TKGT: Redefinition and A New Way of Text-to-Table Tasks Based on Real World Demands and Knowledge Graphs Augmented LLMs

Anonymous ACL submission

Abstract

The task of text-to-table receives widespread attention, but its importance and difficulty are underestimated. Existing works use simple datasets like those from table-to-text tasks and employ methods that ignore domain structures. As a bridge between raw text and statistical analysis, the text-to-table task faces challenges from more complex semi-structured texts that refer to certain domain topics in the real world with obvious entities and events, especially from those of social sciences. In this paper, we analyse the limitation of previous datasets with methods and redefine the text-to-table task, based on which we propose a new dataset called **CPL** (Chinese Private Lending) of case judgments from a real world legal academic project. We further propose TKGT (**T**ext-**KG**-**T**able), a two stages domain-aware pipeline, which firstly generates domain knowledge graphs (KGs) classes semi-automatically from raw text with the mixed information extraction (Mixed-IE) method, then adopts the hybrid retrieval augmented generation (Hybird-RAG) method to transform it to tables for downstream needs under the guidance of KGs classes. Experiment results show that TKGT achieves state-of-the-art (SOTA) performance on both traditional datasets and the CPL. Our code and data are available at https://anonymous.4open.science/r/TKGT-4755.

1 Introduction

Extracting structured information from unstructured or semi-structured text is significant for Natural Language Processing (NLP), as it means extracting valuable information through rule-based, statistical, or deep learning (DL) methods to compress texts and facilitate downstream application (Li et al., 2023a; Sui et al., 2024; Pan et al., 2024).

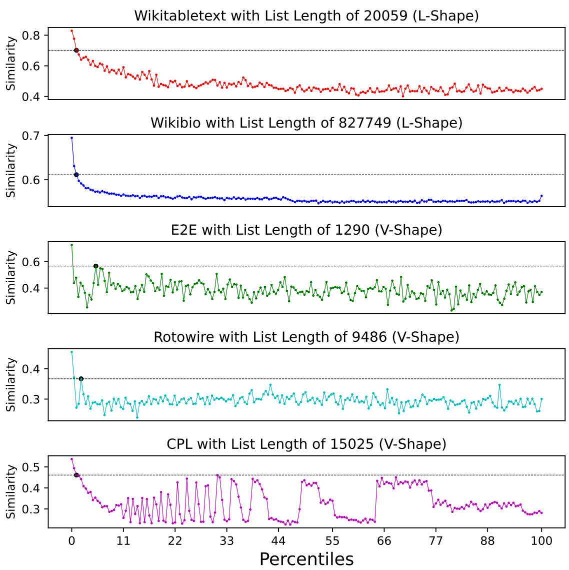

Figure 1: Statistical results of four text-to-table datasets and our **CPL**. The horizontal axis represents the percentile of the ordered word frequency lists, and the vertical axis represents the maximum similarity between each word and datasets' field sets. The intersection point is the maximum value point after 1% of each list.

Recently, with the development of deep learning (DL) especially the LLMs, some works explore the potential for Transformer models to revolutionize traditional IE (Lu et al., 2022; Wang et al., 2023; Ni et al., 2023), while some directly focus on transforming raw text to structured forms such as KGs (Kommineni et al., 2024; Meyer et al., 2023), mind maps (Jain et al., 2024), and tables (Wu et al., 2021; Li et al., 2023b; Sundar et al., 2024; Deng et al., 2024), among which table is the most popular form.

However, the importance and difficulty of text-to-table tasks are underestimated. Datasets currently used are often structurally simple, fictional, and not from *real world demands*. As shown in Table 1, the first four datasets used in current text-to-table tasks share features that the numbers of average words per document and fields are small.

[*]Equal contribution.
[†]Co-corresponding authors.

Datasets	DN	OT	TW	AW/D	TFW(%)	TF	TVTF
Wikitabletext	13318	Entity	185111	13.90	50.04%	2443	2262 / 791 / 1022
Wikibio	728221	Entity	70257683	96.48	45.22%	2996	2771 / 1400 / 1406
E2E	51426	Entity	1152364	22.41	49.04%	7	7 / 7 / 7
Rotowire	4853	Event	1637820	337.49	39.97%	33	33 / 33 / 33
CPL	850	Event	1149207	1105.94	65.58%	97	97 / 97

Table 1: Profiles of five datasets, first four ones in which are originally from table-to-text tasks (Wiseman et al., 2017; Novikova et al., 2017; Bao et al., 2018; Lebret et al., 2016) respectively and pre-processed by (Wu et al., 2021). Abbreviations are used for title, in which DN means document numbers, OT means object type, TW means total words, AW/D means average words per document, TFW means proportions after filtering, TF means total fields and are divided into three parts of train, validation, test respectively in TVTF. CPL has no validation set.

In addition, the two datasets from Wikipedia are essentially relationship extraction (RE) due to the lack of determined and refined fields. Recent work (Deng et al., 2024) proposes a new dataset that generates summary tables of sports competitions from commentary text. However, such a task is still distant from real-world applications.

In contrast, tabular data are important foundations for quantitative statistical analysis, holding tremendous value in various fields, including business intelligence (Vidal-García et al., 2019), natural sciences (Hey et al., 2009), and social sciences(King, 2014). For social scientists adopting the computational social science (CSS) paradigm (Lazer et al., 2009), there is an increasingly urgent need to efficiently extract meaningful information from unstructured or semi-structured texts and store it as tabular data (Gentzkow et al., 2019). This demand is expanding from CSS fields, such as economics (Ash and Hansen, 2023), political science (Grossman and Pedahzur, 2020), and law (Ashley, 2017), to digital humanities disciplines, including history and literature (Michel et al., 2011). Therefore, we redefine the requirements of text-to-table tasks and propose a new dataset called CPL (in Section 2) to fill the gap between existing datasets and real-world demands.

Besides, corresponding methods on previous data remain problems. Text-to-table is initially modeled as Seq2Seq tasks (Wu et al., 2021; Li et al., 2023b), embedding tokens to data-driven learn inner similarities and generate table rows end-to-end; Further researches include inferring table fields (Sundar et al., 2024) before traversing texts with RE and merging finally (Deng et al., 2024). Some works also utilize structures of text and hope to reduce difficulty through segmentation (Jain et al., 2024). After the emergence of LLMs, question and answer (Q&A) is explored as an approach for IE (Wang et al., 2023; Ni et al., 2023). However, existing works ignore the importance and difficulty of building table fields and treat them as known or just extract triples by simply crawling, which are only applicable to simple formats since that identifying valuable information in complex texts and building fields themselves require professional efforts. Besides, it's challenging to guarantee completeness, especially for long texts whose valuable points may scatter globally or in the disguise of multiple perspectives that are common in real world.

We propose TKGT (**T**ext-**KG**-**T**able), a two-stage text-to-table method with KGs as middleware. **In the first stage**, the Mixed-IE method based on regulations, statistics, and DL is used to obtain topic keywords and to construct domain KGs sketch, based on which users can better understand the datasets and easily form uninstantiated KGs adapting to downstream tabular needs using LLMs. **In the second stage**, based on dynamic prompts and Hybrid-RAG supported by descriptions of empty KGs classes, table content can be filled with LLMs Q&A. Through experiments, TKGT achieves SOTA performance on both traditional datasets and CPL. Our contributions are summarized as follows:

- Redefine the characteristics and requirements of text-to-table tasks in a more standardized manner and introduce the CPL, a new and highly challenging manually completed dataset in the field of law.

- Propose the two-stage TKGT, filling the gap in how to obtain table fields based on domain topic structures and use the Hybird-RAG to fill the table with Q&A. We also demonstrate its SOTA performance through experiments.

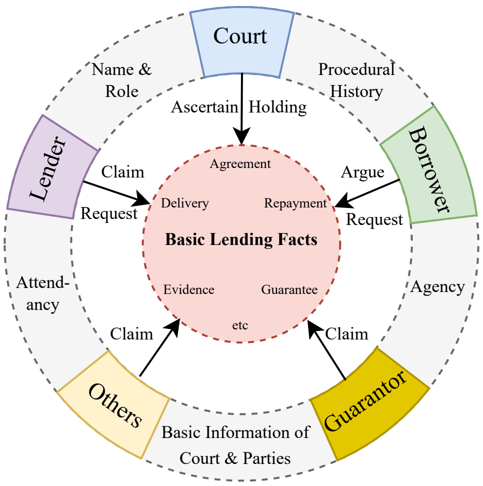

Figure 2: Overview of CPL dataset, which include five role types who have own view for the case facts in the outer layer, and a key contents set of the case in the inner layer.

2 Redefinition and CPL Dataset

The CPL dataset is from a real-world academic project, whose raw texts are from the China Judgments Online (CJO)[1], collected manually by legal experts (Appendix A).

2.1 Redefinition of Text-to-Table

For the convenience of further research, we redefine the requirements of text-to-table as follows. **Firstly**, table fields must be limited and refined to serve a practical need rather than unrestricted key-value pairs. **Secondly**, topic information must be clear, and content can be modeled as multi-attribute entities or multi-entity events. **Thirdly**, the information is relatively complex, requiring certain writing formats with logic for clear organization.

As shown in Figure 1, the maximum similarity curves of Rotowire, E2E and CPL present a **V-shape** pattern that first decreases, then rebounds and oscillates after one percent position at lists, which indicates that there exists not only the field information at the front of lists, but also the shared structural information dissimilar with fields on the semantic meaning. In contrast, curves of the two datasets from Wikipedia consistently decrease as **L-shape**, indicating no obvious structural information and explaining why the field numbers of the two datasets are so large and inconsistent in Table 1.

[1]CJO, established by the Supreme People's Court of the People's Republic of China(SPC), allows the public to freely search, read, download, and analyze cases.https://wenshu.court.gov.cn/

2.2 Statistics of CPL

The CPL dataset contains 850 judgment documents and corresponding tables. **Firstly**, it is a typical event-type dataset, which including one lender, one court, at least one borrower, zero or several guarantors, and other roles like witnesses (Figure 2). As shown in Table 1, it has 1149209 words in total and 1105.94 in each document on average. Fields in this dataset are scalable to fit multiple lending in a case. Actual field numbers depends on specific text contents and exceeds 220 overall, which are ultimately abstracted into 97 core fields considering reusable concepts such as interest and penalty sharing attributes like start date and interest rate. **Secondly**, to reduce the complexity of subsequent works, we filter out stop words and stop position tag, leaving behind 753610 core words, accounting for 65.58% of the total, which is much higher than the other four datasets filtered based on the same strategy (Appendix C). **Thirdly**, as shown in Figure 1, this dataset shows a significant **V-shape**, which is similar to the other two datasets with table structure (Rotowire and E2E). In short, this dataset has longer text, more complex field structures, higher word quality, and distinct semi-structured features.

3 TKGT Two-Stages Pipeline

3.1 Overview

As illustrated in Figure 3, TKGT uses KGs classes as middleware to transform raw texts to tables through two stages. The first stage aims at semi-automatically assisting users to better understand datasets with the Mixed-IE methods, based on which LLMs can be used to mine the topic information and construct domain models in the form of KGs classes without instantiating. The second stage adopts the Hybrid-RAG method to extract values under the guidance of KGs classes and interpret them into tables with specialized fields according to downstream needs using dynamic prompts.

3.2 Mixed-IE Assisted KGs Generation

As illustrated in Figure 3 (a), 1 represents regulations and seed knowledge from human and 2 represents the relevant inner knowledge of LLMs from pre-training, based on which 3 and 4 preprocesses the dataset such as section segmentation, tokenization, position tagging, named entity recognition (NER), and feature distribution statistics as well as filtering, to obtain lists of high term frequency (TF) and document frequency (DF). 5 con-

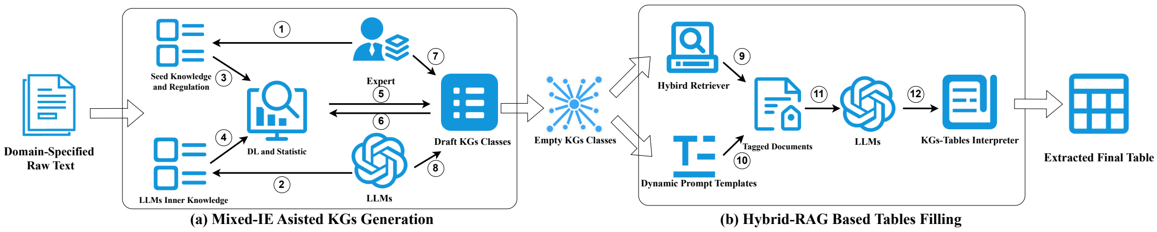

Figure 3: Overview of two-stages pipeline of TKGT.

structs domain models in the form of KGs classes with the joint efforts of both human expert ⑦ and LLMs ⑧, who also check the quality of KGs and iterate it ⑥ to get final KGs classes. Here follows the details of regulations, statistics, and DL, especially LLMs methods, separately.

3.2.1 Regulations

Regulations refer to the structure, format, and logic, which help to decompose complex texts into multiple independent parts, reducing overall complexity. **Firstly**, for general writing sense, writers produce texts logically, such as the *What-Why-How Principle*, which is the inner structure meaning different parts undertake different functions with different information. **Secondly**, complex texts usually adopt explicit structures like hierarchical sections of academic papers to show inner logic clearly to readers. **Finally**, shared elements are usually fixed in the same positions, such as titles, author names, and dates in certain lines. For instance, CPL judgment documents contain the logic of legal trial and usually adopt ordered positional words to present them more clearly, as shown in Appendix B. By decomposing based on regulations, the difficulties of subsequent work can be greatly reduced. Thus, if users want to retrieve identity information, the best choice is to perform small-scale retrieval in the corresponding section.

3.2.2 Statistics

Purposes of statistics are ensuring the completeness of IE to minimize losses of key words and exploring topic and structure information. With mature NLP toolkits and specified filtering, TF and DF reflect both target information of a domain dataset. As shown in Figure 1, after calculating the semantic similarity of words and table fields, documents with the potential for tabulation (Rotowire, E2E, and CPL) will exhibit a **V-shape** pattern. By manually checking frequency lists, it can be found that the first one percent of the front parts of lists contain almost all keywords, while the bottom part of **V-shape** contains structure words dissimilar with fields. Through statistics, users can quickly extract keywords from large text sets and serve for LLMs and human experts, greatly reducing the difficulty of constructing KGs classes with completeness.

3.2.3 LLMs and KGs

An important trend of text-to-table is to break down the original end-to-end paradigm into multiple stages like (Deng et al., 2024) using triplets as middleware. Compared to the topic-ignoring crawling paradigm of triples, KGs can better model entities and events, logically organize different roles and adapt to downstream tabular needs. TKGT statistics overall datasets to obtain relevant KGs classes, which logically conducts retrieving values of certain objects' fields in the second stage. This not only conforms to more interpretable human methodology but is also more accurate and complete. However, considering that KGs generation itself is a difficult task and existing research results only demonstrate the possibility of using LLM to assist human experts in generation (Meyer et al., 2023; Kommineni et al., 2024), we simplify it as a *slack classes mining task* with aims of reducing human expert participation. That is, we do not instantiate KGs and only abstract them as a set of classes with two types of *role entity* and *relation/action* as shown in Appendix C.

3.3 Hybird-RAG Based Table Filling

As illustrated in Figure 3 (b), ⑨ and ⑩ use KGs classes from the first stage to dynamically rewrite prompt templates and guide the hybrid retriever respectively, combining with documents tagged in the first stage to avoid unnecessary queries as LLMs inputs ⑪. With inputs containing a set of retrieved original texts as evidence and prompts, LLMs can get certain values of the KGs classes ⑫ and transform them to table form through the KGs-table interpreter.

3.3.1 Structure-Aware Hybrid-RAG

We create an algorithm for scheduling the RAG process with KGs, which is easy to understand and adapt to other variants.

Algorithm 1 KG Object Label Filling Algorithm

```
1: Initialize an empty KG object
2: while the KG object contains empty labels do
3:    if no entity in KG has filled labels then
4:       Select the entity with highest centrality
5:    else
6:       Calculate the ratio Count(Label|Unfilled) / Count(Label) for each entity
7:       Select the entity with the highest ratio of unfilled labels
8:    end if
9:    if the selected entity's name label is not filled then
10:      Search and extract the entity name
11:   else
12:      Randomly select one unfilled label
13:      Search and extract information for the unfilled label
14:   end if
15:   if the information is found then
16:      Fill the searched information to the label
17:   else
18:      Fill 'Bad Information' to the label
19:   end if
20: end while
```

3.3.2 Rewriting Prompt Dynamically

We also utilize our KG design for query rewriting and summarizing relevant information before passing them into the IE prompt. For query rewriting, we describe the relations between the "to-be-extracted entity" and the label values of its adjacent entities in the prompt. An example prompt is provided, asking the query rewriting model to generate a search query for retrieving relevant information. For information summarization, we describe the same relations between the "to-be-extracted entity" and the label values of its adjacent entities in the prompt, asking the summarization model to retain information that might be useful for answering the user's question as shown in Appendix D.

4 Experiments

This section introduces the experimental setup and results of TKGT's two stages respectively.

4.1 Setup

Datasets. As shown in Table 1, experiments use datasets of Rotowire and E2E with table structure processed by (Wu et al., 2021) and the CPL dataset whose details are at Section 2 for more complex challenges.

Baselines and Models. Considering the extensive exploration of instruction following for various LLMs (Ni et al., 2023; Deng et al., 2024), we pick several popular LLMs as processors and focus on the performance of TKGT on different datasets. Table 3 shows baselines and models used. *(1) For first stage,* we choose LLaMA3-70B[2] to test the ability of KGs classes generation, comparing it with two naive solutions: pure LLM with naive prompt, and LLM with the same prompt template of TKGT's using In-Context-Learning (ICL) and Chain-of-Thought (CoT) but without statistical results. *(2) For the second stage,* for the demands of deploying LLMs on consumer-grade GPUs in many social science scenarios, we choose ChatGLM3-6B[3] to test the ability of table extraction. We also fine-tune it with LoRA (Hu et al., 2021) and compare it with mainstream and SOTA commercial LLM of GPT series[4].

Metrics. *(1) For the first stage*, we develop an evaluation method for the quality of KGs generation aiming at using LLMs to assist humans in constructing domain KGs. We also recruit a group of graduate students with knowledge in law and computer science as referees. For the target dataset, a set of fields is predefined by humans, and weights are assigned to each field on average or based on importance, which sum to 1. By checking the generated fields one by one with the target fields, we can accumulate scores according to the rules in Table 2, whose core principle is whether humans can be inspired naturally by the fields generated by LLMs. *(2) For the second stage*, metric follows the F1 score at three levels defined in (Wu et al., 2021)

4.2 Results of TKGT's First Stage

Since TKGT's first stage is semi-automatic, results can be iteratively improved by feedback from human and LLMs, making it difficult to reproduce. Therefore, we only present results of first iteration, in which TKGT provides predefined few-shot templates and Mix-IE results, guiding LLMs to

[2] https://github.com/meta-llama/llama3
[3] https://github.com/THUDM/ChatGLM3
[4] https://openai.com/index/gpt-4-research/

Matching Degree	Relationship of G&T Fields	Scoring Rules
Totally Match	Match in form or semantics	Obtain the total score of target field only once.
Including	Be a neighboring parent concept	Obtain 75% of the sum of all target fields.
Included	Be a neighboring sub concept	If parent concept is separable, obtain the field score divided by the number of categories each; If not, gain 25%.
Not Match	Completely different	No score.

Table 2: Metrics for the quality of KGs generated by TKGT's first stage, in which *Relationship of G&T Fields* means the best-matching pair of one generated filed and one target field. *Neighboring* refers to the ability to naturally infer parent/child concepts from subsequent textual information.

Stage	Method	Detail
First Stage	Zero-shot	LLaMA3-70B
	Few-shot	LLaMA3-70B & Prompt Template
	TKGT-Stage-1	LLaMA3-70B & Prompt Template & Statistics
Second Stage	Commercial LLM	GPT-3.5-turbo
	SOTA Commercial LLM	GPT-4-turbo
	Open-Source LLM	ChatGLM3-6B
	TKGT-Stage-2	ChatGLM3-6B & LoRA Tuning & RAG & KGs

Table 3: Experiment baselines of TKGT and details. LLaMA3-70B is one of the largest and most powerful open-source LLMs. ChatGLM3-6B is a popular medium-sized open-source LLM. GPT series contain the most popular commercial LLMs.

Subset	Model	The first column F1			Table header Fl			Data cell F1			Error
		Exact	Chrf	BERT	Exact	Chrf	BERT	Exact	Chrf	BERT	
Team	Sent-level RE	85.28	87.12	93.65	85.54	87.99	87.53	77.17	79.10	87.48	0.00
	Doc-level RE	84.90	86.73	93.44	85.46	88.09	87.99	75.66	77.89	87.82	0.00
	Seq2Seq	94.71	94.93	97.35	86.07	89.18	88.90	82.97	84.43	90.62	0.49
	Seq2Seq-c	94.97	95.20	97.51	86.02	89.24	89.05	83.36	84.76	90.80	0.00
	Seq2Seq&set	**96.80**	**97.10**	**98.45**	86.00	89.48	93.11	84.33	85.68	**91.30**	0.00
	T-(No RAG)-T*	72.38	72.84	73.41	**100.0**	**100.0**	**100.0**	64.42	65.53	66.84	0.00
	T-KG-T*	91.44	91.83	93.26	**100.0**	**100.0**	**100.0**	**85.03**	**87.58**	91.21	0.00
Player	Sent-level RE	89.05	93.00	90.98	86.36	89.38	93.07	79.59	83.42	85.35	0.00
	Doc-level RE	89.26	93.28	91.19	87.35	90.22	97.30	80.76	84.64	86.50	0.00
	Seq2Seq	92.16	93.89	93.60	87.82	91.28	94.44	81.96	84.19	88.66	7.40
	Seq2Seq-c	92.31	94.00	93.71	87.78	91.26	94.41	82.53	84.74	88.97	0.00
	Seq2Seq&set	92.83	94.48	**96.43**	88.02	91.60	95.08	83.51	85.75	**90.93**	0.00
	T-(No RAG)-T*	67.51	69.29	69.22	**100.0**	**100.0**	**100.0**	64.27	66.25	66.94	0.00
	T-KG-T*	**93.05**	**94.59**	95.18	**100.0**	**100.0**	**100.0**	**88.26**	**90.18**	90.39	0.00

Table 4: Results of baselines, pure LLMs prompts, and our TKGT model on Rotowire. We show the F1 score based on exact match (Exact), chrf score (Chrf), and BERTScore (BERT) respectively. GLM3-6B refers to the pre-trained ChatGLM3-6B model without any finetuning. * refers to the finetuned IE model tuned on the respective IE finetuning dataset we created based on the corresponding dataset.

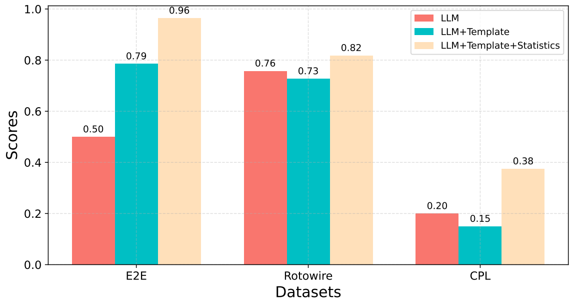

Figure 4: Results of TKGT's first stage.

generate KGs classes for three datasets. Besides, we add ablation elements to it, removing Mix-IE results and few-shot templates. We run 10 times each and submit outputs to a group of human judge with metrics to obtain the best result.

As shown in Figure 4, TKGT achieves the best performance on all datasets, which proves that our method can extract more complete domain models. We observe that scores decrease as the complexity (numbers of fields and structures) of the dataset increases, and TKGT get 0.96 and 0.82 on E2E and Rotowire respectively, indicating that TKGT can generate almost complete structures for traditional datasets. Furthermore, as for Rotowire and CPL, the method with Few-shot templates but without results from Mix-IE gets even lower scores than pure LLM, which means templates without top keywords hinder LLM's ability to exert its inner knowledge and proves the importance of Mixed-IE. Finally, TKGT performs poorly without iteration, proposing further research challenges.

4.3 Results of TKGT's Second Stage

As shown in Table 4 and the first half of Table 5, our TKGT pipeline achieves near SOTA performance with minimal dataset-specific engineering for the Rotowire dataset. Our KG-based design avoids generating incorrect table headers and mismatched table shapes, achieving perfect scores in table header F1 and Error compared to previous methods. The relatively low F1 scores for the first column (Team name) extraction are due to the model's difficulty in identifying 'home team' and 'visiting team' from their positions in the text. We achieve SOTA performance on nearly all metrics. We did not use any RAG technique in the ablation experiment because both the E2E and Rotowire data are short and lack a specific writing style, where RAG might cause more information loss than precision gain. Comparing 'T-(No RAG)-T' and 'T-KG-T' shows the benefits of our KG-guided query, query-rewrite, and summarizing pipeline.

We compare TKGT with larger commercial LLMs on CPL dataset. Despite the base model's limitations, T-KG-T performs comparably to more advanced models like GPT-4-Turbo using naive RAG, showcasing the effectiveness of our KG-guided methods. Fine-tuning the IE model is crucial for 'Text-to-Table' tasks, initially ensuring adherence to the output format, then distinguishing between valid and invalid information cases, and finally accurately extracting valid information. Our KG-guided query, query-rewrite, and summarizing pipeline enhance the model's ability to deliver accurate information by reducing unnecessary context and adding relevant information, ultimately achieving state-of-the-art performance.

5 Related Work

5.1 Text-to-Table Works in Social Science

Text-to-table works in social science are more engineering-oriented, meeting needs of text-as-data (Ash and Hansen, 2023), which involves four core empirical tasks: 1 measure document similarity (Cagé et al., 2020; Kelly et al., 2021); 2 concept detection (Shapiro et al., 2022; Angelico et al., 2022); 3 how concepts are related (Thorsrud, 2020; Ash et al., 2024); 4 associate text to metadata (Ke et al., 2019)). Traditional methods of structuring is manual coding, such as Chang et al. (2021) spending years coding 170 dimensions of property law in 128 jurisdictions to draw the legal family. With the development of NLP, structuring tasks become semi-automated or even fully-automated (Grimmer et al., 2022). Luo et al. (2017) propose an Transformer-based method to simultaneously model charge prediction and relevant article extraction tasks. Mentzingen et al. (2024) first develop a two-stage cascade classifier model that predicts regulatory decisions, based on textual features extracted from the original documents by ML and proceedings' metadata.

5.2 Text-to-Table Works in Computer Science

The research paradigm of text-to-table officially originated from Wu et al. (2021), which uses datasets from table-to-text and an end-to-end sequence generation mode based on the BART model. All rows are generated at once, and the results are controlled using table constraints and column embedding. Li et al. (2023b) improves it by point-

Dataset	Model	The first column F1			Data cell F1			Error
		Exact	Chrf	BERT	Exact	Chrf	BERT	
E2E	NER	85.28	87.12	93.65	85.54	87.99	87.53	0.00
	Seq2Seq	84.90	86.73	93.44	85.46	88.09	87.99	0.49
	Seq2Seq-c	94.71	94.93	97.35	86.07	89.18	88.90	0.00
	Seq2Seq&set	94.97	95.20	**97.51**	86.02	89.24	89.05	0.00
	T-(No RAG)-T (GLM3-6B*)	74.34	76.07	78.92	71.39	73.15	74.07	0.00
	T-KG-T (GLM3-6B*)	**95.14**	**95.87**	96.12	**92.17**	**93.79**	**92.83**	0.00
CPL	T-(Naive RAG)-T (GPT3.5)	84.26	82.67	66.28	79.43	67.73	55.01	0.00
	T-(Naive RAG)-T (GPT4)	**93.41**	**91.73**	80.52	90.27	**88.62**	78.70	0.00
	T-(No RAG)-T (GLM3-6B)	9.97	0.89	0.95	4.60	1.95	0.86	0.00
	T-(Naive RAG)-T (GLM3-6B)	12.25	11.31	11.98	8.87	2.19	1.98	0.00
	T-KG-T (GLM3-6B*)	91.33	88.79	**82.68**	**90.79**	87.58	**82.45**	0.00

Table 5: Results of baselines, pure LLMs prompts, and our TKGT model on CPL. F1 scores are same as Table 4. GLM3-6B refers to the pretrained ChatGLM3-6B model without any finetuning. GLM3-6B* refers to the finetuned IE model tuned on the respective IE finetuning dataset we created based on the corresponding dataset.

ing out the order-insensitive property of rows and adopted a fast method of generating all rows in parallel after generating the header. Sundar et al. (2024) abandons the end-to-end paradigm and adopts a two-stage approach of generating table frameworks and content separately and switches to use conditional Q&A for IE. Deng et al. (2024) further innovates by proposing a new benchmark and uses LLMs prompt engineering to extract triples from the original text and merge them into tables.

5.3 LLMs Prompt and Knowledge Graphs

Prompt originated from the GPT-3 series (Brown et al., 2020), whose works focus on engineering experience and practice, such as the various prompt techniques listed in (Liu et al., 2023). In addition, Sahoo et al. (2024) combines prompt and fine-tuning to explain the essence of instruction following. Wang et al. (2023) further explores the potential of fine-tuned LLMs in IE. As for KGs, recent works explore how to use LLMs to empower the construction of KGs. Meyer et al. (2023) first explores the potential of LLMs to generate KGs in multiple engineering fields, Ni et al. (2023) elucidates the complementary relationship between LLMs and KGs, and Kommineni et al. (2024) proposes a semi-automatic pipeline method using LLMs to assist human experts in generating KGs as the latest research.

5.4 IE and RAG

Retrieval-Augmented Generation (RAG) aims to enhance the factual accuracy of Large Language Models (LLMs) by incorporating relevant textual information, thereby expanding the knowledge base of the training data and reducing hallucination problems (Gao et al., 2024). Khattab et al. (2023) was one of the pioneering works utilizing the in-context learning ability of LLMs to perform knowledge-intensive information retrieval tasks in the form of question-answering. Subsequent research has made various improvements to RAG, such as introducing new data structures for retrieval data (Luo et al., 2023; He et al., 2024) and developing more efficient retrieval pipelines. These advancements include hybrid retrieval methods (Gao et al., 2022), fine-tuning embeddings(Shi et al., 2023), reranking (Yu et al., 2023), and iterative retrieval processes (Cheng et al., 2023).

6 Conclusion

We first review the research field of text-to-table, point out the shortcomings of existing datasets with statistical methods, and redefine the core requirements of this task more comprehensively. Secondly, we propose a social science dataset CPL from real-world structuring requirements, which presents new challenges to the field due to its complexity and semi-structured nature. In addition, to address the shortcomings of existing text-to-table methods that overlook topic and structural information, we propose a two-stage pipeline called TKGT using KGs classes as middleware and demonstrate its SOTA performance through experiments.

Limitations

Although the TKGT pipeline we propose covers the entire process of text-to-table task, it cannot be fully automated in the first stage. On the one hand, this is limited by the current capabilities of LLMs; On the other hand, academic level complex text extraction tasks are extremely challenging even for untrained humans. One possible solution is to build the first stage as a more comprehensive and powerful agent, and explore a more powerful initialization framework that balances universality and practicality. This is also one of our future tasks.

Ethics Statement

This work does not adopt AI assistants. The four datasets we use are entirely from the MIT license open-source pre-processing results of previous work (Wu et al., 2021), while the CPL dataset is sourced from the official judgment documents publicly available on the CJO, which complies with the requirement of transparency in court rulings. The CPL dataset involves real person names and other information. In order to further ensure privacy and ensure the accuracy of named entity recognition during data pre-processing, we randomly replaced the person names using existing named entity recognition techniques (He and Choi, 2021). In addition, all experiments in this work followed the expected purpose of their research. Therefore, to the best of the author's knowledge, we believe that this work will not bring any additional risks.

References

Cristina Angelico, Juri Marcucci, Marcello Miccoli, and Filippo Quarta. 2022. Can we measure inflation expectations using twitter? *Journal of Econometrics*, 228(2):259–277.

Elliott Ash, Germain Gauthier, and Philine Widmer. 2024. Relatio: Text semantics capture political and economic narratives. *Political Analysis*, 32(1):115–132.

Elliott Ash and Stephen Hansen. 2023. Text algorithms in economics. *Annual Review of Economics*, 15(Volume 15, 2023):659–688.

Kevin D. Ashley. 2017. *Artificial Intelligence and Legal Analytics: New Tools for Law Practice in the Digital Age*. Cambridge University Press, Cambridge.

Junwei Bao, Duyu Tang, Nan Duan, Zhao Yan, Yuanhua Lv, Ming Zhou, and Tiejun Zhao. 2018. Table-to-text: Describing table region with natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Julia Cagé, Nicolas Hervé, and Marie-Luce Viaud. 2020. The production of information in an online world. *The Review of Economic Studies*, 87(5):2126–2164.

Yun-chien Chang, Nuno Garoupa, and Martin T Wells. 2021. Drawing the legal family tree: An empirical comparative study of 170 dimensions of property law in 129 jurisdictions. *Journal of Legal Analysis*, 13(1):231–282.

Xin Cheng, Di Luo, Xiuying Chen, Lemao Liu, Dongyan Zhao, and Rui Yan. 2023. Lift yourself up: Retrieval-augmented text generation with self memory. *Preprint*, arXiv:2305.02437.

Zheye Deng, Chunkit Chan, Weiqi Wang, Yuxi Sun, Wei Fan, Tianshi Zheng, Yauwai Yim, and Yangqiu Song. 2024. Text-tuple-table: Towards information integration in text-to-table generation via global tuple extraction. *arXiv preprint arXiv:2404.14215*.

Federal Judicial Center FJC. 2020. *Judicial Writing Manual: A Pocket Guide for Judges*. Lulu.com.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Precise zero-shot dense retrieval without relevance labels. *Preprint*, arXiv:2212.10496.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Matthew Gentzkow, Bryan Kelly, and Matt Taddy. 2019. Text as data. *Journal of Economic Literature*, 57(3):535–574.

Justin Grimmer, Margaret E. Roberts, and Brandon M. Stewart. 2022. *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton University Press, Princeton.

Jonathan Grossman and Ami Pedahzur. 2020. Political science and big data: Structured data, unstructured data, and how to use them. *Political Science Quarterly*, 135(2):225–257.

Han He and Jinho D Choi. 2021. The stem cell hypothesis: Dilemma behind multi-task learning with transformer encoders. *arXiv preprint arXiv:2109.06939*.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Preprint*, arXiv:2402.07630.

Tony Hey, Stewart Tansley, and Kristin Tolle. 2009. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, Redmond, Washington.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Parag Jain, Andreea Marzoca, and Francesco Piccinno. 2024. Structsum generation for faster text comprehension. *arXiv preprint arXiv:2401.06837*.

Zheng Tracy Ke, Bryan T. Kelly, and Dacheng Xiu. 2019. Predicting returns with text data. (26186). DOI: 10.3386/w26186.

Bryan Kelly, Dimitris Papanikolaou, Amit Seru, and Matt Taddy. 2021. Measuring technological innovation over the long run. *American Economic Review: Insights*, 3(3):303–320.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2023. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp. *Preprint*, arXiv:2212.14024.

Gary King. 2014. Restructuring the social sciences: Reflections from harvard's institute for quantitative social science. *PS, Political Science Politics*, 47(1):165–172.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization. *Preprint*, arXiv:1412.6980.

Vamsi Krishna Kommineni, Birgitta König-Ries, and Sheeba Samuel. 2024. From human experts to machines: An llm supported approach to ontology and knowledge graph construction. *arXiv preprint arXiv:2403.08345*.

David Lazer, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy, and Marshall Van Alstyne. 2009. Computational social science. *Science*, 323(5915):721–723.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771*.

Peng Li, Yeye He, Dror Yashar, Weiwei Cui, Song Ge, Haidong Zhang, Danielle Rifinski Fainman, Dongmei Zhang, and Surajit Chaudhuri. 2023a. Table-gpt: Table-tuned gpt for diverse table tasks. *arXiv preprint arXiv:2310.09263*.

Tong Li, Zhihao Wang, Liangying Shao, Xuling Zheng, Xiaoli Wang, and Jinsong Su. 2023b. A sequence-to-sequence&set model for text-to-table generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5358–5370.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. *arXiv preprint arXiv:2203.12277*.

Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. 2017. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, page 2727–2736, Copenhagen, Denmark. Association for Computational Linguistics.

Ziyang Luo, Can Xu, Pu Zhao, Xiubo Geng, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023. Augmented large language models with parametric knowledge guiding. *Preprint*, arXiv:2305.04757.

Hugo Mentzingen, Nuno Antonio, and Victor Lobo. 2024. Joining metadata and textual features to advise administrative courts decisions: a cascading classifier approach. *Artificial Intelligence and Law*, 32(1):201–230.

LP Meyer, C Stadler, J Frey, N Radtke, K Junghanns, R Meissner, G Dziwis, K Bulert, and M Martin. 2023. Llm-assisted knowledge graph engineering: experiments with chatgpt (2023). In *conference proceedings of AI-Tomorrow-23*, volume 29, pages 6–2023.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, THE GOOGLE BOOKS TEAM, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

Xuanfan Ni, Piji Li, and Huayang Li. 2023. Unified text structuralization with instruction-tuned language models. *arXiv preprint arXiv:2303.14956*.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The e2e dataset: New challenges for end-to-end generation. *arXiv preprint arXiv:1706.09254*.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*.

Adam Hale Shapiro, Moritz Sudhof, and Daniel J. Wilson. 2022. Measuring news sentiment. *Journal of Econometrics*, 228(2):221–243.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *Preprint*, arXiv:2301.12652.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 645–654.

Anirudh Sundar, Christopher Richardson, and Larry Heck. 2024. gtbls: Generating tables from text by conditional question answering. *arXiv preprint arXiv:2403.14457*.

Leif Anders Thorsrud. 2020. Words are the new numbers: A newsy coincident index of the business cycle. *Journal of Business Economic Statistics*, 38(2):393–409.

Javier Vidal-García, Marta Vidal, and Rafael Hernández Barros. 2019. *Computational Business Intelligence, Big Data, and Their Role in Business Decisions in the Age of the Internet of Things*, page 1048–1067. IGI Global.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. 2023. Instructuie: multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*.

Sam Wiseman, Stuart M Shieber, and Alexander M Rush. 2017. Challenges in data-to-document generation. *arXiv preprint arXiv:1707.08052*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing. *Preprint*, arXiv:1910.03771.

Xueqing Wu, Jiacheng Zhang, and Hang Li. 2021. Text-to-table: A new way of information extraction. *arXiv preprint arXiv:2109.02707*.

Zichun Yu, Chenyan Xiong, Shi Yu, and Zhiyuan Liu. 2023. Augmentation-adapted retriever improves generalization of language models as generic plug-in. *Preprint*, arXiv:2305.17331.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *Preprint*, arXiv:2403.13372.

A Details of CPL Dataset

In order to study private lending in China, such as the changing patterns of lending behavior, the logic and efficiency of trail, and the policy effects of interest rate regulation, a real-world academic project obtains CPL judgements from the CJO and conducts manual structuring of these judgements. The main goal of this work is to extract the content of each judgment as comprehensively as possible into a structured format in a table.

The project carries out this work through the following steps. **Firstly**, design the format of the table. In different countries, the logic of trials and the writing of judgements are basically the same (FJC, 2020). The core logic of the court's trial is to accurately grasp the claims and grounds of the litigants surrounding the same lending behavior facts, and the court makes its determination and judgment accordingly. And the CPL judgments have a consistent structure. Therefore, the project reassemble the content of the judgement into a *(2×n)×5* format, as shown in Figure 2. The *2* represents the two major dimensions: Basic Information of Court and Parties and Basic Lending Facts. The *n* represents the specific content under each dimension. The *5* represents the five main entities: court, borrower, lender, guarantor, and others. **Secondly**, set over 200 fields and corresponding value ranges by reading judgements and sorting out relevant legal norms. These fields basically cover the core elements of trial, such as the *Elemental Trial Guide*[5] and the *Model Texts of Written Civil Complaints and Statements of Defense*[6] , indicating that this work is thorough and scientific. The Excel table for manual data collection is constructed by professors and graduate students in law. **Thirdly**, complete text-to-table manually. The project recruit undergraduate students with a legal background and conduct a two-week training. The work is carried out in a one-by-one format, with one undergraduate student collecting and one graduate student student reviewing.

This project recruited students and compensated them based on the work-study standards of their respective universities. It provided participants with the full text of instructions, including disclaimers

[5]Issued by The High People's Court of Shandong Province, http://ytzy.sdcourt.gov.cn/ytzy/yhfzyshj/zxht39/sfwj/6518994/index.html

[6]Issued by the Supreme People's Court, the Ministry of Justice, and the All China Lawyers Association, https://pkulaw.com/chl/1b4f90e3dcf35b36bdfb.html

of any risks. The data collection protocol was approved by an ethics review board. The subjects included in CPL dataset are Chinese citizens, primarily from Shanghai, Zhejiang Province, and Anhui Province. We obtained authorization from the project leader to use the CPL dataset.

B Structure of CPL Judgement

中国民间借贷裁判文书

安徽省合肥市包河区人民法院
民 事 判 决 书

(2016)皖0111民初xxxx号

原告：卢xx，男，xxxx年x月x日出生，汉族，住xx。
委托代理人：张xx，xx律师事务所律师。

被告：刘xx，男，xxxx年x月x日出生，汉族，住xx。
委托代理人：沈xx，xx律师事务所律师。

原告卢xx诉被告刘xx民间借贷纠纷一案，本院于2016年3月29日立案受理。依法由审判员方xx适用简易程序公开开庭进行了审理。原告卢xx的委托代理人张xx，被告刘xx的委托代理人沈xx到庭参加诉讼。本案现已审理终结。

原告卢xx诉称:被告自2013年10月起，以经营周转为由向原告借款。……请求法院判令被告向原告偿还借款707100元、利息22921.66元（从2015年7月1日起按同期银行6个月贷款利率计算至起诉之日，后续按照年息4.35%的标准计算至实际还款之日止），两项合计730021.66元；本案诉讼费用由被告承担。

被告刘xx辩称：被告虽然出具了707100元的借条，但没有实际收到借款，原告也未提交转账凭证。请求法院驳回原告的全部诉讼请求。

经审理查明：被告刘xx因缺少周转资金于2013年10月向原告卢xx提出借款……原告催讨借款无果，遂于2016年3月29日向本院提起诉讼。

上述事实，有原告及其配偶王xx的身份证复印件，……等证实。

本院认为：被告刘xx借原告卢xx人民币707100元，有借条、银行转账凭条佐证，本院予以确认。被告出具的借条约定了还款期限，未约定利息，故双方之间是定期无息借贷。……据此，依照《中华人民共和国民法通则》第九十条，《中华人民共和国合同法》第二百零六条、第二百零七条之规定，判决如下：

被告刘xx于本判决生效后十日内偿还原告卢xx借款本金707100元，并支付逾期利息（以707100元为基数，自2015年7月1日起按照同期银行六个月贷款利率计算至借款本金付清之日止）。

如果未按本判决指定的期间履行给付金钱义务，应当依照《中华人民共和国民事诉讼法》第二百五十三条之规定，加倍支付迟延履行期间的债务利息。

案件受理费11100元，减半收取5550元，由被告刘xx负担。

如不服本判决，可在判决书送达之日起十五日内，向本院递交上诉状，并按对方当事人的人数提出副本，上诉于安徽省合肥市中级人民法院。

审判员　　方xx
二〇一六年六月二十二日
书记员　　奚xx

Figure 5: CPL Judgement Demo (Chinese Version).

Due to the issuance of *Specifications for Preparing Civil Judgments by the People's Courts*[7] and the *Style of Civil Litigation Documents*[8] by SPC, CPL judgments have a consistent structure (Figure 5 and Figure 6): ① Basic information of the court, such as the name of the court, the name of the judgment, and the case number; ② Parties and their basic information (e.g., name, address, role); ③ Procedural history; ④ Claims, facts, and grounds of the parties; ⑤ Evidence and facts identified by the court; ⑥ Grounds, judicial basis, and main body of judgment; ⑦ Signatory information, such as the information of the trial personnel and the closed date.

Chinese Private Lending Judgement

The Primary People's Court of Baohe District of Hefei City, Anhui Province
Civil Judgment

(2016) Wan 0111 Min Chu No. xxxx

Plaintiff: Lu xx, male, born on xx, Han ethnicity, residing in xx.
Authorized Agent: Zhangxx, lawyer of xx Law Firm.

Defendant: Liu xx, male, born on xx, Han ethnicity, residing in xx.
Authorized Agents: Shen xx, lawyer of xx Law Firm.

The case of private loan dispute filed by the plaintiff, Lu xx, against the defendant, Liu xx, was accepted by this court on March 29, 2016. In accordance with the law, Judge Fang xx applied the summary procedure and publicly heard the case. The authorized agent of the plaintiff, Zhang xx, and the authorized agent of the defendant, Shen xx, appeared in court to participate in the litigation. The trial has now concluded.

The plaintiff, Lu xx, claimed that since October 2013, the defendant borrowed money from him for business turnover. ……. The plaintiff requested the court to order the defendant to repay the loan of RMB 707,100 and interest of RMB 22,921.66 (calculated at the six-month loan interest rate of the bank from July 1, 2015, to the date of filing, and subsequently at an annual interest rate of 4.35% until the actual repayment date), totaling RMB 730,021.66. The plaintiff also requested that the defendant bear the litigation costs.

The defendant, Liu xx, argued that although he issued the IOU for RMB 707,100, he did not actually receive the loan, and the plaintiff did not provide transfer vouchers. The defendant requested the court to dismiss all the plaintiff's claims.

After the trial, the court has ascertained that the defendant, Liu xx, requested a loan from the plaintiff, Lu xx, due to a shortage of turnover funds in October 2013. ……The plaintiff's efforts to recover the loan were unsuccessful, leading him to file a lawsuit with this court on March 29, 2016.

The above facts are evidenced by the photocopies of the ID cards of the plaintiff and his spouse Wang xx, …….

Holding: the defendant, Liu xx, borrowed RMB 707,100 from the plaintiff, Lu xx, as evidenced by the IOU and bank transfer receipts, which this court confirms. The IOU issued by the defendant specified a repayment period but did not specify interest, indicating a fixed-term interest-free loan. ……Therefore, in accordance with Article 90 of the General Principles of the Civil Law of the People's Republic of China, and Articles 206 and 207 of the Contract Law of the People's Republic of China, the judgment is as follows:

The defendant, Liu xx, shall repay the plaintiff, Lu xx, the loan principal of RMB 707,100 and overdue interest (calculated on the basis of RMB 707,100 from July 1, 2015, at the six-month bank loan interest rate until the principal is fully repaid) within ten days after this judgment takes effect.

If the defendant fails to fulfill the monetary obligations within the specified period, he shall pay double the interest on the debt for the period of delayed performance in accordance with Article 253 of the Civil Procedure Law of the People's Republic of China.

The case acceptance fee is RMB 11,100, halved to RMB 5,550, to be borne by the defendant, Liu xx.

If dissatisfied with this judgment, ppeal shall be brought by the dissatisfied party to the Intermediate People's Court of Hefei City of Anhui Province via this court within 15 days from the issuance of this decision in the number of copies corresponding to the number of adverse parties.

Judge: Fang xx
June 22, 2016
Clerk: Xi xx

Figure 6: CPL Judgement Demo (English Version).

C Details of TKGT's First Stage

Slack classes. To simplify KGs, we abstract it as two basic classes of *role entity classes* and *relation/action classes*. The former can represent any entity such as humans or objects, while the latter broadly represents relationships or behaviors that require multi-party participation.
Toolkits. We used existing NLP methods in TKGT.

[7] https://pkulaw.com/chl/4c13be0c1802426abdfb.html?way=listView
[8] https://www.court.gov.cn/susong.html

For Chinese, we use Hanlp's (He and Choi, 2021) sentence splitter as well as its integrated tokenizer, position tagger, and Chinese NER model. As for English, we use nltk's tokenizer and position tagger. *As for stop Words*, we use Chinese stop words from https://blog.csdn.net/qq_33772192/article/details/91886847 and English stop words from spaCy[9]. *As for stop position taggers*, due to the differences in the categories of parts of speech between Chinese and English, we choose positions to use based on the CTB tag set for Chinese, while the positions to disable based on the NLTK tag set for English as follows.

```
[
  used_pos_zh = ["NR", "NN", "CD", "VV",
                 "NT", "FW", "AD", "JJ" ],
  stop_pos_en = ["CC", "DT", "EX", "IN",
                "MD", "PDT", "POS", "PRP",
                 "RP", "SYM", "TO", "UH",
                 "WDT" , "WP"]
]
```

D Prompt Example

D.1 Information Extraction Prompt

We design the prompt to contain 3 parts as the IE task the model would complete would also follow three key steps: First, the assistant checks if the provided paragraph contains the attribute values corresponding to the role; if not, it responds with 'Bad Information'. Second, if the paragraph contains the relevant attribute values, the assistant extracts and provides the value according to the specified requirements. Third, the assistant responds to the user's question in the format of the provided in-context examples. Each example outlines the role, attribute, related context, value scope, question, and answer, ensuring the assistant's responses are precise and consistent. The relatively low F1 scores for the first column (Team name) extraction are due to the model's difficulty in identifying 'home team' and 'visiting team' from their positions in the text.

[9] https://spacy.io/

Query Rewrite Prompt Structure

{GENERAL_GUIDELINE_QUERY_RE}

<In-Context Example i>
Entity: {ENTITYi}
Target: {TARGETi}
Entity Description: {KGDESCi}
Naive Query: {QUERYi}
Question: Write a query to extract {TARGETi} of {ENTITYi}.
Answer: {QUERYi}

Below is the usr's question:
Entity: {ENTITY}
Target: {TARGET}
Entity Description: {KGDESC}
Naive Query: {QUERY}
Question: Write a query to extract {TARGET} of {ENTITY}.
Answer:

Figure 8: Structure of Query Rewrite Prompt.

Query Rewrite Prompt Structure

{GENERAL_GUIDELINE_QUERY_RE}

<In-Context Example i>
Entity: {ENTITYi}
Target: {TARGETi}
Entity Description: {KGDESCi}
Naive Query: {QUERYi}
Question: Write a query to extract {TARGETi} of {ENTITYi}.
Answer: {QUERYi}

Below is the usr's question:
Entity: {ENTITY}
Target: {TARGET}
Entity Description: {KGDESC}
Naive Query: {QUERY}
Question: Write a query to extract {TARGET} of {ENTITY}.
Answer:

Figure 9: Structure of Information Summary Prompt.

IE Prompt Structure

{GENERAL_GUIDELIN_IE}

<In-Context Example i>
Role: {ROLEi}
Attribute: {FIELDi}
Related Context: {RELATED_CONTEXTi}
Value scope: {SCOPEi}
Question: What's the value of {ROLEi}'s {FIELDi}?
Answer: {ANSWERi}

Below is the usr's question:
Role: {ROLE}
Attribute: {FIELD}
Related Context: {RELATED_CONTEXT}
Value scope: {SCOPE}
Question: What's the value of {ROLE}'s {FIELD}?
Answer:

Figure 7: Structure of Information Retrieving Prompt.

E Fine-tuning Setting

E.1 Fine-tuning Parameter and Setting

We use the open-source library LLaMA-Factory (Zheng et al., 2024) to fine-tune all models. LoRA (Hu et al., 2021) is used as the fine-tuning. The pre-trained weights are downloaded from the huggingface library (Wolf et al., 2020). We load the models with FP16 as the precision and optimize them with an Adam optimizer (Kingma and Ba, 2017). The learning rate is set to 1e-4 with cosine decay and the batch size is 2 per device. The maximum length for the input and generated sentence concatenation is 2048. We warm up the model with 3,000 steps and evaluate the model every 500 steps. A linear scheduler is also used. The LoRA rank is set to 16, and the α is set to 32.

E.2 Fine-tuning Data Preparation

In this subsection, we detail the data collection process for fine-tuning the Information Extraction (IE) model. Our approach to constructing the fine-tuning dataset aligns with the structure of the 'TKGT' framework. The IE model is employed only at the IE stage following 'Query Generation', 'Query Rewrite', and 'Information Summarizing'. To ensure consistency between the fine-tuning data and inference stages, we utilize pre-trained models for query rewriting and information summarizing.

Fine-tuning the IE model is crucial for enhancing the performance of 'Text-To-Table' tasks. Initially, the model learns to adhere to the specified output format. Subsequently, it differentiates between cases containing valid information (Good Information Case) and those that do not (Bad Information Case). Finally, the model identifies and extracts valid information accurately.

The fine-tuning dataset is composed in the following format:

```
[
  {"instruction": <ie task id>,
  "input": <ie prompt>,
  "output": <ground truth>},
  ...
]
```

F Computing Cost

F.1 Cost of Stage 1 Inference

Although we can measure the coverage of zero-shot and few-shot performance of KG generation, constructing an accurate domain-specific KG for information extraction depends on human expert judgment, the complexity of the text data, and the granularity of the information designed to be extracted to form the outcome table. For the E2E and Rotowire datasets, we report that LLaMa3-70B is able to construct acceptable KG classes with a single prompt. However, for more complex datasets like CPL, it requires significantly more iterations and human expert involvement in constructing the KG.

F.2 Cost of Stage 2 Inference

We can estimate the cost of stage 2 inference following the T-KG-T pipeline. For each document, suppose there are n variables in total and m variables are 'easy and obvious'[10] that can be easily extracted. For every variable that needs to go through the pipeline for extraction, it must undergo 'Query Rewrite', 'Information Summarization', 'Information Retrieving', and 'Information Extraction' processes, totaling 3 prompts and 1 retrieval. The algorithm ensures that each variable goes through the pipeline at most once.

Therefore, to extract the document, we would need a maximum of $3\times(n-m)$ model requests and $n - m$ retrievals[11]. For a typical CPL document, we extract around 150 variables, which implies an upper bound of 450 prompts and 150 retrieval actions. This translates to approximately 8 minutes of runtime on a single RTX 3090 GPU.

[10] In the CPL case, variables like 'case ID', 'court name', and 'date' are always in the same place in the legislation document (typically, these values are placed at a fixed location in the title, before the first paragraph, or at the end).

[11] The number of model requests and retrievals depends on the document's content. For example, if the defendant has not appeared in court, logic is added to avoid extracting variables that would only have non-null values when the defendant is present in court.