Computer Engineering and Applications Vol. 15, No. 2, June 2026

LoTQA: Local Benchmarking of Large Language Models for Table Question Answering

Muhammad Arya All Fajri[1], Muhammad Ikhsan Riski Pratama[2], Firdaus[3], Abdiansah[4]

Department of Computer Engineering,Faculty of Computer Science, Universitas Sriwijaya
**09012682529003@student.unsri.ac.id*

ABSTRACT

TableQA plays an important role in supporting data-driven decision making and improving information retrieval efficiency. The use of Large Language Models (LLMs) through cloud services or external APIs enables systems to automatically understand table structures and question contexts, perform generalisation, contextual reasoning, and understand semantic relationships between entities in tables to generate more relevant and accurate answers. This approach results in significant increases in computational costs, potential data security risks, and limitations in model development, customisation, and testing. This research proposes LoTQA for the TableQA task. LoTQA is an approach that utilises local execution to evaluate and compare LLM methods in generating answers from structured table data. Performance evaluation on LoTQA (Qwen3:4b, LoRA Fine-tuned) obtained SacreBLUE of 8.613, BLEU-1 of 35.623, BLEU-2 of 26.592, BLEU-3 of 22.723, ROUGE-1 of 0.364, ROUGE-2 of 0.177, ROUGE-L of 0.311, and METEOR of 0.317. These results indicate that the LoTQA method is quite good at providing semantically meaningful sentences for predictions, even with low resources. The results of the performance evaluation for each LLM model used show that the Qwen3:4b model achieved the highest scores for SacreBLEU, ROUGE-1, ROUGE-2, ROUGE-L, and METEOR. This study shows that LoTQA performs quite well on the TableQA task, despite low resources.

Keywords: LoTQA, Large Language Models (LLMs), Table Question Answering, TableQA

1. INTRODUCTION

Table Question Answering (TableQA) is a field in natural language processing that focuses on table-based question answering systems, where the system must understand the table structure and provide relevant answers to user questions [1], [2]. TableQA is used to support data-driven decision-making, improve information retrieval efficiency, and facilitate access to structured data. Manual answer retrieval is difficult because it requires time, special skills, and difficulty in handling complex natural language questions [3]. Answer retrieval in TableQA can be done more effectively by utilising Large Language Models (LLMs) that are capable of understanding natural language questions and table structures automatically. LLMs have advantages in generalisation, contextual reasoning, and understanding semantic relationships between entities in tables to produce more relevant and accurate answers [4].

The application of LLM on TableQA is generally carried out through cloud-based external API services. Cloud-based execution results in a significant increase in computing costs and has the potential to pose data security risks. The use of external APIs limits model development and customisation, and hinders the testing process. Local execution (on-device or local runtime) is a solution because it reduces costs, improves security, and provides full control over model development, customisation, and testing [4].

Research on TableQA using LLM has been conducted by various studies. [5] applied the SynTab-LLaVA approach based on LLM to the FeTaQA dataset with a BLEU score of 35.45. [3] applied the T5-large architecture combined with TAPAS, fine-tuned, and seq2seq to the FeTaQA dataset, but the research was not optimal with a BLEU score of 30.54, ROUGE-1 of 0.63, ROUGE-2 of 0.41, ROUGE-L of 0.53, and METEOR of 0.49. [6] applied an LLM-based Tree-Of-Table approach to the FeTaQA dataset with a BLEU score of 34.73, ROUGE-1 of 0.68, ROUGE-2 of 0.46, and ROUGE-L of 0.58. [7] applied an LLM-based Chain-of-Table approach to the FeTaQA dataset with a BLEU score of 32.61, ROUGE-1 of 0.66, ROUGE-2 of 0.44, and ROUGE-L of 0.56.

This study proposes LoTQA for TableQA tasks. LoTQA is an approach that utilises local execution to evaluate and compare LLM methods in generating answers from structured table data. This approach aims to reduce the high computational costs of cloud-based external API services, improve security, and facilitate model development, customisation, and testing. This research compares LLM methods to determine the most effective method for the TableQA task. The success rate of the methods proposed in this research is measured by calculating the performance results of SacreBLEU, BLEU-1, BLEU-2, BLEU-3, ROUGE-1, ROUGE-2, ROUGE-L, and METEOR.

2. RELATED WORK

Research on TableQA has been conducted by various studies. [8] applied the GPT-3.5-turbo architecture combined with Relevant-Cell-based Rationales on the FeTaQA dataset with a SacreBLEU score of 41.3, ROUGE-1 of 0.70, ROUGE-2 of 0.50, ROUGE-L of 0.61, and METEOR of 0.58. However, this research was still conducted through an external cloud-based API service. [9] applied the GPT-3.5-turbo architecture combined with Text-Davinci-003 and ReAcTable on the FeTaQA dataset with a ROUGE-1 score of 0.71, ROUGE-2 of 0.46, and ROUGE-L of 0.61, however, this research is still conducted through an external cloud-based API service. [10] applied the GPT-3-Codex architecture combined with dater on the FeTaQA dataset with a SacreBLEU score of 39.92, ROUGE-1 of 0.66, ROUGE-2 of 0.45, and ROUGE-L of 0.56. however, the research is still conducted through an external cloud-based API service. [11] applied the GPT-3.5-turbo architecture combined with instruction, LoRA, Llama3-8b and TQAgent with a SacreBLEU score of 28.62 and ROUGE-1 of 0.61, but the research was still conducted through an external cloud-based API service. [12] applied Reason SFT (Supervised Fine-Tuning) combined with LLM-based RL (Reinforcement Learning) on the FeTaQA dataset with a SacreBLEU score of 43.18, but the research was still conducted through an external cloud-based API service. [13] applied the GPT-4 architecture combined with mistral-7b and cliff (based on Named Entity Recognition (NER) models and MaskRel) on the FeTaQA

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

dataset with a SacreBLEU score of 41.88, but the research was still conducted through an external cloud-based API service. [14] applied LLM-based CABINET (Content Relevance-Based Noise Reduction for Table Question-Answering) on the FeTaQA dataset with a SacreBLEU score of 40.5, but the research was still conducted through an external cloud-based API service. [15] applied the TableLlama architecture (LLaMA2-7b fine-tuned) combined with LongLoRA and TableInstruct on the FeTaQA dataset with a SacreBLEU score of 39.05, but the research was still conducted through an external cloud-based API service. [16] applied the Phi3-7b architecture combined with instruction, GPT-3.5-turbo, and GPT-4-turbo on the FeTaQA dataset with a SacreBLEU score of 38.13, but the research was still conducted through an external cloud-based API service. [17] applied the HeLM architecture (LLaMA2-13b fine-tuned) combined with QLoRa and ChatGPT-based reasoning distillation on the FeTaQA dataset with a SacreBLEU score of 36.74, ROUGE-1 of 0.696, ROUGE-2 of 0.482, and ROUGE-L of 0.595, but the research was still conducted through an external cloud-based API service. [18] applied the GenTaP architecture (BART with intermediate pre-training) on the FeTaQA dataset with a SacreBLEU score of 36.74, ROUGE-1 of 0.689, ROUGE-2 of 0.476, ROUGE-L of 0.587, and METEOR of 0.545, but the research is still being conducted through an external cloud-based API service. [19] applied a GPT-3 architecture combined with GPT-3, Codex, T5-3b, and UnifiedSKG on the FeTaQA dataset, achieving a SacreBLEU score of 33.44, a ROUGE-1 score of 0.6521, ROUGE-2 of 0.4309, ROUGE-L of 0.5531, and METEOR of 0.5123. However, this research was still conducted through an external cloud-based API service. [20] GPT-3.5-turbo-0125 architecture combined with fine-tuning, triples, and RAG on the FeTaQA dataset with a SacreBLEU score of 31.3, ROUGE-1 of 0.67, ROUGE-2 of 0.44, and ROUGE-L of 0.55, but the research is still conducted through an external cloud-based API service. [21] applied the GPT-3.5-turbo architecture combined with EnoTab on the FeTaQA dataset with a SacreBLEU score of 30.46, ROUGE-1 of 0.67, ROUGE-2 of 0.45, and ROUGE-L of 0.57, but the research is still conducted through an external cloud-based API service. [22] applied the GPT-4o architecture combined with TableMaster on the FeTaQA dataset with a SacreBLEU score of 28.94, ROUGE-1 of 0.6606, ROUGE-2 of 0.4529, and ROUGE-L of 0.5456, but the research is still being conducted through an external cloud-based API service. [23] applied the GPT-4 architecture combined with 2-shot CoT on the FeTaQA dataset with a BLEU-1 score of 62.2, BLEU-2 of 48.6, BLEU-3 of 39.2, ROUGE-1 of 0.6606, ROUGE-2 of 0.4529, and ROUGE-L of 0.5456. but the research is still being conducted through an external cloud-based API service.

3. METHOD

Text preprocessing is the process of converting the original language into a more structured format by removing information that has no significant meaning so that it can be processed accurately using a model. Text preprocessing is carried out by matching letter forms and removing punctuation marks that have no significant meaning [24]. The stages of text preprocessing in the FeTaQA dataset applied in this study can be seen in Figure 1.

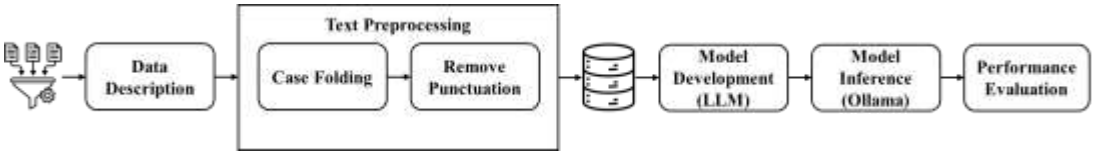

FIGURE 1. Illustration of Research Methodology Flow.

Figure 1 illustrates the research methodology process flow. The initial stage of the research methodology flow is data description, which is carried out to explain the characteristics and structure of the raw data used. The text preprocessing stage is carried out to standardise letters to lowercase (case folding) and remove punctuation, so that the dataset can be processed by the model. The model development (LLM) stage is carried out to adjust the model to the research dataset through a fine-tuning process so that it is able to understand the context of the table and specific questions. The model inference (Ollama) stage is carried out to run the prediction process using LLM so that the output is a response to the question. The final stage is performance evaluation, which is carried out to assess the performance of the model based on the evaluation metrics used.

3.1 DATA DESCRIPTION

The data used in this study is the FeTaQA dataset in English obtained from GitHub [25]. The FeTaQA dataset contains 10,330 questions and answers in English for TableQA tasks with descriptive (free-form answer) answer types. The FeTaQA dataset consists of 7,326 training data, 1,001 validation data, and 2,003 test data. Each row in the FeTaQA dataset contains questions and answers that are interrelated and require reasoning based on the available information.

3.2 TEXT PREPROCESSING

Text preprocessing is the process of converting the original language into a more structured format by removing information that has no significant meaning so that it can be processed accurately using a model. Text preprocessing is carried out by matching letter forms and removing punctuation marks that have no significant meaning [24]. The stages of text preprocessing in the FeTaQA dataset applied in this study can be seen in Figure 2.

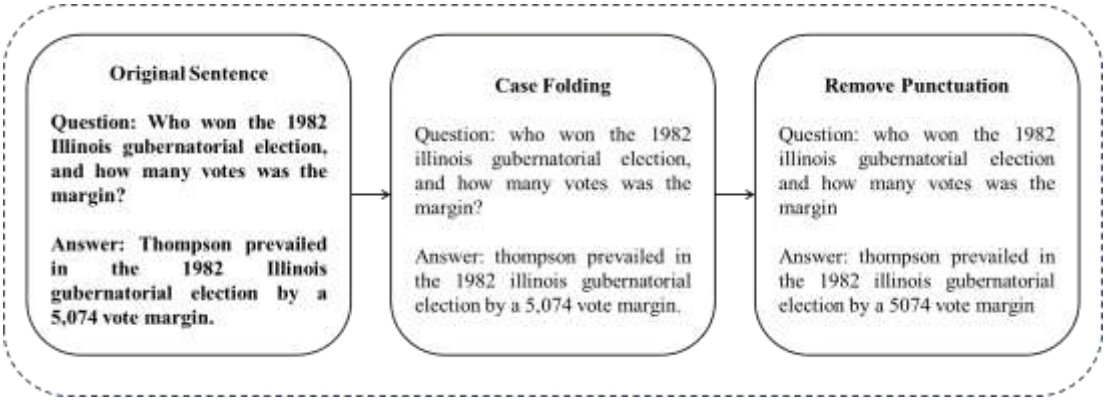

FIGURE 2. Illustration of Text Preprocessing Stages in the FeTaQA Dataset.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

Figure 2 illustrates the text preprocessing stages in the FeTaQA dataset. The text preprocessing stage begins with case folding, which converts capital letters to lowercase letters. The punctuation removal stage is applied to remove punctuation marks that have no significant meaning. The result of text preprocessing is text that has been cleaned of irrelevant elements, making it more structured, ready for further analysis, and capable of being accurately processed using Large Language Models (LLM).

3.3 DEVELOPMENT OF LARGE LANGUAGE MODELS (LLM)

The model used in this study is based on Large Language Models (LLM). LLM is a highly complex artificial intelligence system that has the ability to learn from a large amount of available text data [26]. This study utilises several LLM variants, namely TinyDolphin:1.1B, Granite3.1-MoE:1B, Falcon3:1B, Granite3.1-MoE:3B, LlaMa3.2:3b, Qwen3:4b, LlaMa3.1: latest, Qwen2.5:7b, Mistral:7B, and Falcon3:7b. The LLM models used each have a parameter range from 1 to 7 billion. The LLM models were developed by adjusting the model instructions using system prompts and fine-tuning. The fine-tuning stage was used to adjust the models to the FeTaQA dataset. The development process was carried out using the Parameter-Efficient Fine-Tuning (PEFT) approach to efficiently optimise some parameters by adjusting the LLM model parameters, as well as reducing computational costs and memory usage [27]. One of the most common PEFT methods is Low-Rank Adaptation (LoRA).

LoRA reduces memory and computational usage for fine-tuning by updating a portion of the low-rank weight matrices that can be trained, without making changes to the overall main model weights [28]. The PEFT approach, such as LoRA, enables the LLM model adjustment process to be carried out efficiently in terms of memory usage and training time while maintaining the performance of the base model [29]. The model training process was carried out with the following fine-tuning parameter configuration: *max_seq_length* = 2048, *rank* (r) = 24, *lora_alpha* = 32, *lora_dropout* = 0.05, *per_device_train_batch_size* = 4, *gradient_accumulation_steps* = 2, *warmup_steps* = 200, *num_train_epochs* = 3, *learning_rate* = 5×10^{-5}, *logging_steps* = 10, *optim* = "*adamw_8bit*", *weight_decay* = 0.01, *lr_scheduler_type* = "*linear*", *seed* = 3407. The result of developing an LLM model to optimise the model against the FeTaQA dataset in order to produce more accurate predictions at the model inference stage.

3.4 MODEL INFERENCE (OLLAMA)

The model inference stage is carried out using the Ollama platform, which functions as a system for running fine-tuned models locally. The fine-tuned model is run through a system prompt (system instruction) containing questions and instructions in accordance with the FeTaQA dataset. Ollama processes the system prompt (system instruction) using a model that has been loaded together with the LoRA adapter training results, which contain fine-tuning parameters to adjust the model to the FeTaQA dataset. The model inference process is carried out with the following parameter configuration: *temperature* = 0.3, *top_p* = 0.9, *repeat_penalty* =

1.1, *num_ctx* = 4096, *num_predict* = 256. The results of model inference are text outputs of model answers to each question in the FeTaQA dataset. These outputs are used for performance evaluation using the SacreBLEU, BLEU-1, BLEU-2, BLEU-3, ROUGE-1, ROUGE-2, ROUGE-L, and METEOR evaluation metrics to measure the level of conformity of model answers with reference answers in the FeTaQA dataset.

3.5 PERFORMANCE EVALUATION

The evaluation stages were conducted to measure the performance of the method after testing it on TableQA tasks. Performance evaluation on TableQA was carried out using metrics that assess the similarity between the model's answers and the reference answers [30]. The performance evaluation measures used in this study are BLEU, ROUGE, and METEOR [31].

BLEU (Bilingual Evaluation Understudy) is an evaluation metric that measures the conformity between the model's output sentences and human reference sentences based on word order [32]. The BLEU evaluation metrics used are SacreBLEU, BLEU-1, BLEU-2, and BLEU-3 [33].

SacreBLEU is a standardised version of BLEU designed to produce uniform and consistent BLEU calculations [34]. The SacreBLEU calculation process can be seen in Equation (1) [35].

$$SacreBLEU = BP \cdot exp(\sum_{i=1}^{n} \quad w_i log\ p_i) \tag{1}$$

where

$$BP = \begin{cases} e^{(1-\frac{r}{c})}, & if\ c<r \\ 1, & if\ c \geq r \end{cases}$$

BP is a brevity penalty to adjust the score so that shorter texts do not receive excessive scores, i is the i-th n-gram, n is the n-gram, w_i is the weight of the i-th n-gram, p_i is the i-th precision, c is the length (number of words) of the model's output sentence (candidate), r is the length (number of words) of the reference sentence.

BLEU-1 is an evaluation metric that measures the similarity between the model's output unigram sentence and human reference sentences based on word order [34]. The BLEU-1 calculation process can be seen in Equation (2) [36].

$$BLEU-1 = BP \cdot exp(log\ p_1) \tag{2}$$

where

$$BP = \begin{cases} e^{(1-\frac{r}{c})}, & if\ c<r \\ 1, & if\ c \geq r \end{cases}$$

BP is a brevity penalty to adjust the score so that shorter texts do not receive excessive scores, i is the i-th n-gram, n is the n-gram, p_1 is the 1-th precision, c is the length (number of words) of the model's output sentence (candidate), r is the length (number of words) of the reference sentence.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

BLEU-2 is an evaluation metric that measures the similarity between the model's output bigram sentence and human reference sentences based on word order [37]. The BLEU-2 calculation process can be seen in Equation (3).

$$BLEU - 2 = BP \cdot exp\left(\frac{1}{2} log\ p_1 + \frac{1}{2} log\ p_2\right) \quad (3)$$

where

$$BP = \begin{cases} e^{(1-\frac{r}{c})}, & if\ c<r \\ 1, & if\ c \geq r \end{cases}$$

BP is a brevity penalty to adjust the score so that shorter texts do not receive excessive scores, i is the i-th n-gram, n is the n-gram, p_1 is the 1-th precision, p_2 is the 2-th precision, c is the length (number of words) of the model's output sentence (candidate), r is the length (number of words) of the reference sentence.

BLEU-3 is an evaluation metric that measures the similarity between the model's output trigram sentence and human reference sentences based on word order [38]. The BLEU-3 calculation process can be seen in Equation (4).

$$BLEU - 3 = BP \cdot exp\left(\frac{1}{3} log\ p_1 + \frac{1}{3} log\ p_2 + \frac{1}{3} log\ p_3\right) \quad (4)$$

where

$$BP = \begin{cases} e^{(1-\frac{r}{c})}, & if\ c<r \\ 1, & if\ c \geq r \end{cases}$$

BP is a brevity penalty to adjust the score so that shorter texts do not receive excessive scores, i is the i-th n-gram, n is the n-gram, p_1 is the 1-th precision, p_2 is the 2-th precision, p_3 is the 3-th precision, c is the length (number of words) of the model's output sentence (candidate), r is the length (number of words) of the reference sentence.

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is an evaluation metric that measures the degree of lexical overlap between the reference text and the prediction [39]. The most commonly used ROUGE evaluation metrics are ROUGE-1, ROUGE-2, and ROUGE-L [40].

ROUGE-1 is an evaluation metric that measures the amount of unigram or single-word overlap between the reference text and the prediction [41]. The ROUGE-1 calculation process can be seen in Equation (5) [42].

$$ROUGE - 1 = 2 \times \frac{R_1 \times P_1}{R_1 + P_1} \quad (5)$$

Where

$$R_1 = \frac{Number\ of\ matching\ unigrams\ between\ prediction\ and\ reference}{Number\ of\ unigrams\ in\ the\ reference}$$

$$P_1 = \frac{Number\ of\ matching\ unigrams\ between\ prediction\ and\ reference}{Number\ of\ unigrams\ in\ the\ prediction}$$

R_1 is the recall value in ROUGE-1, P_1 is the precision value in ROUGE-1.

ROUGE-2 is an evaluation metric that measures the overlap of bigrams or adjacent word pairs between the reference text and the prediction [43]. The ROUGE-L calculation process can be seen in Equation (6) [44].

$$ROUGE - 2 = 2 \times \frac{R_2 \times P_2}{R_2 + P_2} \tag{6}$$

Where

$$R_2 = \frac{Number\ of\ matching\ bigrams\ between\ prediction\ and\ reference}{Number\ of\ bigrams\ in\ the\ reference}$$

$$P_2 = \frac{Number\ of\ matching\ bigrams\ between\ prediction\ and\ reference}{Number\ of\ bigrams\ in\ the\ prediction}$$

R_2 is the recall value in ROUGE-2, P_2 is the precision value in ROUGE-2.

ROUGE-L is an evaluation metric that measures the similarity between reference text and prediction by finding the longest sequence of identical words between the two [31]. The ROUGE-L calculation process can be seen in Equation (7) [45].

$$ROUGE - L = \frac{(1+\beta^2) \times R_L P_L}{R_L + \beta^2 P_L} \tag{7}$$

Where

$$R_L = \frac{L(X,Y)}{m}$$

$$P_L = \frac{L(X,Y)}{n}$$

β is the recall weight against precision, X is the reference text, Y is the predicted text, $L(X,Y)$ is the longest sequence of words that appear in the same order between the reference text (X) and the predicted text (Y), R_L is the recall value in L, P_L is the precision value in L, m is the length (number of words) of the reference text, n is the length (number of words) of the predicted text.

METEOR (Metric for Evaluation of Translation with Explicit ORdering) is an evaluation metric that measures the semantic similarity between the model's output sentences and human reference sentences based on word order, synonyms, and root words [46]. The METEOR calculation process can be seen in Equation (8) [47].

$$METEOR = F_{Mean} \times (1 - Penalty) \tag{8}$$

Where

$$F_{Mean} = \frac{10PR}{R + 9P}$$

$$Penalty = 0.5 \times (\frac{h}{u})^3$$

F_{Mean} is the harmonic mean between precision (P) and recall (R). $Penalty$ is a reduction factor used to lower the score when the word order in the model's output text differs from the word order in the human reference text. P is precision (the ratio between the number of matching words between the model's output text and the human reference text and the total number of words in the model's output text). R is recall (the ratio between the number of matching words and the total number of words in the human reference text). h is the number of chunks (sequential groups of matching words), u is the number of matching words.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

4. RESULT AND DISCUSSION

4.1 ANALYSIS AND INTERPRETATION OF RESULT

The analysis and interpretation stages involved comparing the results of the model performance evaluation on the TableQA task with previous studies and comparing the performance evaluation results for each LLM model used. The model performance evaluation results were obtained from testing using NVIDIA RTX4070 12GB and 32GB RAM (16GB already in use). A comparison of the performance evaluation results for each LLM model used can be seen in Table 1.

TABLE 1.
Comparison of Performance Evaluation Results for Each LLM Model Used.

Model LLM	SacreBLUE	ROUGE-1	ROUGE-2	ROUGE-L	METEOR
TinyDolphin:1.1B	3.285	0.2218	0.0895	0.1739	0.3853
Granite3.1-MoE:1B	3.064	0.2197	0.0862	0.1710	0.3941
Granite3.1-MoE:3B	3.356	0.2283	0.0937	0.1780	0.4169
Falcon3:1B	3.397	0.1900	0.0809	0.1524	0.3414
LLaMa3.2:3b **(Fine-Tuned Model)**	2.447	0.1683	0.0683	0.1335	0.3422
Qwen3:4b **(Proposed Model)**	**5.222**	**0.2620**	**0.1172**	**0.2117**	**0.3707**
Llama3.1:latest	2.742	0.1805	0.0765	0.1420	0.3771
Qwen2.5:7b	2.773	0.1861	0.0773	0.1453	0.3830
Mistral:7B	3.053	0.2051	0.0867	0.1606	0.4111
Falcon3:7b	3.217	0.1837	0.0799	0.1473	0.3513

Based on Table 1, a comparison of the performance results of each LLM model used based on the ROUGE and METEOR evaluation metrics in the TableQA task. The performance evaluation results for each LLM model used show that the Qwen3:4b model has the highest scores for SacreBLEU, ROUGE-1, ROUGE-2, ROUGE-L, and METEOR, at 5.222, 0.2620, 0.1172, 0.2117, and 0.3707, respectively. The 7B parameter model cannot be run using LoRA fine-tuning due to low resources. The model proposed in this study is Qwen3:4B, which shows the best performance and can be run using LoRA fine-tuned. The LLaMA3.2:3B model was fine-tuned using LoRA because it has the largest parameters after Qwen3:4B. A comparison of performance evaluation results for the proposed method with previous studies based on the BLEU evaluation metric can be seen in Table 2.

TABLE 2.
Comparison of Performance Evaluation Results in the Proposed Method with Previous Research Based on the BLEU Evaluation Metric.

Method	Runtime Environment	Result			
		SacreBLUE	BLEU-1	BLEU-2	BLEU-3
GPT-3.5-turbo and Relevant-Cell-based Rationales [8]	Cloud-based external API	41.3	-	-	-
GPT-3-Codex and Dater [10]	Cloud-based external API	30.92	-	-	-
GPT-3.5-turbo, Instruction, LoRA , Llama3-8b, and TQAgent [11]	Cloud-based external API	28.62	-	-	-
Reason SFT-RL [12]	Cloud-based external API	**43.18**	-	-	-
GPT-4, mistral-7b, and cliff [13]	Cloud-based external API	41.88	-	-	-
CABINET [14]	Cloud-based external API	40.5	-	-	-
TableLlama, LongLoRA, and TableInstruct [15]	Cloud-based external API	39.05	-	-	-
Phi3-7b, Instruction, GPT-3.5-turbo, and GPT-4-turbo [16]	Cloud-based external API	38.13	-	-	-
HeLM, QLoRA, and ChatGPT-Based Reasoning Distillation [17]	Cloud-based external API	36.74	-	-	-
GenTaP [18]	Cloud-based external API	36.74	-	-	-
GPT-3, Codex, T5-3b, and UnifiedSKG [19]	Cloud-based external API	33.44	-	-	-
GPT-3.5-turbo-0125, Fine-tuned, Triples, and RAG [20]	Cloud-based external API	31.3	-	-	-
GPT-3.5-turbo and EnoTab [21]	Cloud-based external API	30.46	-	-	-
GPT-4.o and TableMaster [22]	Cloud-based external API	28.94	-	-	-
GPT-4 and 2-shot CoT [23]	Cloud-based external API	-	**62.2**	**48.6**	**39.2**
LoTQA (LLaMa3.2:3b, LoRA Fine-tuned)	**Local**	5.791	15.411	9.53	7.151

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

LoTQA (Qwen3:4b, LoRA Fine-tuned) (Proposed Research)	**Local**	8.613	35.623	26.592	22.723

Based on Table 2, a comparison of the performance results of the proposed method with several other methods using the FeTaQA dataset based on the BLEU evaluation metric in the TableQA task. Previous studies only calculated the SacreBLEU performance evaluation score, except for the study by [23]. The study [12] produced the best SacreBLEU value compared to other previous studies, but it only calculated the SacreBLEU performance evaluation value and was still run through a cloud service or external API. The study [23] produced the best BLEU-1, BLEU-2, and BLEU-3 scores compared to other previous studies, but it did not calculate the SacreBLEU performance evaluation score and was still run through a cloud service or external API. The results of LoTQA (Qwen3:4b, LoRA Fine-tuned) produced the best SacreBLEU, BLEU-1, BLEU-2, and BLEU-3 scores compared to LoTQA (LLaMa3.2:3b, LoRA Fine-tuned). The SacreBLEU, BLEU-1, BLEU-2, and BLEU-3 results of the method proposed in this study show fairly good performance compared to the methods used in other studies, despite low resources. The SacreBLUE results show that LoTQA performs quite well in measuring the similarity between the model's output sentences and human reference sentences based on word order using the standard version of BLEU, which is designed to produce uniform and consistent calculations, despite low resources. BLUE-1 shows that LoTQA performs quite well in measuring the unigram alignment between model output sentences and human reference sentences based on word order, despite low resources. BLUE-2 shows that LoTQA performs quite well in measuring bigram alignment between model output sentences and human reference sentences based on word order, despite low resources. BLUE-3 shows that LoTQA performs quite well in measuring the trigram alignment between model output sentences and human reference sentences based on word order, despite low resources. A comparison of the performance evaluation results of the proposed method with previous studies can be seen in Table 3.

TABLE 3.
Comparison of Performance Evaluation Results in the Proposed Method with Previous Research Based on ROUGE and METEOR Evaluation Metrics

Mehod	Runtime Environment	Result			
		ROUGE-1	ROUGE-2	ROUGE-L	METEOR
GPT-3.5-turbo and Relevant-Cell-based Rationales [8]	Cloud-based external API	0.70	**0.50**	**0.61**	**0.58**
GPT-3.5-turbo, Text-Davinci-003, and ReAcTable [9]	Cloud-based external API	**0.71**	0.46	**0.61**	-

GPT-3-Codex and Dater [10]	Cloud-based external API	0.66	0.45	0.56	-
GPT-3.5-turbo, Instruction, LoRA , Llama3-8b, and TQAgent [11]	Cloud-based external API	0.61	-	-	-
HeLM, QLoRA, and ChatGPT-Based Reasoning Distillation [17]	Cloud-based external API	0.696	0.482	0.595	-
GenTaP [18]	Cloud-based external API	0.689	0.476	0.587	0.545
GPT-3, Codex, T5-3b, and UnifiedSKG [19]	Cloud-based external API	0.6521	0.4309	0.5531	0.5123
GPT-3.5-turbo-0125, Fine-tuned, Triples, and RAG [20]	Cloud-based external API	0.67	0.44	0.55	-
GPT-3.5-turbo and EnoTab [21]	Cloud-based external API	0.67	0.45	0.57	-
GPT-4.o and TableMaster [22]	Cloud-based external API	0.6606	0.4529	0.5456	-
GPT-4 and 2-shot CoT [23]	Cloud-based external API	0.658	0.428	0.544	-
LoTQA (LLaMa3.2:3b, LoRA Fine-tuned)	**Local**	0.328	0.144	0.277	0.307
LoTQA (Qwen3:4b, LoRA Fine-tuned) (Penelitian yang diusulkan)	**Local**	0.364	0.177	0.311	0.317

Based on Table 3, a comparison of the performance results of the proposed method with several other methods using the FeTaQA dataset based on the ROUGE and METEOR evaluation metrics for the TableQA task. Research [11] only calculated the ROUGE-1 performance evaluation scores. Research [10], [17], [20], [21], [22], [23] did not calculate the METEOR performance evaluation score. Research [18], [19] is still conducted through cloud services or external APIs. Research [9] produced the best ROUGE-1 and ROUGE-L scores compared to other previous studies, but this study did not calculate the METEOR performance evaluation scores and was still conducted through cloud services or external APIs. The study [8] produced the best ROUGE-2, ROUGE-L, and METEOR scores compared to other previous studies, but the study was still conducted through cloud services or external APIs. The results of LoTQA (Qwen3:4b, LoRA Fine-tuned) produced the best ROUGE-1, ROUGE-2, ROUGE-L, and METEOR scores compared to LoTQA (LLaMa3.2:3b, LoRA Fine-tuned). The ROUGE-1, ROUGE-2, ROUGE-L, and METEOR results of the proposed method in this study show fairly good performance compared to the methods used in

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

other studies, despite low resources. The ROUGE-1 results show that LoTQA performs quite well in measuring the unigram overlap between the reference text and the prediction, despite low resources. The ROUGE-2 results show that LoTQA performs quite well in measuring the bigram overlap between the reference text and the prediction, despite low resources. The ROUGE-L results show that LoTQA performs quite well in measuring the similarity between the reference text and the prediction by finding the longest common sequence of words between the two, despite low resources. The METEOR results show that LoTQA performs quite well in measuring the semantic similarity between the model's output sentences and human reference sentences based on word order, synonyms, and root words, despite low resources.

5. CONCLUSION

This study utilises LoTQA in Table Question Answering (TableQA) tasks. Performance evaluation results on the Qwen3:4B model achieved the highest scores for SacreBLEU, ROUGE-1, ROUGE-2, ROUGE-L, and METEOR. Overall, the measurements show that LoTQA's performance on the TableQA task is quite good compared to the methods used in other studies, despite low resources. These results indicate that the proposed method performs quite well on the TableQA task, despite low resources.

REFERENCES

[1] N. Jin, J. Siebert, D. Li, dan Q. Chen, Ed., "A Survey on Table Question Answering: Recent Advances," dalam *Knowledge Graph and Semantic Computing: Knowledge Graph Empowers the Digital Economy*, vol. 1669, dalam Communications in Computer and Information Science, vol. 1669. , Singapore: Springer Nature Singapore, 2022. doi: 10.1007/978-981-19-7596-7.

[2] Y. Tao *dkk.*, "KFEX-N: A Table-Text Data Question-Answering Model Based on Knowledge-Fusion Encoder and EX-N Tree Decoder," *Neurocomputing*, vol. 593, hlm. 127795, 2024, doi: https://doi.org/10.1016/j.neucom.2024.127795.

[3] L. Nan *dkk.*, "FeTaQA: Free-Form Table Question Answering," *Trans. Assoc. Comput. Linguist.*, vol. 10, hlm. 35–49, Jan 2022, doi: 10.1162/tacl_a_00446.

[4] B. Xiao, B. Kantarci, J. Kang, D. Niyato, dan M. Guizani, "Efficient Prompting for LLM-Based Generative Internet of Things," *IEEE Internet Things J.*, hlm. 1–1, 2024, doi: 10.1109/JIOT.2024.3470210.

[5] B. Zhou *dkk.*, "SynTab-LLaVA: Enhancing Multimodal Table Understanding with Decoupled Synthesis," *2025 IEEECVF Conf. Comput. Vis. Pattern Recognit. CVPR*, hlm. 24796–24806, Agu 2025, doi: 10.1109/CVPR52734.2025.02309.

[6] D. Ji *dkk.*, "Tree-of-Table: Unleashing the Power of LLMs for Enhanced Large-Scale Table Understanding," 13 November 2024, *arXiv*: arXiv:2411.08516. doi: 10.48550/arXiv.2411.08516.

[7] Z. Wang *dkk.*, "Chain-of-Table: Evolving Tables in the Reasoning Chain for Table Understanding," 19 Januari 2024, *arXiv*: arXiv:2401.04398. doi: 10.48550/arXiv.2401.04398.

[8] Z. Yang, S. Wang, Y. Yan, P. Liu, dan D. Yu, "Enhancing Free-Form Table Question Answering Models by Distilling Relevant-Cell-Based Rationales," dalam *Chinese Computational Linguistics*, vol. 14761, M. Sun, J. Liang, X. Han, Z. Liu, Y. He, G. Rao, Y. Chen, dan Z. Tian, Ed., dalam Lecture Notes in Computer Science, vol. 14761. , Singapore: Springer Nature Singapore, 2024, hlm. 3–18. doi: 10.1007/978-981-97-8367-0_1.

[9] Y. Zhang, J. Henkel, A. Floratou, J. Cahoon, S. Deep, dan J. M. Patel, "ReAcTable: Enhancing ReAct for Table Question Answering," *Proc. VLDB Endow.*, vol. 17, no. 8, hlm. 1981–1994, Apr 2024, doi: 10.14778/3659437.3659452.

[10] Y. Ye, B. Hui, M. Yang, B. Li, F. Huang, dan Y. Li, "Large Language Models Are Versatile Decomposers: Decompose Evidence and Questions for Table-Based Reasoning," dalam *SIGIR '23: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, Jul 2023, hlm. 174–184. doi: 10.1145/3539618.3591708.

[11] J. Zhao *dkk.*, "TQAgent: Enhancing Table-Based Question Answering with Knowledge Graphs and Tree-Structured Reasoning," *Appl. Sci.*, vol. 15, no. 7, hlm. 3788, Mar 2025, doi: 10.3390/app15073788.

[12] F. Lei *dkk.*, "Reasoning-Table: Exploring Reinforcement Learning for Table Reasoning," 2 Juni 2025, *arXiv*: arXiv:2506.01710. doi: 10.48550/arXiv.2506.01710.

[13] S. Duong *dkk.*, "SCOPE: A Self-Supervised Framework for Improving Faithfulness in Conditional Text Generation," 19 Februari 2025, *arXiv*: arXiv:2502.13674. doi: 10.48550/arXiv.2502.13674.

[14] S. Patnaik, H. Changwal, M. Aggarwal, S. Bhatia, Y. Kumar, dan B. Krishnamurthy, "CABINET: Content Relevance Based Noise Reduction for Table Question Answering," 13 Februari 2024, *arXiv*: arXiv:2402.01155. doi: 10.48550/arXiv.2402.01155.

[15] T. Zhang, X. Yue, Y. Li, dan H. Sun, "TableLlama: Towards Open Large Generalist Models for Tables," dalam *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Mexico City, Mexico: Association for Computational Linguistics, 2024, hlm. 6024–6044. doi: 10.18653/v1/2024.naacl-long.335.

[16] N. Deng dan R. Mihalcea, "Rethinking Table Instruction Tuning," 1 Agustus 2025, *arXiv*: arXiv:2501.14693. doi: 10.48550/arXiv.2501.14693.

[17] J. Bian *dkk.*, "HeLM: Highlighted Evidence Augmented Language Model for Enhanced Table-to-Text Generation," 27 April 2024, *arXiv*: arXiv:2311.08896. doi: 10.48550/arXiv.2311.08896.

[18] P. Shi *dkk.*, "Generation-Focused Table-Based Intermediate Pre-Training for Free-Form Question Answering," *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 10, hlm. 11312–11320, Jun 2022, doi: 10.1609/aaai.v36i10.21382.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

[19] T. Xie *dkk.*, "UnifiedSKG: Unifying and Multi-Tasking Structured Knowledge Grounding with Text-to-Text Language Models," 18 Oktober 2022, *arXiv*: arXiv:2201.05966. doi: 10.48550/arXiv.2201.05966.

[20] H. Sholehrasa, S. S. Norouzi, P. Hitzler, dan M. Jaberi-Douraki, "Knowledge in Triples for LLMs: Enhancing Table QA Accuracy with Semantic Extraction," Okt 2024, doi: 10.48550/arXiv.2409.14192.

[21] S. Ye *dkk.*, "When TableQA Meets Noise: A Dual Denoising Framework for Complex Questions and Large-Scale Tables," 22 September 2025, *arXiv*: arXiv:2509.17680. doi: 10.48550/arXiv.2509.17680.

[22] L. Cao dan H. Liu, "TableMaster: A Recipe to Advance Table Understanding with Language Models," 2 Mei 2025, *arXiv*: arXiv:2501.19378. doi: 10.48550/arXiv.2501.19378.

[23] Y. Zhao, H. Zhang, S. Si, L. Nan, X. Tang, dan A. Cohan, "Investigating Table-to-Text Generation Capabilities of LLMs in Real-World Information Seeking Scenarios," *Proc. 2023 Conf. Empir. Methods Nat. Lang. Process. Ind. Track*, hlm. 160–175, Des 2023, doi: 10.18653/v1/2023.emnlp-industry.17.

[24] M. O. Hegazi, Y. Al-Dossari, A. Al-Yahy, A. Al-Sumari, dan A. Hilal, "Preprocessing Arabic Text on Social Media," *Heliyon*, vol. 7, no. 2, hlm. e06191, Feb 2021, doi: 10.1016/j.heliyon.2021.e06191.

[25] L. Nan *dkk.*, "Yale-LILY / FeTaQA." 28 Januari 2022. doi: 10.1162/tacl_a_00446.

[26] Z. Lin *dkk.*, "Medical Visual Question Answering: A Survey," *Artif. Intell. Med.*, vol. 143, hlm. 102611, Sep 2023, doi: 10.1016/j.artmed.2023.102611.

[27] L. Wang *dkk.*, "Parameter-Efficient Fine-Tuning in Large Language Models: A Survey of Methodologies," *Artif. Intell. Rev.*, vol. 58, no. 8, hlm. 227, Mei 2025, doi: 10.1007/s10462-025-11236-4.

[28] M. Zhu dan P. H. Nguyen, "A Survey of LoRA Algorithm Variations for Language Models," dalam *Natural Language Processing and Information Systems*, R. Ichise, Ed., Cham: Springer Nature Switzerland, Jul 2025, hlm. 275–290.

[29] Y. Mao *dkk.*, "A Survey on LoRA of Large Language Models," *Front. Comput. Sci.*, vol. 19, no. 7, hlm. 197605, Jul 2025, doi: 10.1007/s11704-024-40663-9.

[30] A. Farea, Z. Yang, K. Duong, N. Perera, dan F. Emmert-Streib, "Evaluation of Question Answering Systems: Complexity of Judging a Natural Language," *ACM Comput. Surv.*, vol. 58, no. 1, hlm. 1–43, Agu 2025, doi: 10.1145/3744663.

[31] A. Moreno-Cediel, J.-A. del-Hoyo-Gabaldon, E. Garcia-Lopez, A. Garcia-Cabot, dan D. de-Fitero-Dominguez, "Evaluating the Performance of Multilingual Models in Answer Extraction and Question Generation," *Sci. Rep.*, vol. 14, no. 1, hlm. 15477, Jul 2024, doi: 10.1038/s41598-024-66472-5.

[32] N. M. Guerreiro, R. Rei, D. V. Stigt, L. Coheur, P. Colombo, dan A. F. T. Martins, "Xcomet: Transparent Machine Translation Evaluation Through Fine-Grained Error Detection," *Trans. Assoc. Comput. Linguist.*, vol. 12, hlm. 979–995, Sep 2024, doi: 10.1162/tacl_a_00683.

[33] M. Toshpulatov, W. Lee, J. Jun, dan S. Lee, "Deep Learning Pathways for Automatic Sign Language Processing," *Pattern Recognit.*, vol. 164, hlm. 111475, Agu 2025, doi: 10.1016/j.patcog.2025.111475.

[34] G. Dugac dan T. Altwicker, "Classifying Legal Interpretations Using Large Language Models," *Artif. Intell. Law*, Apr 2025, doi: 10.1007/s10506-025-09447-9.

[35] H. Sun dan B. Kong, "Sustainable Improvement and Application of Multilingual English Translation Quality Using T5 and MAML," *Discov. Artif. Intell.*, vol. 4, no. 1, hlm. 98, Des 2024, doi: 10.1007/s44163-024-00213-5.

[36] Y.-H. Chen, E. J.-L. Lu, dan K.-H. Cheng, "Enhancing SPARQL Query Generation for Question Answering with a Hybrid Encoder–Decoder and Cross-Attention Model," *J. Web Semant.*, vol. 87, hlm. 100869, Des 2025, doi: 10.1016/j.websem.2025.100869.

[37] D. Suhartono, M. R. N. Majiid, dan R. Fredyan, "Towards Automatic Question Generation Using Pre-Trained Model in Academic Field for Bahasa Indonesia," *Educ. Inf. Technol.*, vol. 29, no. 16, hlm. 21295–21330, Nov 2024, doi: 10.1007/s10639-024-12717-9.

[38] D. Rathod, A. K. Yadav, M. Kumar, dan D. Yadav, "Character-Level Encoding Based Neural Machine Translation for Hindi Language," *Neural Process. Lett.*, vol. 57, no. 2, hlm. 23, Feb 2025, doi: 10.1007/s11063-025-11718-0.

[39] X. Chen, Z. Chen, dan S. Cheng, "CoTHSSum: Structured Long-Document Summarization Via Chain-of-Thought Reasoning and Hierarchical Segmentation," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 37, no. 4, hlm. 40, Mei 2025, doi: 10.1007/s44443-025-00041-2.

[40] A. Auriemma Citarella, M. Barbella, M. G. Ciobanu, F. De Marco, L. Di Biasi, dan G. Tortora, "Assessing the Effectiveness of ROUGE as Unbiased Metric in Extractive Vs. Abstractive Summarization Techniques," *J. Comput. Sci.*, vol. 87, hlm. 102571, Mar 2025, doi: 10.1016/j.jocs.2025.102571.

[41] K. M. Rani Krishna, K. Somasundaram, P. Arulmozhivarman, S. A. Immanuel, dan E. R. Rajkumar, "Deep Learning for Text Summarization Using NLP for Automated News Digest," *Sci. Rep.*, vol. 15, no. 1, hlm. 36343, Okt 2025, doi: 10.1038/s41598-025-20224-1.

[42] H. Huang *dkk.*, "GPT-Based Lifelong Learning and ANFIS-Driven Reply Memory Ratio Prediction for Aspect-Based Sentiment Analysis," *Complex Intell. Syst.*, vol. 11, no. 11, hlm. 463, Okt 2025, doi: 10.1007/s40747-025-02086-2.

[43] H. Zhang, P. S. Yu, dan J. Zhang, "A Systematic Survey of Text Summarization: From Statistical Methods to Large Language Models," *ACM Comput. Surv.*, vol. 57, no. 11, hlm. 1–41, Jun 2025, doi: 10.1145/3731445.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

[44] Y. Jin, Q. Shi, dan Q. Liu, "CFAS: Consensus-Focused Abstractive Meeting Summarization Through Multi-Party Discourse Modeling," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 37, no. 7, hlm. 187, Agu 2025, doi: 10.1007/s44443-025-00210-3.

[45] W. Xu, C. Huang, S. Gao, dan S. Shang, "LLM-Based Agents for Tool Learning: A Survey," *Data Sci. Eng.*, Jun 2025, doi: 10.1007/s41019-025-00296-9.

[46] R. Ilyas, M. Khodra, R. Munir, R. Mandala, dan D. Widyantoro, "Generating Paraphrase Using Simulated Annealing for Citation Sentences," *Informatics*, vol. 10, no. 2, hlm. 34, Mar 2023, doi: 10.3390/informatics10020034.

[47] M. Y. Mohammed, S. A. Ali, S. K. Ali, A. A. Majeed, dan E. H. Mohamed, "Aftina: Enhancing Stability and Preventing Hallucination in AI-Based Islamic Fatwa Generation Using LLMs and RAG," *Neural Comput. Appl.*, vol. 37, no. 25, hlm. 20957–20982, Sep 2025, doi: 10.1007/s00521-025-11229-y.