stadium, drew the largest crowd of 31,818 fans.<br>• The lowest attendance of 8,000 was recorded at the match against Cork City on August 3, 1990.<br>• The match at home against Celtic on November 20, 1990, drew a significant crowd of 41,156 spectators.<br>• On August 15, 1990, the match against Rangers, which had the highest attendance of 31,818, was played at a shared ground.</td></tr>
<tr><td>QTP Reasoner</td><td>• Manchester United’s fan base and support seemed to positively impact their overall performance and results during the season, as evidenced by the high attendance numbers and successful matches, such as the one against Celtic.<br>• Manchester United faced defeat against Celtic but still managed to secure a high attendance of 41,658, indicating strong fan support regardless of match outcomes.<br>• The fluctuations in attendance at home matches could be influenced by various factors such as the opponents, the significance of the match, and the overall team performance.,<br>• The attendance numbers at matches held at Old Trafford peaked at 41,658 for the game against Celtic on November 20, 1990.<br>• There were also matches with lower attendance figures, such as the match against the Irish League on August 11, 1990.’,<br>• The attendance at home matches reflected varying levels of fan sentiment or support, with the highest attendance recorded in the game against Celtic, which ended in a 1-3 loss for Manchester United.,<br>• Manchester United faced fluctuations in attendance numbers throughout the season, with varying figures for different matches.</td></tr>
</table>

Table 11: Example of generated knowledge on InsTaSumm testset using GPT-3.5 as backbone summarizer.

| **Filtered-out training example in $D'$ after Knowledge Quality Enhancement** | |
|---|---|
| Input Table | title : 2008 in DREAM - Events list<br>col : # \| Event Title \| Date \| Arena \| Location \| Attendees \| Broadcast<br>row 1 : 7 \| Fields Dynamite!! 2008 \| December 31, 2008 \| Saitama Super Arena \| Saitama, Saitama, Japan \| 25,634 \| Tokyo Broadcasting System; HDNet<br>row 2 : 6 \| Dream 6: Middleweight Grand Prix 2008 Final Round \| September 23, 2008 \| Saitama Super Arena \| Saitama, Saitama, Japan \| 20,929 \| SkyPerfect; HDNet<br>row 3 : 5 \| Dream 5: Lightweight Grand Prix 2008 Final Round \| July 21, 2008 \| Osaka-jo Hall \| Osaka, Osaka, Japan \| 11,986 \| SkyPerfect; HDNet<br>row 4 : 4 \| Dream 4: Middleweight Grand Prix 2008 Second Round \| June 15, 2008 \| Yokohama Arena \| Yokohama, Kanagawa, Japan \| 14,037 \| SkyPerfect; HDNet<br>row 5 : 3 \| Dream 3: Lightweight Grand Prix 2008 Second Round \| May 11, 2008 \| Saitama Super Arena \| Saitama, Saitama, Japan \| 21,789 \| SkyPerfect; HDNet<br>row 6 : 2 \| Dream 2: Middleweight Grand Prix 2008 First Round \| April 29, 2008 \| Saitama Super Arena \| Saitama, Saitama, Japan \| 21,397 \| SkyPerfect; HDNet<br>row 7 : 1 \| Dream 1: Lightweight Grand Prix 2008 First Round \| March 15, 2008 \| Saitama Super Arena \| Saitama, Saitama, Japan \| 19,120 \| Tokyo Broadcasting System; HDNet |
| Mined & Pruned Knowledge | **Aspect-a**: Audience Engagement<br><br>**Insight-a$_1$**: The total attendance at DREAM events in 2008 was 125,522.<br>*(Factuality Verification filtered)*<br><br>**Insight-a$_2$**: The audience size varied based on the type of event or tournament, with some events attracting higher attendance figures than others.<br><br>**Insight-a$_3$**: It was broadcasted by Tokyo Broadcasting System and HDNet, ensuring widespread coverage and viewer engagement both locally in Japan and internationally.<br>*(Importance Scoring filtered)*<br><br>**Insight-a$_4$**: The broadcast of events by different networks like Tokyo Broadcasting System and HDNet may have influenced the overall audience engagement with DREAM events in 2008.<br><br>**Insight-a$_5$**:The attendance figures, especially the high numbers at Saitama Super Arena events, reflect the strong interest and appeal of DREAM events to the public in 2008.<br><br>**Aspect-b**: Geographical Spread<br><br>**Insight-b$_1$**: Three different venues hosted DREAM events in 2008: Saitama Super Arena in Saitama, Osaka-jo Hall in Osaka, and Yokohama Arena in Kanagawa.<br>*(Importance Scoring filtered)*<br><br>**Insight-b$_2$**: This global reach underscored the growing popularity of MMA and the organizational prowess of DREAM in staging high-profile events that captured the attention of diverse audiences across different continents.<br>*(Importance Scoring filtered)*<br><br>**Insight-b$_3$**: The geographical spread of events did impact the overall audience engagement, with Saitama Super Arena attracting the highest attendance figures compared to Osaka-jo Hall and Yokohama Arena.<br><br>**Insight-b$_4$**: The Saitama Super Arena emerged as the most frequented location, with impressive attendance figures, possibly due to its popularity and draw compared to other venues like Osaka-jo Hall and Yokohama Arena<br><br>**Insight-b$_5$**: The attendance at events in Saitama, particularly at the Saitama Super Arena, was significantly higher compared to events in Osaka and Kanagawa, indicating varying levels of audience engagement across these locations.<br><br>... |
| Reference Summary | The table "2008 in DREAM - Events list" provides a detailed overview of the DREAM events held in 2008, showcasing the distribution of events across different arenas and their respective attendance figures. Notably, the majority of events (four) took place at the Saitama Super Arena, attracting a total of 108,869 attendees, while two events were hosted in other arenas with a combined attendance of 26,023. This stark contrast in attendance between events at the Saitama Super Arena and other venues highlights the popularity and draw of the former location. Additionally, the geographical spread of DREAM events in 2008 encompassed three distinct venues: Saitama Super Arena in Saitama, Osaka-jo Hall in Osaka, and Yokohama Arena in Kanagawa. Among these, the Saitama Super Arena emerged as the most frequented location, with an impressive attendance of 25,634. Furthermore, the televised events in 2008, namely Fields Dynamite!! 2008 and Dream1:Lightweight Grand Prix 2008 First Round, broadcasted by the Tokyo Broadcasting System, attracted significant audiences of 25,634 and 19,120 attendees respectively. This data underscores the diverse event landscape of DREAM in 2008 and sheds light on the varying levels of audience engagement across different venues and broadcasts. |

Table 12: Filtered-out example from pruned dataset $D'$ after the knowledge quality enhancement.

| Coarse-to-fine Knowledge Mining (Generate $\mathcal{A}$, $\mathcal{Q}$) |
|---|
| **[Task Description]**<br>Generate intermediate knowledge from the table for generating the target summary("Summary:"). The knowledge should contain (Coarse-level Aspects) and (Fine-level Questions for each Aspects). These knowledge should be a crucial cue to generate the target summary, but you should not explicitly state the target summary and also pretend that you don't know the target summary. Generate five Fine-level Questions for each Aspects. The questions must be the list of five different strings. Use Example 1 as reference, respond to Example2.<br><br>**[Example 1]**<br>**Table:** *title + serialized table*<br>**Summary:** *target summary*<br><br>**Knowledge:**<br>*(Coarse-level Aspect): aspect 1*<br>*(Fine-level Questions):*<br>*Q1-1, Q1-2, ... Q1-5*<br>...<br>*(Coarse-level Aspect): aspect $n$*<br>*(Fine-level Questions):*<br>*Q$n$-1, Q$n$-2, ... Q$n$-5*<br><br>**[Example 2]**<br>**Table:** {table}<br>**Summary:** {summary}<br><br>**Knowledge:** |

Table 13: The prompt used for Coarse-to-Fine Knowledge Mining (Aspect and Question).

| Coarse-to-fine Knowledge Mining (Generate $\mathcal{E}, \mathcal{I}$) |
|---|
| **[Task Description]**<br>First, find answer for each given questions from the given table. Refer to the summary if it is needed. Then, select releavant column and rows from the table to answer the given questions from the table. Be careful not to omit any questions, answer all questions provided. Use Example 1 as reference, respond to Example2.<br><br>**[Example 1]**<br>**Table:** *serialized table*<br>**Summary:** *target summary*<br><br>**Questions:**<br>*Q-1, Q-2, ... Q-5*<br>**Question, Relevant column & row , Answer triples:** *(Q-1, E-1, I-1)*<br><br>**[Example 2]**<br>**Table:** {table}<br>**Summary:** {summary}<br><br>**Questions:** {questions}<br><br>**Question, Relevant column & row , Answer triples:** |

Table 14: The prompt used for Coarse-to-Fine Knowledge Mining (Evidence and Insight).

| Reasoner Training Instruction prompt (Aspect-focused Question Generation Task) |
|---|
| **[Task Description]**<br>Generate the intermediate knowledge for making the insightful summary of the table. First, the Knowledge should contain multiple (Coarse-level Aspect). Then for each single (Coarse-level Aspect), multiple (Fine-level Question) should be generated. These knowledge should be a crucial cue to generate the insightful summary.<br><br>**Table:** {table}<br>**Response:** |

Table 15: The instruction prompt used for training reasoner in Aspect-Focused Question Generation Task

| Reasoner Training Instruction prompt (Evidence-focused Insight Generation Task) |
|---|
| **[Task Description]**<br>Your task is find insights from the table by answering for the given Question. You will be provided a Table and Question. Before finding answer from the table, carefully look up the given table and pinpoint the relevant columns and rows as the evidence for the answer.<br><br>**Table:** {table}<br>**Questions:** {question}<br>**Response:** |

Table 16: The instruction prompt used for training reasoner in Evidence-Focused Insight Generation Task

| Zero-shot Summariazation prompt |
|---|
| **[Task Description]**<br>Generate Summary of the given table based on the given Knowledge. You can refer to the Knowledge to make a complete paragraph-form summary.<br><br>**Table:** {table}<br>**Knowledge:** {generated _knowledge}<br>**Summary:** |

Table 17: The prompt used for zero-shot summarization. We provide generated knowledge from QTP and other reasoner baselines as the additonal input.

| Insightfulness-level Evaluation prompt (G-EVAL) |
|---|
| **[Task Description]**<br>You will be given one summary written for a table. Your task it to rate the summary on one metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.<br><br>Evaluation Criteria:<br>Analytical Depth (1-5) - the extent to which the summary provides insightful interpretations, explains the implications, and contextualizes the information from the source table.<br><br>Evaluation Steps:<br>1. Examine the source table thoroughly, identifying the main topic, all key data points, and potential implications or contexts that could be drawn from the data.<br>2. Review the summary and assess its depth in analyzing the table. Determine if the summary offers deeper insights beyond the basic data, including interpretations of the implications, the significance of the information, and any contextual analyses.<br>3. Assign a score for Analytical Depth on a scale of 1 to 5, where 1 indicates a minimal or superficial analysis with no significant insights beyond the direct data, and 5 indicates a highly insightful summary that effectively delves into the broader implications, contextual significances, and thorough interpretations of the information from the table.<br><br>**Source Table:** {table}<br>**Summary:** {summary}<br>**Evaluation Form(scores ONLY):**<br>**Analytic Depth:** |

Table 18: The prompt for insightfulness-level evaluation. We adopt G-EVAL (Liu et al., 2023a) approach.

**Faithfulness-level summary evaluation (GPT4-Acc)**

**[Task Description]**
Your task is to evaluate whether the given summary faithfully state the facts from the given source table. Read the table and check whether each single sentence in the summary is true or false. Think step-by-step to verify each sentence. Use Example1 as reference, respond to Example2.

**[Example 1]**
**Table:** *table*
**Summary:** *generated_summary*

**Evaluation:**
(Sentence1): *sent1*
(explanation): *explanation*
(Verification): *T/F*
...
(Sentence $n$): *sent* $n$
(explanation): *explanation*
(Verification): *T/F*

**[Example 2]**
**Table:** {table}
**Summary:** {summary}

**Evaluation:**

Table 19: The prompt for faithfulness-level summary evaluation.

**Pairwise Summary Quality Comparison (Criteria: Comprehensive)**

**[Task Description]**
You will be given two different summary written for a table. Your task it to rate each summary on one metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.

Evaluation Criteria:
Comprehensiveness - the extent to which the summary covers all the essential and important information presented in the source table.

Evaluation Steps:
1. Examine the source table thoroughly, identifying the main topic and all key points presented.
2. Review the two different summary and compare each with the source table. Determine if the summary encapsulates all essential information without omitting any significant details.
3. Assign two score for comprehensiveness on a scale of 1 to 5 for each summary, where 1 indicates a minimal coverage of information and 5 indicates a comprehensive summary that includes all essential details from the source table
4. Choose the better summary in terms of the comprehensiveness score.

**Source Table:** {table}

**Summary A:** {summary_A}

**Summary B:** {summary_B}

Generate Your assessment in the following format:
**Better Summary Index: [Index]**(Use A for select Summary A and B for select Summary B)

Table 20: The prompt for pairwise summary quality comparison (criteria: comprehensive)

| Pairwise Summary Quality Comparison (Criteria: Informative) |
|---|
| **[Task Description]**<br>You will be given two different summary written for a table. Your task it to rate each summary on one metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.<br><br>Evaluation Criteria:<br>Informativeness - the extent to which the summary provides clear, accurate, and relevant information derived from the source table, contributing effectively to the understanding of the table.<br><br>Evaluation Steps:<br>1. Carefully examine the source table to understand the main topic and capture all pertinent information.<br>2. Review each of the two summaries to assess how effectively they convey the key information from the source table. Evaluate if the summaries provide clear and relevant information that aids in understanding the main points.<br>3. Assign a score for informativeness on a scale of 1 to 5 for each summary, where 1 indicates that the summary provides minimal or unclear information, and 5 indicates that the summary offers clear, accurate, and relevant information enhancing understanding of the main topic.<br>4. Determine and indicate which summary is better in terms of informativeness by comparing the assigned scores.<br><br>**Source Table:** {table}<br><br>**Summary A:** {summary_A}<br><br>**Summary B:** {summary_B}<br><br>Generate Your assessment in the following format:<br>**Better Summary Index: [Index]**(Use A for select Summary A and B for select Summary B) |

Table 21: The prompt for pairwise summary quality comparison (criteria: informative)

| Pairwise Summary Quality Comparison (Criteria: Natural) |
|---|
| **[Task Description]**<br>You will be given two different summary written for a table. Your task it to rate each summary on one metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.<br><br>Evaluation Criteria:<br>Naturalness - the extent to which the summary reads naturally, reflecting an easy and relaxed use of language that is clear and correct.<br><br>Evaluation Steps:<br>1. Thoroughly understand the source table, including its data and context.<br>2. Review each of the two summaries, assessing how naturally the information is conveyed. Focus on the use of language, including syntax, semantics, and overall coherence.<br>3. Assign a naturalness score from 1 to 5 for each summary. A score of 1 indicates that the summary is unnatural or difficult to understand, while a score of 5 suggests that the summary reads clearly and smoothly, with a natural flow.<br>4. Compare the scores to determine which summary best achieves naturalness in its language and presentation.<br><br>**Source Table:** {table}<br><br>**Summary A:** {summary_A}<br><br>**Summary B:** {summary_B}<br><br>Generate Your assessment in the following format:<br>**Better Summary Index: [Index]**(Use A for select Summary A and B for select Summary B) |

Table 22: The prompt for pairwise summary quality comparison (criteria: natural)

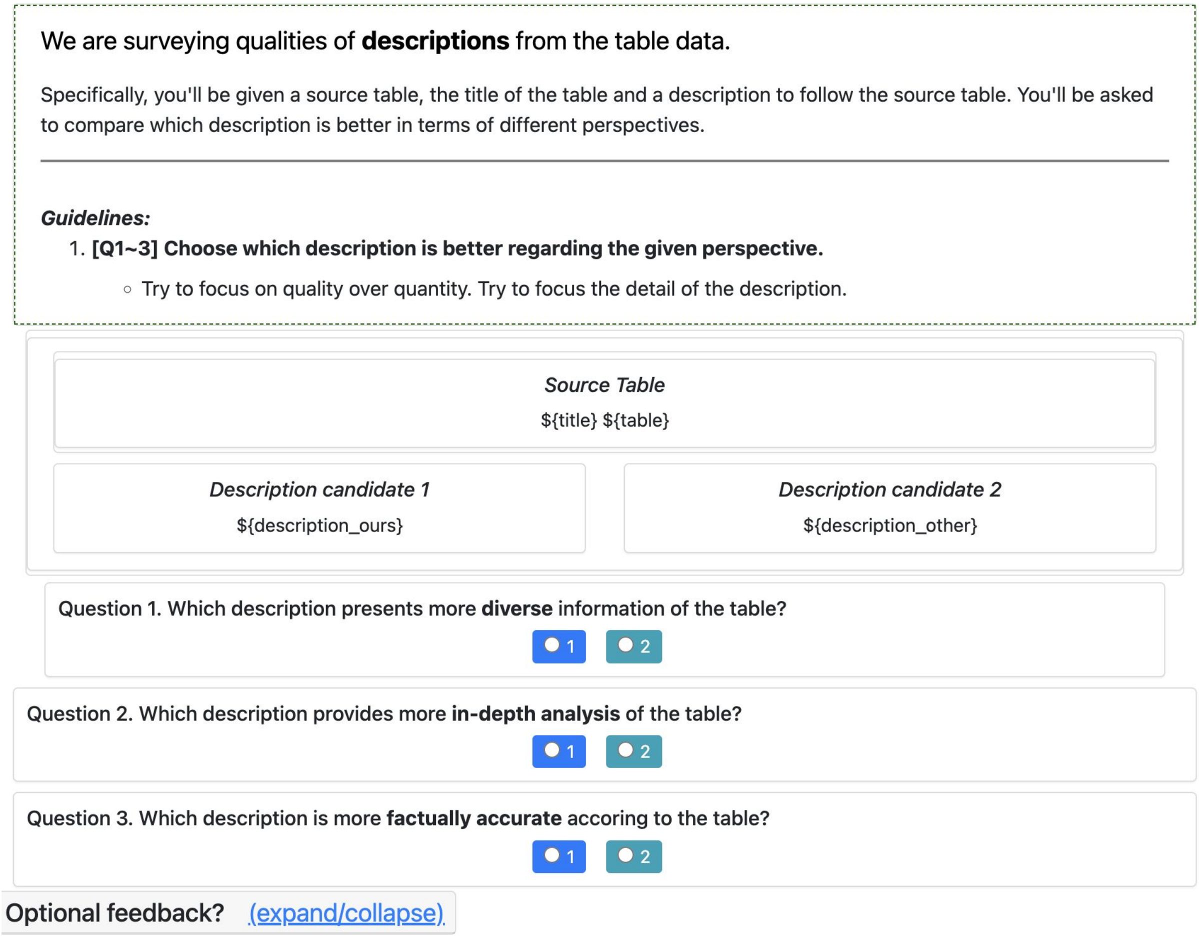


Figure 5: Annotator interface of human evaluation on reasoner generated knowledge quality