NAACL 2024

Exploring the Impact of Table-to-Text Methods on Augmenting LLM-based Question Answering with Domain Hybrid Data

Dehai Min, Nan Hu, Rihui Jin, Nuo Lin, Jiaoyan Chen,
Yongrui Chen, Yu Li, Guilin Qi, Yun Li, Nijun Li, Qianren Wang

Introduction

NAACL 2024

- Enhancing LLMs in Domain-Specific Question Answering
  - ➢ Domain-Specific Fine-Tuning (DSFT)
  - ➢ Retrieval-Augmented Generation (RAG)

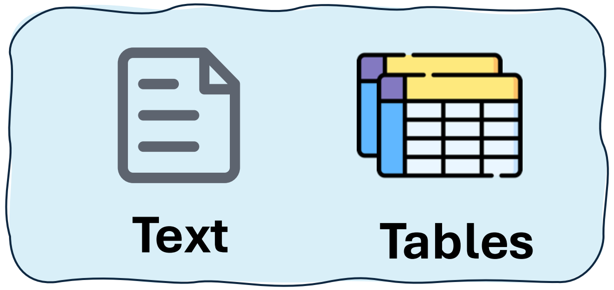
Both rely on domain-specific corpus

- Real-World Data Consists of Hybrid Data (Text and Tables)

Common in : Scientific Literature , Medical Reports, etc.

Tables alongside text provide :

- Supplementary or complementary information
- Enhancing the understanding of the content



Domain Documents

Current Methods and Their Drawbacks

- Method 1 : Flattening Tables (Concatenates table cells row by row)

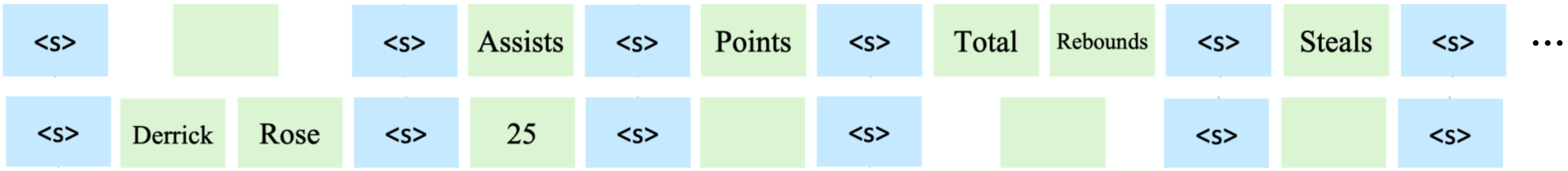
  Results in :

  ➢ The loss of structural information

  ➢ Disrupts the informational links between cells

  ➢ Introduces the non-natural language text

	Assists	Points	Total rebounds	Steals
Al Horford	5	15	7	3
Isaiah Thomas		27		
Marcus Smart	7	15		
Carmelo Anthony		29		
Kristaps Porzingis		22	12	
Derrick Rose		25		



Current Methods and Their Drawbacks

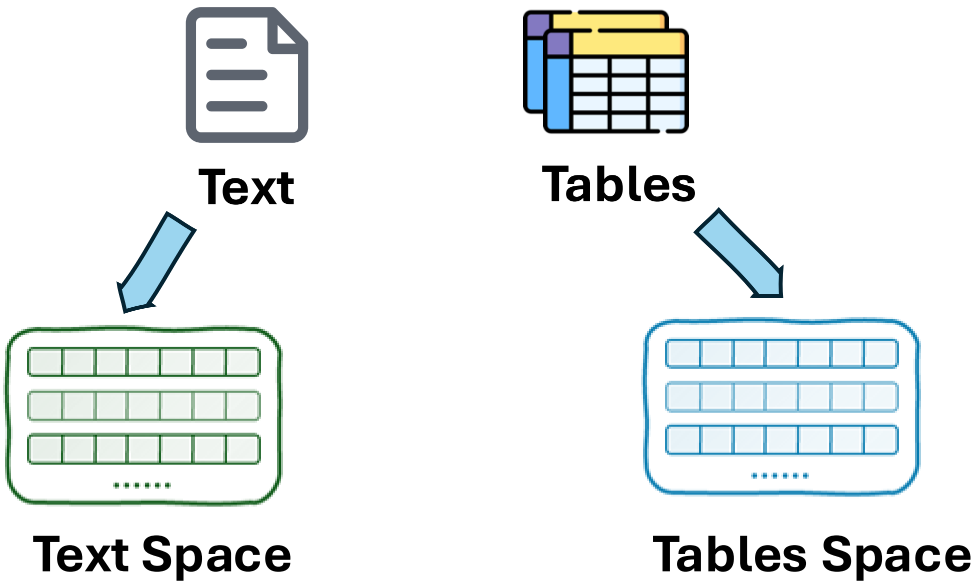
NAACL 2024

- Method 2 : Mapping Text and Tables to Different Vector Spaces

  Results in :

  ➢ Increases the complexity of system (needs multimodal models or multiple models)

  ➢ Disrupts the semantic connection between the two types of data (Text and Tables)



NAACL 2024

Table-to-Text Generation

- Generates natural language statements that faithfully describe the information in the provided table
- Four representative table-to-text strategies:
  - ❑ 1. Markdown format.
  - ❑ 2. Template serialization: a set of templates designed.
  - ❑ 3. TPLM-based method: fine-tuning Traditional PLM, like BART, on specific task datasets
  - ❑ 4. LLM-based method: ChatGPT, one-shot in-context learning setting.

Frenquency Band	Channel Bandwidth	Peak Data Rate
6 GHz	320 MHz	11.53 Gbps
5 GHz	160 MHz	5.765 Gbps
2.4 GHz	40 MHz	1.376 Gbps
. . .		

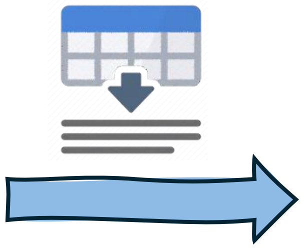


Table-to-Text Generation

The 6 GHz band offers a channel bandwidth of 320 MHz. It can reach a peak data rate of 11.53 Gbps (gigabits per second). The 5 GHz band has a channel bandwidth of 160 MHz. Its peak data rate is 5.765 Gbps ...

Advantages of Using Table-to-Text Generation

NAACL 2024

- Transforms hybrid data into a unified natural language representation
  - 1. Simplifies hybrid data scenarios into pure text scenarios
  - 2. Seamlessly integrates with any SOTA LLMs (which typically focus on text understanding and processing)
  - 3. Pure text format is easy for training domain-specific LLMs

- Preserves the semantic connections between the data
  - 1. Preserves the integrity of document content

    → beneficial for the model to learn a complete knowledge by finetune
  - 2. Facilitates information retrieval in RAG systems

Research Gap

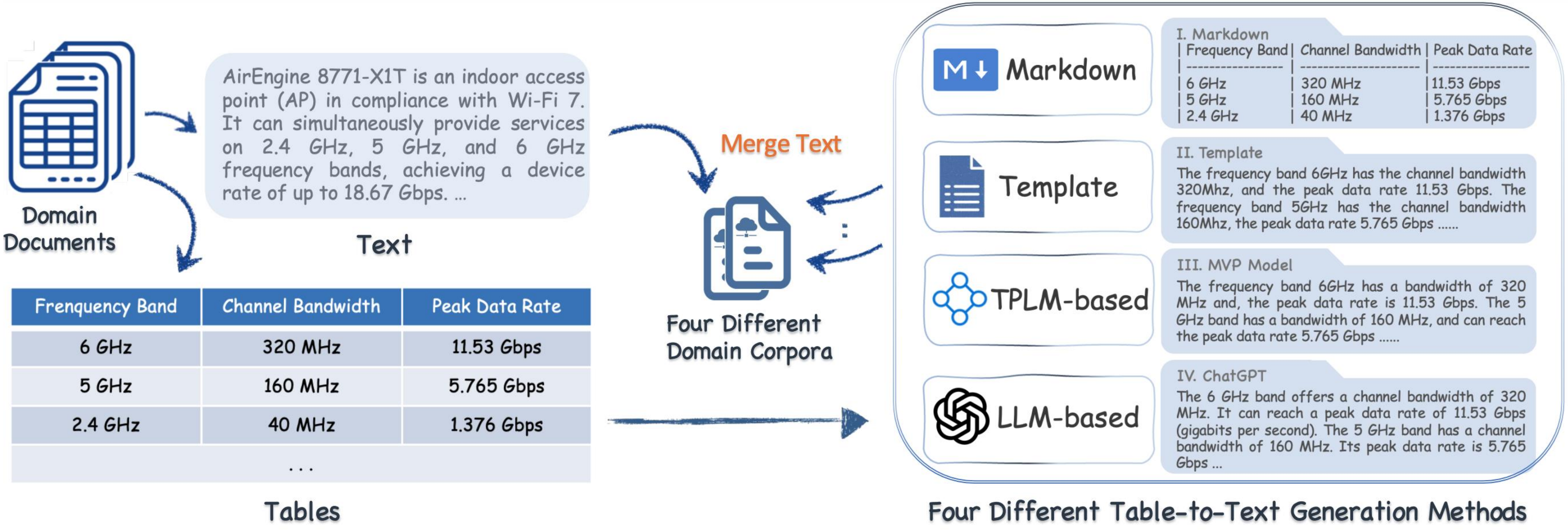
- The lack of comparative analysis on how different table-to-text methods affect the performance of domain-specific QA systems.

We address this research gap:

- Step 1: Innovatively integrates table-to-text generation into the LLM-based Domain QA framework
- Step 2: Conducts extensive experiments with different table-to-text methods on two types of QA systems

NAACL 2024

Building Domain Corpora with Table-to-text



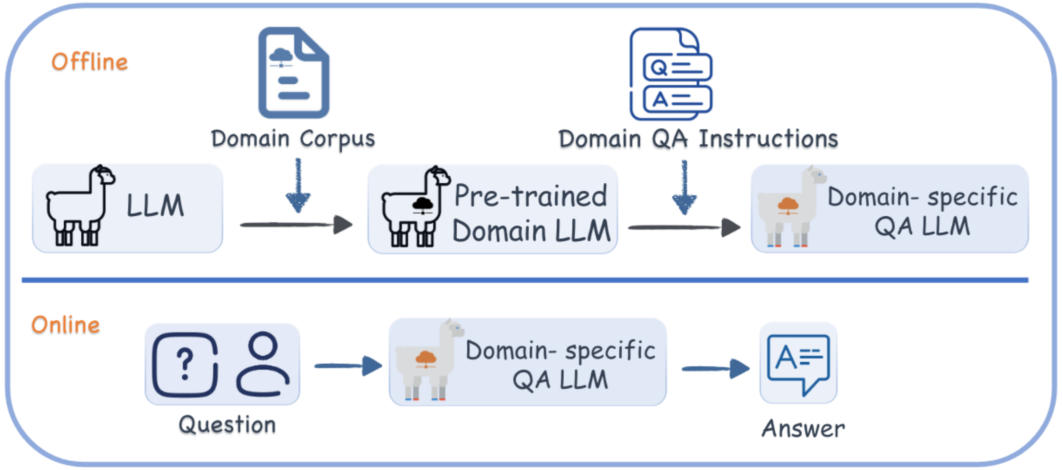
Building LLM-based QA Systems with Domain Corpora

System1 - DSFT:

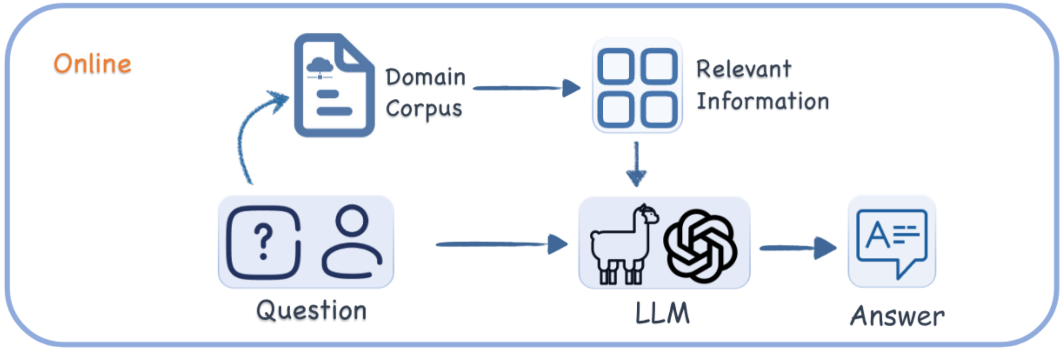
- Step 1: Incrementally pre-train the LLM on the domain corpus
- Step 2: Instruction tuning on the QA task

System 2 - RAG:

- LangChain framework
- Dense Passage Retriever (DPR) method for information retrieval



(a) Domain-Specific Fine-Tuning QA system



(b) Retrieval-Augmented Generation QA system

Dataset

NAACL 2024

ICT-DATA:

- Real-world industry hybrid dataset, English.
- Based on 170 technical documents related to ICT products
- 178 million words, 6GB text storage size
- Table data accounts for about 18% of the total word count

ICTQA:

- 9k questions with long-form answers
- Test set: 500 questions, whose answers involve knowledge from both tables and text.

ICT: Information and Communication Technology

Evaluation Metrics

Automated Evaluation:

- GPT-4 as an evaluator
- In-context learning: one demonstration
- Range: 0 to 5, discrete values. larger denotes better
- Based on helpfulness and similarity to the golden answer

Human Evaluation:

- 3 evaluators with domain knowledge
- Same scoring criteria with GPT-4

Experimental Setup

DSFT Paradigm:

- Meta's OPT (1.3B to 13B)
- Llama2-base (7B, 13B)
- QLoRA for pre-training and instruction fine-tuning

RAG Paradigm:

- Llama2-chat (7B, 13B, and 70B)
- GPT-3.5-turbo
- BGE model for DPR embedding
- Top-3 relevant text chunks based on similarity

Fair Comparison: the same settings on four different corpora.

Results and Analysis

NAACL 2024

Metrics	Table-to-Text Method	Domain-Specific Fine-Tuning OPT-1.3B	OPT-2.7B	OPT-6.7B	OPT-13B	Llama2-7B	Llama2-13B	Retrieval-Augmented Generation GPT-3.5-turbo	Llama2-7B	Llama2-13B	Llama2-70B
Human Eval.	Markdown	2.05	2.41	2.38	2.51	2.82	3.05	3.29	**3.72**	3.98	3.94
	Template	2.04	2.40	2.26	2.47	2.82	3.04	3.36	3.44	3.96	3.76
	TPLM-based	2.12	2.43	2.43	2.58	**3.20**	3.13	3.26	3.27	3.92	3.64
	LLM-based	**2.18**	**2.57**	**2.51**	**2.62**	2.96	**3.19**	**3.62**	3.71	**4.26**	**4.09**
	RSD(%)	2.80	3.40	5.00	3.00	7.60	3.00	7.20	9.00	6.80	9.00
GPT-4 Eval.	Markdown	1.74	2.16	2.27	2.25	2.7	3.06	3.28	**3.66**	3.67	**3.74**
	Template	1.81	2.22	2.39	2.34	2.84	3.08	3.27	3.06	3.38	3.37
	TPLM-based	2.33	2.46	2.45	2.53	**3.20**	3.19	3.28	2.9	3.41	3.30
	LLM-based	**2.57**	**2.69**	**2.73**	**2.86**	3.06	**3.30**	**3.64**	3.59	**3.69**	3.54
	RSD(%)	16.60	10.60	9.20	12.20	10.00	4.80	7.40	15.20	6.20	8.80

Relative Score Differences (RSD):

- 2.8% to 9.0% in human evaluation
- 4.8% to 16% in GPT4 evaluation

significantly impact the performance of systems

Performs well in DSFT paradigm:

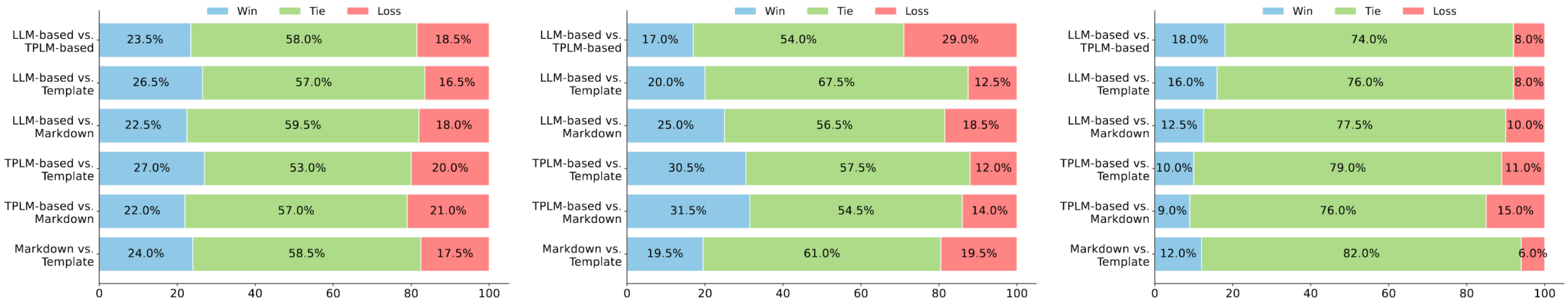
- LLM-based method
- TPLM-based method

Performs well in RAG paradigm:

- LLM-based method
- Markdown format (surprise!)

Results and analysis

NAACL 2024



(a) OPT-6.7B in DSFT Paradigm (b) Llama2-7B in DSFT Paradigm (c) Llama2-70B in RAG Paradigm

Comparison of human evaluation scores between QA models using different Table-to-Text methods.

‘A vs. B win’ indicates the percentage of test set instances where Model A’s score surpasses Model B’s.

Results and analysis

NAACL 2024

RQ: What are the potential reasons for their different performances?

In DSFT Paradigm:

Freq (k)	$C_1\cdot$ Markdown	$C_2\cdot$ Template	$C_3\cdot$ TPLM-based	$C_4\cdot$ LLM-based
Term	821	1040	2358	2254
Verbs	313	315	682	1207

Absolute frequency of verbs and terms contained in the corpora C_i generated by different methods.

higher frequency of domain-specific terms and verbs leads to better system performance.

- *LM-based methods tend to supplement the domain entities as subjects/objects.
- Template methods use more pronouns, and monotonous predicates.
- Markdown format only retains the original content in the tables.

Results and analysis

NAACL 2024

RQ: What are the potential reasons for their different performances?

In RAG Paradigm:

Under the same LLM reader setup:

Semantic representations quality

↓

Retrieval accuracy

↓

RAG performance

A t-SNE visualization of chunk clusters in the embedding space.

Retrieval-friendly method: LLM-based | Markdown format

Results and analysis

NAACL 2024

Some practical suggestions for choosing table-to-text methods

Ready-to-use tips

DSFT Paradigm:

- LLM-based method (Pros: best performance; Cons: GPU/API cost, Data leakage risks)
- TPLM-based(Can well-tuned on this task. A good alternative for LLM)

RAG Paradigm:

- LLM-based method
  - best performance
- Markdown format (viable substitute)
  - ✓ easy-to-use
  - ✓ GPU-Free

Freq (Avg.)	Markdown	Template	TPLM-based	LLM-based
Text Len	998	1259	1138	897

The average length of text generated by different methods for each table.

More Concise Text → Less Memory Costs
Less GPU Costs

Thank You

Dehai Min

Master Student

Southeast University & Monash University

Homepage: https://zhishanq.github.io/