

Article

Cross-Lingual Adaptation for Multilingual Table Question Answering and Comparative Evaluation with Large Language Models [†]

Sanghyun Cho ¹, Minho Kim ^{2,*}, Hye-Lynn Kim ³, Jung-Hun Lee ⁴, Hyuk-Chul Kwon ⁵ and Soo-Jong Lim ¹

¹ Electronics and Telecommunications Research Institute, Daejeon 34129, Republic of Korea; shcho.nlp@etri.re.kr (S.C.); isj@etri.re.kr (S.-J.L.)

² Division of Artificial Intelligence, Korea Maritime and Ocean University (KMOU), Busan 49112, Republic of Korea

³ HDC Labs, Seoul 06722, Republic of Korea; helen6339@hdc-labs.com

⁴ Department of Data Informatics, Korea Maritime and Ocean University (KMOU), Busan 49112, Republic of Korea; it_leejh@kmou.ac.kr

⁵ Department of Electrical and Computer Engineering, Pusan National University, Busan 46241, Republic of Korea; hckwon@pusan.ac.kr

* Correspondence: minho@kmou.ac.kr

[†] This paper is an extended version of our paper presented at the 2025 IEEE International Conference on Big Data and Smart Computing (BigComp), Kota Kinabalu, Malaysia, 9–12 February 2025.

Abstract

Table question answering has been studied using datasets drawn from a variety of tabular sources and task formats. However, most publicly available resources have been created in high-resource languages such as English. For low-resource languages, researchers are often required to construct new datasets or translate existing ones, which incurs substantial time, effort, and financial cost. In contrast to natural language text, table data consists of structured entries whose interpretation is less affected by language-specific syntax or word order. In this work, we present a cost-effective strategy for multilingual table QA that relies on selectively translating only the questions of existing datasets. Leveraging the language-agnostic structure of tables, our approach maintains the original table content while translating queries into multiple target languages. To address possible performance drops caused by using table data in the source language rather than the target language, we apply cross-lingual adaptation techniques using contrastive learning and adversarial training. In addition, to strengthen reasoning ability while avoiding degradation in languages not seen during pre-training, we perform supplementary pre-training of a RoBERTa-based multilingual encoder with SQL-derived table data. Finally, we extend our investigation beyond encoder-based architectures and evaluate decoder-only large language models under the same multilingual table QA setting. The experiments show that LLaMA-3 models exhibit strong cross-lingual generalization even without using translated table context and often achieve competitive performance using only Korean table data. Moreover, the performance gap among training configurations such as translated queries or translated datasets is notably smaller compared to encoder-based models, highlighting the inherent multilingual robustness of modern LLMs. We further evaluate LLaMA-3 models on domain-specific table datasets and observe that domain knowledge acquired from Korean tables transfers effectively across languages even without multilingual supervision, underscoring the potential of LLMs for specialized multilingual table reasoning. These findings demonstrate that LLMs can serve as an effective alternative for multilingual table QA, particularly in low-resource or partially translated environments.



Academic Editor: Fernando Bobillo

Received: 7 December 2025

Revised: 4 January 2026

Accepted: 13 January 2026

Published: 1 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: table question answering; cross-lingual adaptation; pretrained language models; large language models

1. Introduction

As digital information continues to grow across different platforms, the need for clear and structured representations of data has become increasingly important. Tables have emerged as one of the most practical formats for organizing information because they present complex content in a compact and accessible way. They are now widely used in scientific data collections, public administrative documents, business reports, and many other fields where large amounts of information must be interpreted quickly and accurately. As a result, the ability to understand and extract information from tables has become essential for effective data analysis and reliable decision making.

Although interest in table-based analysis has increased, the development of table question answering resources has not progressed at the same pace. Many machine reading comprehension datasets exist in various languages, but table question answering (QA) datasets are available only for a limited number of languages. For instance, the TyDi QA benchmark [1] covers 11 typologically diverse languages, whereas well-known benchmarks often exist only in English [2–5].

Creating new resources requires substantial human effort, and translating existing datasets introduces additional challenges. Prior work on multilingual tabular benchmarks typically relies on qualified bilingual translators and extensive post-editing/validation, which makes data creation costly and time-consuming [6]. Human translation demands bilingual experts, which increases cost. While automatic translation is cheaper, machine-translated questions can introduce translation mistakes and inconsistencies, often requiring careful correction or post-editing to ensure faithful grounding [7,8]. In severe cases, entire sections of the dataset must be corrected or reannotated. These limitations make it difficult for researchers to build multilingual systems for table understanding.

At the same time, tables possess characteristics that make them naturally suitable for multilingual learning. A table is composed of distinct cells, each containing a small unit of information. Because the structure remains constant across languages, understanding the content of a table does not depend heavily on grammar, word order, or language-specific patterns. This structural stability allows models to handle the same table in several languages with minimal modification. It also suggests a promising direction for reducing the cost of multilingual dataset construction by keeping the table in its original form while translating only the questions. This property also allows existing table QA annotations to be reused across languages without reconstruction. Since answer labels are grounded in table cells rather than surface word order, machine translation can be applied without breaking the alignment between questions and annotated answers.

In this work, we take advantage of these properties to propose a cost-effective approach for multilingual table question answering. Instead of translating full datasets, we keep the table context in its original language and translate only the questions into each target language. This selective translation significantly reduces data creation costs but may also cause performance drops because the model receives questions and tables written in different languages. To address this issue, we apply contrastive learning and adversarial training so that the model can better align representations across languages and learn to generalize even when the table remains untranslated. We also study how table-specific pre-training methods influence multilingual performance using both masked language modeling objectives and SQL-based training resources.

Our experiments cover a diverse set of languages including Korean, Thai, Vietnamese, Arabic, Japanese, and Indonesian. These languages differ in word order, writing systems, and morphological patterns, which allows us to observe how multilingual table question answering models behave across a wide range of linguistic settings.

Beyond encoder-based models, we further investigate whether recent decoder-only large language models can perform multilingual table reasoning under the same constraints. Modern large language models show strong cross-lingual generalization and powerful reasoning abilities, but it is still unclear how well these abilities extend to structured tabular data. To explore this question, we evaluate several LLaMA 3 models [9] with different parameter sizes and training configurations. We analyze how translated questions, translated tables, and domain-specific characteristics influence model behavior. We also examine whether domain knowledge learned from Korean table data can transfer to other languages in specialized fields such as public administration, legal documents, and product specifications.

By combining encoder-based and decoder-based methods within a single experimental framework, this study provides a comprehensive view of multilingual table question answering. It highlights the strengths and limitations of each modeling approach and examines whether the challenges identified in encoder-based settings, particularly the dependence on translated table context, also appear in modern large language models.

2. Related Work

2.1. Domain Adaptation

Domain adaptation refers to a collection of techniques aimed at learning representations that remain effective across different domains, allowing models to perform well even when the target domain has limited or no labeled data. These approaches have shown strong effectiveness in various tasks, including machine reading comprehension. Ganin et al. [10] introduced one of the early adversarial frameworks for domain adaptation. Their method incorporates both a task classifier and a domain classifier and employs a gradient reversal layer to encourage the learning of domain-invariant representations. Building on this idea, Lee et al. [11] applied adversarial learning to machine reading comprehension, demonstrating that it helps models generalize across datasets drawn from different domains. In their setup, the domain discriminator receives the [CLS] token representation from BERT as input, while the reader model is optimized to minimize the Kullback–Leibler divergence between the discriminator’s output distribution and a uniform prior. Wang et al. [12] proposed a complementary adaptation strategy that creates pseudo questions by using unlabeled passages in the target domain. Their approach updates the encoder parameters to maximize the loss of the domain classifier, thereby encouraging the model to acquire domain-invariant features. Bai et al. [13] introduced DAMT², a domain-adaptive multi-task transformer designed for low-resource machine reading comprehension. Their architecture separates learning into two components: a task-aware transformer layer trained on the source MRC task and a domain-aware layer optimized on an unsupervised objective using target-domain data. During fine-tuning with a small amount of labeled target-domain MRC data, the domain-aware layer is removed, and training continues only on the task-aware layer.

2.2. Contrastive Learning Approaches

Contrastive learning trains a model to pull the representations of positive sample pairs closer together while pushing apart the representations of negative pairs. Contrastive learning has shown remarkable success in visual representation learning [14], particularly when combined with data augmentation techniques. Kim et al. [15] introduced a self-

supervised contrastive regularization approach aimed at improving generalization across varying domain distributions. Contrastive learning has been employed in a wide range of tasks and modalities to achieve representation alignment. For example, VLPT [16] pre-trains a vision–language model through cross-modal alignment, encouraging high mutual information between paired images and textual descriptions. Within natural language processing, contrastive techniques also have been applied to multiple objectives, including cross-lingual alignment during model pre-training [17], the construction of sentence embeddings [18], and entity typing tasks [19].

2.3. Table Question Answering

Pre-training for table question answering has utilized a wide range of data sources, including Wikipedia tables, spreadsheet files, synthetically generated queries, and SQL-based datasets. Models such as TAPAS and TaBERT [20] are trained on tabular corpora using a masked language modeling objective, where the model learns to recover the original content of masked table cells. TUTA [21] introduced a pre-training approach that incorporates tree-structured attention mechanisms and positional embeddings to better capture the spatial layout and hierarchical relationships within tables. To enhance numerical reasoning capabilities, ForTAP incorporates spreadsheet data during pre-training. During this process, it is trained with two objectives: (1) predicting the appropriate function or operand for a given cell, and (2) identifying which cells are referenced by a formula cell. TAPEX [22] presents a table-focused pre-training framework in which the model implicitly acquires SQL execution abilities by taking synthetic SQL queries together with table data as input and learning to generate the corresponding execution outputs.

In this study, we pre-trained a multilingual table question answering (TQA) model using a masked language modeling objective on multilingual Wikipedia tables, following the strategy employed in TAPAS [23]. We also incorporated the table-based pre-training data provided in TAPEX, enabling our multilingual model to adopt a similar pre-training scheme to TAPEX. To investigate how such supplementary pre-training influences multilingual QA performance, we further performed experiments using the K-adapter [24], in which the encoder’s original parameters remain fixed while only the adapter layers are updated.

2.4. Large Language Models

In this work, we employ two categories of foundation models, Gemma-3 [25] and LLaMA-3 [26], to investigate how recent decoder-based large language models behave in multilingual table question answering. Gemma-3 is used to examine the effect of replacing commercial translation models with open source models when translating table data into multiple target languages. LLaMA-3 is used to analyze the performance of decoder-based language models in multilingual table question answering, a task that has been mostly studied with encoder-based architectures in previous research.

Gemma-3 is an improved successor to Gemma-2 [27] and maintains a similar overall transformer architecture [28] while introducing important enhancements for long context processing and multilingual capability. The difference lies in the ratio of the two attention types: while Gemma-2 alternates them in a one to one pattern, Gemma-3 adopts a structure in which multiple local attention layers are followed by a single global attention layer. This modification improves memory efficiency for longer context processing while retaining the general characteristics of the original design. Compared with Gemma-2, the pre-training corpus of Gemma-3 contains a much larger proportion of multilingual text, which strengthens its ability to generalize across languages. Gemma 3 is released in four model sizes: 1B, 4B, 12B, and 27B parameters. In this study, we use the 12B and 27B variants to

translate table data and question sets into several target languages for the construction of our multilingual table question answering datasets.

LLaMA-3 is the successor to LLaMA-2 [9] and incorporates several architectural improvements for more efficient and accurate text generation. The model adopts grouped query attention (GQA) [29], which improves inference efficiency while maintaining the quality of multi-head attention. It also employs an enhanced positional embedding mechanism based on rotary position encoding, which stabilizes reasoning over longer contexts. LLaMA-3 expands its vocabulary to 128K tokens, adding a substantial number of non-English tokens in order to improve subword compression and multilingual representation. Its pre-training corpus also contains a larger share of multilingual text, which supports stronger cross-lingual generalization. To evaluate the behavior of decoder-based language models in multilingual table question answering, we experiment with three LLaMA-3 variants: LLaMA 3.1-8B, LLaMA-3.2-1B and LLaMA-3.2-3B. These models allow us to analyze performance differences across model scales and to examine how decoder-based architectures process structured tabular information and multilingual reasoning.

Although most existing large language models are predominantly trained on English-centric corpora, several recent efforts have focused on building high-quality foundation models for non-English languages. For example, the Polyglot-Ko [30] introduces a series of Korean-centric LLMs trained on a large-scale, carefully curated 1.2 TB Korean corpus, demonstrating that monolingual models can outperform multilingual counterparts on Korean-specific tasks by leveraging linguistically aligned tokenization and domain-appropriate data sources.

Domain-Specific Table QA Datasets

Real-world documents frequently contain tables across diverse domains, including law, public administration, and product specifications. Prior work has shown that such tables exhibit substantial variation in structure and terminology, often containing legal expressions, administrative terminology, or technical product specifications that do not appear in general-purpose datasets [31]. This diversity makes domain-specific table QA inherently more challenging than QA over general web documents.

To support research on domain robustness, earlier work constructed domain-oriented table QA datasets by collecting tables from the National Law Information Center, public data sources, and product comparison platforms [31]. These datasets highlight that domain-specific tables differ not only in vocabulary but also in layout and required knowledge. In this work, we further incorporate these domain-specific datasets into our LLM-based experiments to examine whether the domain knowledge learned from Korean source data can effectively transfer to other target languages.

3. Multilingual Question Answering Dataset Construction

3.1. Dataset Construction Using Commercial LLMs

In this section, we describe how we constructed our multilingual Table QA dataset for training and evaluation. We explain (i) how we selected source tables, (ii) how we translated and curated multilingual questions, and (iii) how we ensured translation quality and diversity. Figure 1 summarizes the overall workflow.

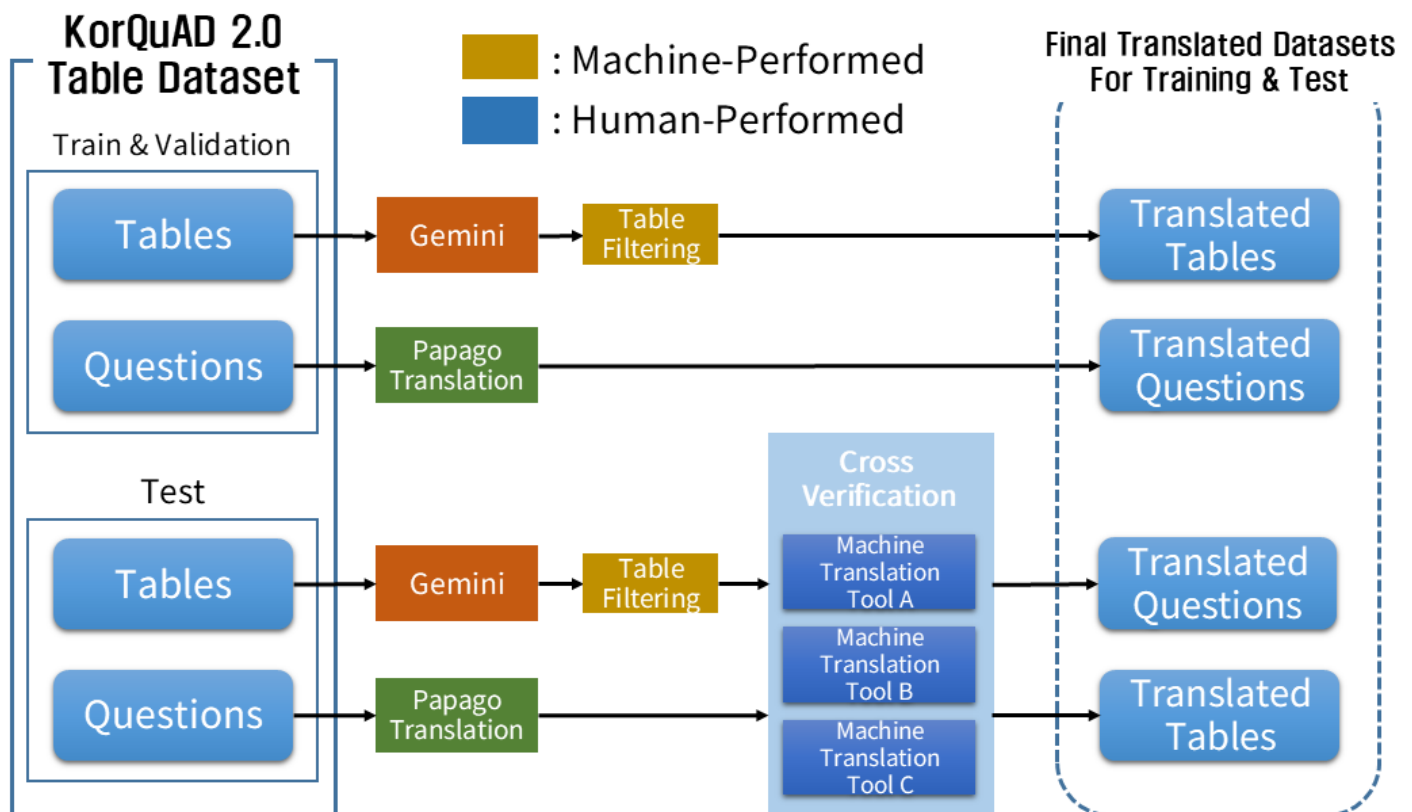


Figure 1. Overview of multilingual TQA dataset construction process [32].

In this study, we adopt the KorQuAD 2.0 dataset as the primary resource to construct our Table QA dataset. Although several TQA benchmarks exist, including WikiTQ, WikiSQ, and SQL-based datasets, we chose a Korean dataset for validating the translated data because the researcher involved in this work is a native Korean speaker. KorQuAD 2.0 [33] was constructed by annotating answers in Korean Wikipedia articles, and from this corpus, we extracted only the examples in which the annotated answers appear within tables. Additionally, we employed different machine translation systems (<https://papago.naver.com/>) (<https://www.deepl.com/ko/translator>) (accessed on 1 December 2025) depending on how each portion of the dataset was utilized, such as when it served as training data, evaluation data, question text, or contextual information.

Translating table data into multiple languages is computationally expensive and often introduces structural inconsistencies. To mitigate this issue, we adopt a two-step translation strategy. We first translate tables into English and then translate the English tables into the remaining languages.

During this process, we frequently encountered errors where the delimiters that define table rows and columns were incorrectly produced by the translation model. These errors often resulted in translated tables that preserved most of the textual content but differed structurally from the original tables. A typical failure case involves the omission of a cell in a specific row, leading to inconsistent column counts across rows. For example, while translating a table listing episode numbers, subtitles, and air dates, the translated table may correctly generate the episode and subtitle fields but omit the air date for a single row, thereby breaking row and column alignment.

To remove such faulty translations, we implemented a filtering procedure that verifies whether the translated tables preserve the original row and column counts and checks that non-translatable elements, such as dates, times, and numerical values, remain unchanged and appear in their correct positions. Tables that fail any of these structural consistency

checks are excluded from further processing to ensure the integrity of table-based annotations.

For question translation, we adopted more specialized machine translation tools because the volume of question text is much smaller than that of tables. To prevent translation errors from affecting the evaluation set, we added a quality assurance step that involved back-translating the output into Korean and reviewing it with human annotators. Furthermore, because different translation systems often produce outputs with varying linguistic styles and semantic nuances, we used multiple tools and cross-validated their results to reduce tool-specific bias.

The translated table data in the training set was not used in our primary experiments. Instead, it served as part of an ablation study to analyze the effect of including translated table context in multilingual question answering. The validation set was constructed by partitioning the translated portion of the training data.

3.2. Dataset Construction Using Open LLMs

Recent studies [34,35] have increasingly adopted open source large language models for data creation, augmentation, and multilingual corpus expansion. Following this trend, we employed open LLMs to translate the original Korean table-based question answering data into multiple target languages.

Our goal in applying open LLMs was twofold. First, we aimed to evaluate whether multilingual training data generated by open models could serve as an effective alternative to data translated by commercial LLMs. Second, we assessed whether the resulting translated datasets were sufficient to support downstream multilingual table question answering without a degradation in model performance. For this purpose, we used two instruction-tuned variants of Gemma-3, specifically Gemma-3-12b-it and Gemma-3-27b-it, to translate the table context and question data into six target languages.

Before translating into the final target languages, the original Korean table data were first translated into English, and the resulting English tables were subsequently translated into each target language. This intermediate English translation step was applied to prevent structural corruption and to improve translation quality for languages where direct Korean-to-target translation caused degradation.

After the translation process, the translated table data were processed using the same filtering procedure described in Section 3.1. This ensured that structural consistency, such as the number of rows and columns, and the validity of non-translatable cell contents (e.g., dates, times, and numerical values) were preserved.

To further guarantee that table structure is strictly preserved during translation, we used the following prompt when applying open LLMs:

You are an expert translator for English → {target_lang} table data.

The following table is in Markdown format, and each row is separated by the “|” character.

Your task is to translate only the English text inside the table into natural and concise {target_lang}, without changing the table structure.

Rules:

1. Do NOT change the number or positions of “|” in any line.
- Do NOT add or remove any cells (columns).
- Do NOT add or delete any rows (lines).
2. Keep numbers (e.g., 2009, 2011) and reference markers (e.g., [11,12]) exactly as they are.
3. Do NOT output any explanation, comments, or extra text outside the table.

Output only the translated table.

- Input Table -
{input_table}

- Translated Table - {translated_table}

Unlike the setting with commercial LLMs, where only the questions were translated while the original Korean tables were retained due to the high cost of translating large amounts of tabular data, the use of open LLMs allowed us to translate the entire table context as well. Since open LLMs can be executed with high throughput using inference frameworks such as VLLM, the cost of generating multilingual table contexts becomes substantially lower, enabling full translation of all training tables rather than question only translation.

3.3. Domain-Specific Table Question Answering

Compared to the KorQuAD-based setting, constructing multilingual training data for the DomainTQA benchmark presents additional challenges. The dataset includes highly specialized domains such as law and IT products, where tables contain domain-specific terminology and technical expressions. In our preliminary analysis, we observed that both open LLMs and cost-efficient commercial LLMs often struggled to produce reliable translations for these domains. The resulting outputs frequently lacked terminological accuracy or failed to preserve subtle domain semantics, making large-scale multilingual translation substantially more costly and less effective than in general-purpose settings.

For this reason, translating the entire DomainTQA training set into multiple languages, as we did for KorQuAD-based TQA, was not considered practical from both cost and quality perspectives. Instead, we used only the original Korean tables for training. Large language models exhibit strong cross-lingual generalization even without multilingual supervision, and we empirically found that LLMs were considerably more robust than encoder-based models in transferring their reasoning ability from Korean to other target languages. Based on these observations, we incorporated DomainTQA dataset [31] as an additional evaluation setting in our LLM-based experiments to examine whether LLMs can generalize to domain-specialized table reasoning tasks under limited multilingual supervision.

In the LLM-based experiments, we retained the original Korean training data without performing multilingual translation for the DomainTQA benchmark. For evaluation, we translated the test sets into the target languages using GPT-4o. All translated outputs were subsequently verified through the same validation pipeline used for the KorQuAD-based dataset, including checks for table structure preservation, consistency of non-translatable elements, and semantic fidelity to the source Korean content.

In terms of data scale, the DomainTQA benchmark covers three specialized domains: Public Administrative Document Tables, Legal Document Tables, and Product Specification Tables. The training set contains 1214 instances for Product Specification Tables, 923 for Legal Document Tables, and 6193 for Public Administrative Document Tables. The corresponding test sets consist of 814, 647, and 1303 instances for the same domains, respectively.

4. Multilingual Table Question Answering

In this section, we present the contrastive learning and adversarial training strategies used to enable our TQA model to generalize across multiple languages. We also outline our analysis of how additional table-based pre-training influences the multilingual performance of TQA models, supported by further experiments incorporating table pre-training objectives.

The proposed framework consists of three components that address different aspects of cross-lingual table question answering. First, table-based pre-training aims to adapt the encoder to the structural properties of tables, such as row-column alignment and cell-level semantics, providing a strong initialization for table understanding. Second, contrastive learning is introduced to encourage language-invariant representations by aligning semantically equivalent question-table pairs across languages. Finally, adversarial training is applied to further reduce language-specific signals in intermediate representations, promoting robustness to cross-lingual variation. These components are designed to be complementary, but their interaction depends on the strength of structural priors learned during pre-training.

Figure 2 presents the overall architecture of the multilingual TQA framework. To simulate a realistic low-resource setting in which translating and re-annotating the full dataset is impractical, our experiments adopt a selective question translation approach. In this setup, the table data in the training set remains in its original language, and only the question texts are translated into the target languages.

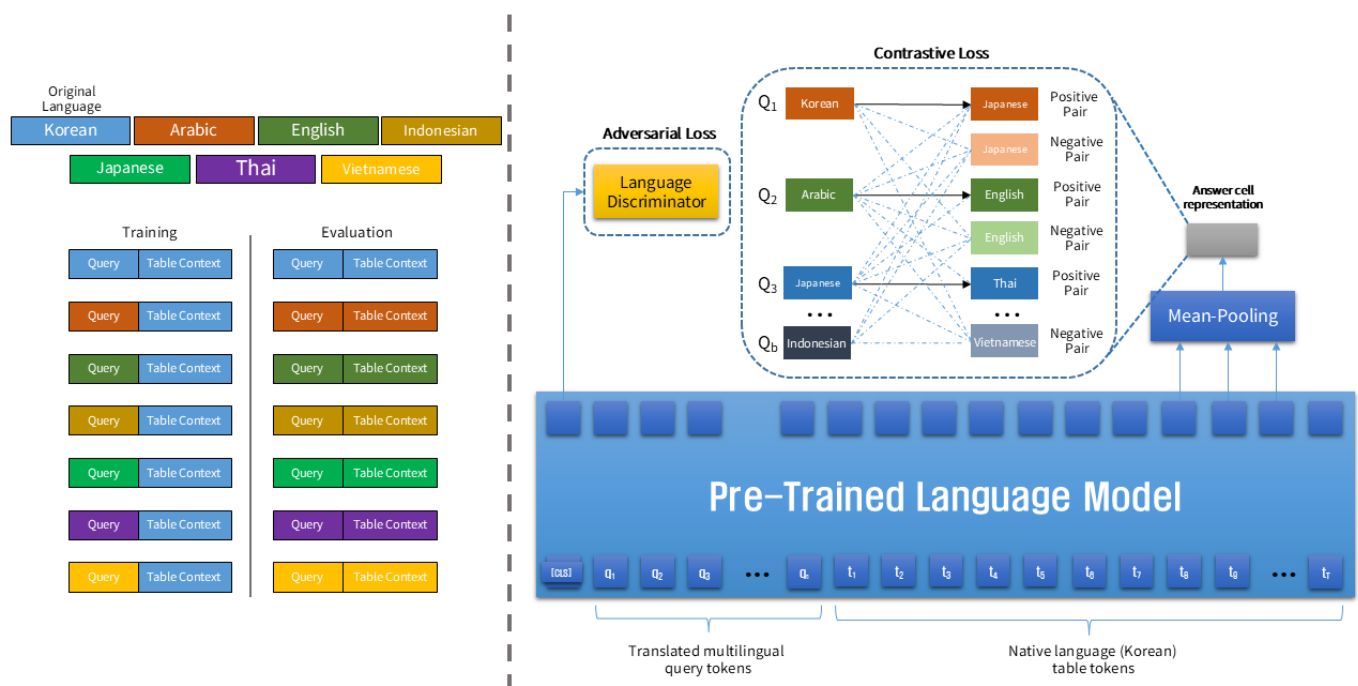


Figure 2. Overall architecture and cross-lingual adaptation methods for multilingual TQA [32]. The left side of the dashed line indicates the data input/processing flow, while the right side indicates the model architecture.

The dataset employed in our experiments, KorQuAD 2.0 [33], is a machine reading comprehension benchmark. Within table contexts, the annotated answers correspond either to an entire cell's content or to a substring within a cell. The QA model is trained to identify the start and end indices of the answer span in the input context. The loss function for the model is defined in Equation (1).

$$L_{QA} = -\frac{1}{N} \sum_i^N \log P(y_i^s | p_i^s) + \log P(y_i^e | p_i^e) \quad (1)$$

Here, N denotes the number of training instances. For each instance i , y_i^s and y_i^e represent the ground-truth start and end indices of the answer span within the input context, respectively. The terms p_i^s and p_i^e denote the predicted probability distributions

over all token positions for the start and end indices produced by the model. The loss function minimizes the negative log-likelihood of the correct start and end positions, following the standard span-based question answering formulation.

4.1. Contrastive Learning

Contrastive learning is introduced to explicitly align representations of semantically equivalent question-table pairs across languages. Unlike table-based pre-training, which focuses on structural understanding, contrastive learning directly targets cross-lingual invariance by pulling together representations from different languages that correspond to the same underlying table content.

We conducted experiments in which the multilingual TQA model was trained using datasets that included only translated question texts. To ensure that the encoder produced consistent cell representations, we incorporated contrastive learning so that queries expressing the same meaning would be mapped to similar representations, even when written in different languages.

In our contrastive learning setup, the model acquires cross-lingual representations by maximizing the inner product between the hidden states of queries that correspond to the same table entry across languages. For this purpose, we use the mean-pooled hidden states of the answer cells generated by the encoder.

To train the model, we prepare two mini-batches that contain queries for the same instances but written in different languages. We first sample a mini-batch of N instances, each of which includes two queries expressed in two randomly selected languages. Queries for the same instance are treated as positive pairs.

Inspired by the hard negative strategy introduced in [36], we also incorporate negative samples in one of the mini-batches. The first mini-batch consists of:

$$[\text{CLS}] q_{i_j} [\text{SEP}] t_i$$

The second minibatch includes:

$$[\text{CLS}] q_{i_k} [\text{SEP}] t_i$$

and the negative example:

$$[\text{CLS}] q_{k^*(i)_k} [\text{SEP}] t_i$$

Here, i indicates the data index, j and k represent language indices, and k^* denotes the index of the data instance with the highest BM25 score for the query q_{i_k} .

The contrastive loss function used in our training setup is defined in Equation (2).

$$L_{ctl} = \frac{-\log(e^{\text{sim}(h_{a_j}, h_{a_j})})}{\sum_i^N e^{\text{sim}(h_{a_j}, h_{i_k})} + \sum_i^N e^{\text{sim}(h_{a_j}, h_{k^*(i)_k})}} \quad (2)$$

Here, h_{a_j} represents the hidden vector computed from the input that contains the i th table instance paired with the query translated in the j_{language} . The function sim denotes the inner product between two representation vectors. This contrastive objective is added to the QA loss, and the model is trained to jointly minimize the combined loss.

4.2. Adversarial Training

Adversarial training complements contrastive learning by discouraging the encoder from encoding language-specific features. While contrastive learning enforces similarity between aligned examples, adversarial training operates at the distribution level, reducing language-identifiable signals in intermediate representations.

To promote cross-lingual generalization and prevent the model from overfitting to training data that pairs source-language tables with target-language queries, we incorporate adversarial training. In addition to the QA model, we introduce a language discriminator that receives the feature representation and predicts the language of the input query. For this module, we use the [CLS] token embedding as the discriminator's input.

The overall training procedure consists of two stages: updating the QA model and updating the discriminator. During the first stage, the QA component is trained both to produce correct answer spans by minimizing L_{QA} and to confuse the language discriminator by minimizing L_{adv} .

Following MRQA [11], the adversarial loss L_{adv} encourages the model to fool the language discriminator by minimizing the Kullback–Leibler divergence between the discriminator's predicted distribution and a uniform distribution. The loss used in the first stage is defined in Equations (3) and (4).

$$L_{adv} = KL(U|h_i) \quad (3)$$

$$L_{(w^a)} = L_{adv} + L_{ctl} + L_{QA} \quad (4)$$

Here, h_i refers to the [CLS] token representation of the i -th input. The term L_{ce} denotes the contrastive loss defined in Equation (3). Although training can be performed using only the required L_{adv} , the combined loss formulation shown above may also be applied.

In the second phase, the discriminator's parameters are updated so that it can accurately identify the language of the input query. The loss function for training the discriminator is defined in Equation (5).

$$L_{discriminator(w^d)} = \frac{1}{N} \sum_i^N \log P(d_i|h_i) \quad (5)$$

Here, $\log P(d^i|h_i)$ denotes the probability assigned by the discriminator to the language label of the corresponding input.

4.3. Pre-Training over Tables

Many prior studies on table question answering have demonstrated substantial performance gains through table-specific pre-training. Motivated by these findings, we also conducted pre-training for our multilingual TQA model. The pre-training procedure consisted of two strategies: the first applied masked language modeling using Wikipedia table data [23], and the second employed a cell-selection objective based on synthetic SQL data [22].

To build the corpus for the maskLM objective, we extracted table data from the Wikipedias of multiple countries, along with related textual content. When tables appeared within a paragraph, we collected the accompanying text directly. Otherwise, we paired each table with the text segment exhibiting the highest Jaccard similarity to the table. In this experiment, we did not incorporate any additional table-specific embeddings. Instead, we followed the serialization approach used in TAPEX [22], formatting each table as: “row 1: cell text 1 | cell text 2 | ... ; row N: cell text 1 | cell text 2 | ...”.

We additionally performed pre-training using a synthetic SQL corpus. While TAPEX [22] employs an encoder–decoder framework that directly produces SQL execution outputs, our experiments use an encoder-only model. Consequently, rather than generating SQL results, we reformulated the pre-training objective to predict the answer cells. From the full SQL pre-training corpus, we selected only the examples that contain a

single answer cell so that the prediction layer's parameters could be directly reused during fine-tuning. The model was trained according to the procedure described in Equation (4).

Since the SQL queries and table data used for pre-training are written in English, such additional pre-training could negatively affect the model's multilingual performance. To address this concern, we performed supplementary pre-training using an adapter-based approach. Following [37], we incorporated a table-specific adapter layer and trained only this layer using SQL-based pre-training objectives. During the pre-training stage, updates were applied solely to the adapter layer and the prediction layer, whereas in the fine-tuning stage, the entire model's parameters were optimized.

When strong structural priors are already learned through table-based pre-training, additional alignment objectives may impose redundant or conflicting constraints on the representation space. This interaction can lead to over-regularization, which helps explain the performance degradation observed in certain combinations of pre-training and adaptation strategies.

4.4. Training with Large Language Models

In addition to the encoder-based multilingual TQA model, we further apply a decoder-only LLM to investigate its ability to perform table question answering. For training the LLM, we adopt the standard CausalLM-based Supervised Fine-tuning (SFT) objective. Given an input prompt consisting of a task instruction, the translated flattened table context, and the translated question, the model is trained to autoregressively generate the gold answer. The loss is computed only on the answer tokens by masking out the prompt portion. The prompt format used for SFT is as follows:

```
You are an AI assistant that must answer strictly based on the provided
documents (Context).
```

```
Rules:
```

1. Your answer must be grounded ONLY in the given Context.
2. Do NOT infer or create information that is not explicitly stated in the documents.
3. Provide clear, concise, and accurate answers.

```
<Context>
{table_text}
</Context>
<Question>
{question}
</Question>
<Answer> {answer} </Answer>
```

5. Experiments

5.1. Experimental Setup

In this section, we describe the experimental setup used for multilingual table question answering. XLM-RoBERTa was adopted as the backbone encoder, leveraging its multilingual pre-training through cross-lingual representation learning [38] (<https://huggingface.co/FacebookAI/xlm-roberta-base> (accessed on 1 December 2025)). To construct the masked language modeling (maskLM) corpus, we collected tables and infoboxes from the Wikipedia editions of all target languages. Because English Wikipedia contains a disproportionately large volume of tabular content, only a subset of English tables was included to prevent the corpus from becoming imbalanced. The number of extracted tables and infoboxes for each language is presented in Table 1. While a larger

amount of English data could have been gathered, we limited its size to maintain a comparable scale across languages. Although English still accounts for a larger portion of the corpus, the pre-training objective primarily aims to adapt the model to table structure rather than to learn language-specific patterns.

Table 1. Amount of pretraining data per language [32].

Language	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Table	81,789	1,407,649	98,523	590,523	124,000	62,718	110,093
Infobox	319,954	2,341,176	189,858	1,113,837	205,865	75,429	197,199

For the cell selection pre-training objective, we used the SQL-based corpus released by [22]. Unlike large-scale table crawls, this dataset consists of carefully curated tables, and SQL queries were generated by sampling headers and values according to predefined templates. This construction allowed the model to learn structural reasoning signals through SQL execution supervision.

We run all experiments on two RTX-3090 GPUs (EMTEK Inc., Seoul, Republic of Korea). To fit this budget, we pre-train maskLM and cell selection on subsets of their corpora. We train maskLM for 300k steps (batch size 4) and cell selection for 720k steps (batch size 24).

For fine-tuning, we used the table-based subset of the KorQuAD 2.0 dataset [33], a Korean machine reading comprehension benchmark modeled after SQuAD [39]. KorQuAD 2.0 contains answer annotations across both unstructured passages and structured elements such as lists and tables. In this study, we extracted only the examples whose answers appear within tables to specialize the task for table question answering.

To provide a more detailed analysis of model behavior, we categorized evaluation questions into two types. Cell-matching questions require identifying a specific table cell that directly satisfies a given condition. Reasoning questions, in contrast, demand additional inference, such as identifying extremal values, comparing quantities, or interpreting row and column order. This categorization allowed us to better analyze how different model components contribute to distinct reasoning requirements.

During dataset construction, we applied strict filtering procedures to avoid training on erroneous translations. If a translation error or table misinterpretation was detected in any of the six target languages during the training data validation stage, the entire instance was removed. For the evaluation data, both automatic and manual verification were performed. To minimize unnecessary data loss, only the language-specific entry containing the error was discarded, and the remaining languages were retained. Consequently, the number of evaluation instances varies across languages. Table 2 summarizes the final training and evaluation sizes for each language.

Table 2. Amount of training and evaluation data per language [32].

		Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Train & Validation		11,353						
Evaluation	Total	410	548	406	416	617	427	422
	Cell Matching	324	432	319	324	483	335	334
	Reasoning	86	116	87	92	134	92	88

Table 3 summarizes the number of translated training instances generated by the open LLMs for each target language. As shown in table, the final number of translated instances differs across languages. This variation arises from the filtering stage. In some cases, the open LLM did not fully follow the translation instruction and produced table

texts with structural inconsistencies such as incorrect row or column alignment. These malformed outputs were removed during filtering, resulting in language-specific differences in dataset size.

Table 3. Statistics of table data translated by each model.

Models	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Gemini	7487	9274	7548	7487	11,353	7773	8007
Gemma-3-12b-it	8361	9113	8580	8559	11,353	8345	8209
Gemma-3-27b-it	6260	6358	5465	6131	11,353	6197	5950

5.2. Experimental Results with Encoder-Based Models

Before presenting a detailed analysis of the experimental results, we clarify the notation used in the tables. CTL denotes models trained with contrastive learning, while Adv indicates models trained with adversarial training. MLM refers to models that additional pre-trained using masked language modeling, and CS denotes models pre-trained with the cell selection task. Adapter indicates models trained using the k-adapter framework, where the backbone parameters of the pre-trained language model are frozen and only adapter modules are updated.

Table 4 compares cross-lingual adaptation strategies for multilingual TQA. Overall, combining CTL and Adv yields the best performance across most languages. By question type, Adv helps more on cell matching, while CTL provides larger gains on reasoning.

Table 4. Performance of multilingual table question answering by data type and language (F1/EM) [32]. Bold values indicate the best performance among the evaluated models.

Model	Languages						
	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Total							
Vanilla	53.75/44.00	60.15/53.47	58.50/51.60	61.14/56.38	71.58/63.79	48.37/41.07	57.02/49.16
+ CTL	60.88/57.45	63.14/60.43	65.64/62.71	61.29/59.75	72.96/68.01	57.14/53.99	61.66/57.00
+ Adv	60.56/57.94	66.15/63.00	63.64/60.98	61.71/60.00	71.00/66.39	54.57/51.87	62.38/59.14
+ CTL + Adv	61.20/58.19	66.36/63.00	65.76/62.96	65.65/63.85	73.53/68.50	55.81/53.28	63.40/60.33
Cell Matching							
Vanilla	55.21/43.65	61.38/53.48	60.13/52.20	64.02/58.82	74.17/65.14	48.92/40.71	57.57/48.64
+ CTL	61.90/58.20	66.13/63.72	68.17/65.09	63.07/61.30	76.52/70.74	56.85/53.59	62.90/57.95
+ Adv	62.65/60.06	69.46/66.51	67.29/64.79	64.96/62.84	74.37/69.29	54.91/52.39	64.36/61.56
+ CTL + Adv	63.12/60.06	68.12/64.65	68.74/66.35	67.33/65.32	76.15/70.33	55.98/53.89	64.45/60.96
Reasoning							
Vanilla	48.28/45.34	55.60/53.44	52.53/49.42	51.02/47.82	62.26/58.95	46.40/42.39	54.94/51.13
+ CTL	57.01/54.65	52.05/48.27	56.40/54.02	55.02/54.34	60.16/58.20	58.18/55.43	56.95/53.40
+ Adv	52.70/50.00	53.85/50.00	50.52/47.12	51.23/50.00	58.90/55.97	53.37/50.00	54.92/50.00
+ CTL + Adv	53.96/51.16	59.81/56.89	54.87/50.57	59.78/58.69	64.13/61.94	55.20/51.08	59.46/57.95

In the cell-matching category, the Adv model outperformed the CTL model, whereas in the reasoning category, the CTL model yielded larger performance gains than Adv. These results indicate different roles for the two methods. Adversarial training mainly improves cross-lingual generalization, while contrastive learning contributes more to reasoning-oriented representations.

Table 5 presents the multilingual TQA performance for each language under different pre-training strategies. In this table, MLM refers to the model pre-trained with masked

language modeling objectives using Wikipedia tables, and CS denotes the model pre-trained with the cell selection task based on synthetic SQL data. Adapter represents the model in which an adapter layer was added and the backbone parameters were frozen during pre-training. The numbers in brackets show how much the model improved compared to the vanilla model.

Table 5. Performance of models based on pre-training methods (F1/EM) [32]. Bold values indicate the best performance among the evaluated models.

Model	Languages						
	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Vanilla	53.75/44.00	60.15/53.47	58.50/51.60	61.14/56.38	71.58/63.79	48.37/41.07	57.02/49.16
MLM	59.39/56.47 (+5.64/+12.47)	63.40/61.53 (+3.25/+8.06)	64.87/61.97 (+6.37/+10.37)	65.31/64.09 (+4.17/+7.71)	72.21/68.34 (+0.63/+4.55)	56.04/53.52 (+7.67/+12.45)	59.81/56.53 (+2.79/+7.37)
CS	64.01/60.39 (+10.26/+16.39)	74.37/72.16 (+14.22/+18.69)	73.01/70.37 (+14.51/+18.77)	70.77/69.15 (+9.63/+12.77)	78.34/74.67 (+6.76/+10.88)	60.63/57.27 (+12.26/+16.20)	65.56/62.70 (+8.54/+13.54)
MLM + CS	68.41/64.54 (+14.66/+20.54)	75.72/72.52 (+15.57/+19.05)	73.84/71.35 (+15.34/+19.75)	71.26/70.12 (+10.12/+13.74)	81.22/77.59 (+9.64/+13.80)	60.06/57.04 (+11.69/+15.97)	71.18/67.93 (+14.16/+18.77)
Adapter + CS	62.84/59.65 (+9.09/+15.65)	69.59/65.93 (+9.44/+12.46)	68.92/65.67 (+10.42/+14.07)	62.60/60.72 (+1.46/+4.34)	74.96/69.96 (+3.38/+6.17)	57.86/54.46 (+9.49/+13.39)	64.81/61.28 (+7.79/+12.12)
Adapter + MLM + CS	67.28/64.54 (+13.53/+20.54)	70.18/67.03 (+10.03/+13.56)	69.45/67.16 (+10.95/+15.56)	66.65/64.81 (+5.51/+8.43)	78.77/75.00 (+7.19/+11.21)	58.51/55.63 (+10.14/+14.56)	63.97/60.80 (+6.95/+11.64)

Even though the model was pre-trained only on SQL queries and English table data, the model trained with the cell selection objective achieved the highest performance in every language. When we first applied maskLM pre-training and then performed additional pre-training with the cell selection task, the performance improved substantially. In contrast, the model pre-trained with the adapter layer showed a decline in performance. These results indicate that using English SQL data for pre-training did not negatively affect multilingual performance. However, freezing the backbone model's weights reduced the overall effectiveness of the pre-training.

The variation in performance gains across languages can be attributed to multiple factors, and we observe that Korean, Japanese, and Thai exhibit relatively smaller improvements compared to other languages. This difference does not indicate a failure of the proposed approach but rather reflects variation in improvement magnitude under diverse linguistic and data conditions. For example, Korean and Japanese differ typologically from other target languages in terms of word order and morphological structure, which can influence how effectively shared multilingual representations are exploited. Similarly, differences in available training data and resource conditions across languages may affect the extent to which additional pre-training and adaptation objectives can be leveraged.

Importantly, although the magnitude of improvement varies across languages, all target languages consistently show substantial gains over the vanilla model. These results demonstrate that the proposed approach is broadly applicable across diverse linguistic settings, even though the extent of improvement naturally differs depending on linguistic characteristics and data conditions.

Table 6 summarizes the results obtained when various adaptation strategies were applied to each pre-trained model. Overall, models with additional table-based pre-training exhibit smaller performance gains relative to the vanilla model. However, this effect is not uniform across all pre-training configurations. In particular, performance degradation due to adaptation is observed primarily when masked language modeling and the cell selection objective are jointly applied during pre-training. In contrast, models pre-trained with a single objective do not consistently show comparable degradation when adaptation methods are introduced.

Table 6. Performance variation by adaptation method for each pre-training model (F1/EM) [32]. Bold values indicate the best performance among the evaluated models.

Model	Korean (Original Table Language)				Others			
	Training Method							
	Vanilla	+ CTL	+ Adv	+ Adv + CTL	Vanilla	+ CTL	+ Adv	+ Adv + CTL
Vanilla	71.58/63.79	72.96/68.01	71.00/66.39	73.53/68.50	56.49/49.28	63.17/59.97	63.88/61.35	64.62/61.87
MLM	72.22/68.34	74.17/69.64	74.19/69.96	75.70/71.75	61.46/59.02	63.24/60.69	62.99/60.01	63.18/60.94
CS	78.34/74.67	77.89/74.02	79.08/75.32	77.04/73.21	68.06/65.34	69.36/66.60	67.46/64.94	67.02/64.09
MLM + CS	81.22/77.59	79.67/75.64	80.95/77.11	80.92/76.94	70.07/67.25	68.47/65.88	69.56/67.10	69.06/66.51
Adapter + MLM + CS	78.77/75.00	79.04/75.32	77.83/73.53	77.79/74.51	66.00/63.33	66.71/64.22	65.12/62.76	66.64/64.16

These findings suggest that cross-lingual adaptation, which explicitly modifies the representation space, can interfere with representations that have already been strongly shaped by multiple task-oriented pre-training objectives. This interaction appears to be less pronounced when the pre-training objective remains more general, such as masked language modeling alone, indicating that the negative effect arises from representation-level interactions rather than from the adaptation strategies themselves.

Notably, this negative interaction is substantially reduced in the adapter-based setting. Since adapter-based training freezes the backbone parameters of the pre-trained language model and updates only lightweight adapter modules, the representations learned during pre-training are largely preserved. This observation suggests that the performance degradation observed when both pre-training objectives are jointly applied arises from representation modifications introduced during cross-lingual adaptation, which can interfere with knowledge acquired in earlier pre-training stages.

5.3. Experimental Results with Large Language Models

In this subsection, we present the experimental results obtained from fine-tuning and evaluating LLMs on multilingual table question answering. Our analysis focuses on how different parameter sizes, training data configurations, and domain conditions influence model performance across languages and table types.

Table 7 presents a comparison of model performance depending on the translation models used to construct the training data. Overall, the performance gap between models trained on data translated by open LLMs and those translated by commercial LLMs was relatively small. This trend suggests that the quality of recent open-source LLMs has improved substantially, and that the impact of occasional translation errors in the training data remains limited as long as the amount of erroneous data is not large.

Table 7. Performance comparison of LLaMA-3.2-3B-Instruct on multilingual KorQuAD 2.0 using different translation models (F1/EM). Bold values indicate the best performance among the evaluated models.

Translation Models	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Gemini	75.31/63.90	81.00/74.45	79.11/72.05	75.48/68.31	87.47/79.26	74.20/64.56	78.33/71.17
Gemma-3-12b-it	76.82/63.52	82.44/74.84	81.98/73.30	76.80/69.36	90.04/81.47	75.16/65.22	78.39/69.17
Gemma-3-27b-it	77.83/64.48	82.09/74.14	80.33/71.95	76.21/67.74	88.39/77.81	75.72/65.31	76.21/67.74

Models trained on data translated by Gemma-3 tended to achieve slightly higher F1 scores, while those trained on Gemini-translated data showed slightly higher EM scores. Interestingly, the model trained on data translated by Gemma-3-12B performed better than the one translated by Gemma-3-27B. This result seems to come not from translation quality itself but from the fact that Gemma-3-12B produced outputs that matched the required

format more reliably. Since fewer examples were filtered out, the model was able to learn from a larger amount of training data.

Table 8 reports the performance of LLaMA-3 models with different parameter sizes under three training data settings. Korean-Only refers to models trained solely on the original Korean KorQuAD 2.0 data. Multi-lang Query uses the same Korean table data but replaces the queries translated into target languages. Multi-lang Dataset uses Gemini-translated tables and queries, where all content is translated into each target language.

Table 8. Performance comparison of LLaMA-3 models with different parameter sizes under varying training data configurations derived from KorQuAD 2.0 (F1/EM).

Train Dataset	Size	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Korean-Only	1B	64.32/43.50	71.93/61.87	69.01/57.93	65.07/52.80	80.61/70.00	63.27/50.98	67.43/56.80
Multi-lang Query	1B	59.55/34.84	78.43/68.16	75.53/61.70	65.15/44.43	84.90/76.11	65.67/48.16	72.17/59.46
Multi-lang Dataset	1B	66.66/48.70	74.48/64.44	71.86/60.05	68.46/58.42	80.55/69.69	66.78/54.94	71.80/60.98
Korean-Only	3B	70.41/57.65	78.33/71.81	73.53/66.34	71.11/62.22	87.20/79.89	68.11/59.00	73.14/65.65
Multi-lang Query	3B	72.59/51.87	83.87/76.94	82.54/74.56	70.29/49.95	91.63/85.19	74.36/59.00	79.57/69.93
Multi-lang Dataset	3B	75.31/63.90	81.00/74.45	79.11/72.05	75.48/68.31	87.47/79.26	74.20/64.56	78.33/71.17
Korean-Only	8B	77.20/65.92	79.51/71.89	78.76/70.69	71.93/63.17	87.34/80.02	72.77/65.92	75.81/67.45
Multi-lang Query	8B	51.28/36.09	83.81/76.24	80.08/71.56	55.99/37.86	91.43/85.63	48.52/37.70	76.23/64.03
Multi-lang Dataset	8B	77.78/67.46	82.69/76.16	79.87/72.14	74.99/67.17	88.87/81.53	76.83/64.75	78.49/70.40

For the 8B model trained with the Multi lang Query setup, we observed a notable drop in performance for Arabic, Japanese, and Thai. Under the supervised fine tuning setting adopted in this study, all LLaMA 3 models were trained and evaluated using identical prompts and system instructions. Despite this controlled setup, the performance degradation was observed only for the 8B model in these specific languages. This suggests that the reduced scores are unlikely to stem from insufficient instruction clarity or deficiencies in reasoning ability. Instead, the degradation primarily arises from model-specific and language-specific generation behaviors that lead to deviations from the required answer format, which in turn negatively affect strict evaluation metrics such as EM and F1.

Aside from these outliers, all three model sizes (1B, 3B, and 8B) generally showed higher performance when trained with translated data than when trained only on Korean data. This indicates that adding target-language supervision, whether through translated queries or fully translated datasets, helps improve cross-lingual table QA performance.

However, unlike the encoder-based results in Table 9, where models trained on fully translated datasets showed a clear advantage over those trained with translated queries only, the LLaMA-3 models exhibited much smaller differences across the two settings. This suggests that decoder-based LLMs already possess strong cross-lingual generalization, and therefore do not rely as heavily on fully translated table data to achieve competitive performance on multilingual TQA tasks.

Table 10 presents the performance of LLaMA-3 models of different parameter sizes on each domain of the DomainTQA dataset. The tables in the Public Administrative Document domain are extracted from government documents and therefore resemble many of the Wikipedia tables used in KorQuAD. They contain fewer domain-specific terms and tend to follow more familiar tabular structures. As a result, models trained on the multilingual KorQuAD 2.0 dataset generally outperformed those trained solely on the Korean DomainTQA tables for this domain.

Table 9. Performance of multilingual TQA model with addition of translated table context data (F1/EM) [32]. Bold values indicate the best performance among the evaluated models.

Model	Korean (Original Table Language)				Others			
	Number of Translated Train Table Dataset							
	0	500	1000	5000	0	500	1000	5000
MLM	72.22/68.34	76.72/73.05	77.21/73.37	79.61/75.97	61.46/59.02	65.36/62.10	66.32/64.03	69.42/67.00
MLM + CTL	74.17/69.64	77.97/73.70	76.50/72.40	79.36/75.48	63.24/60.69	66.04/63.35	65.37/62.73	69.43/66.87
MLM + Adv	74.19/69.96	75.35/72.07	75.76/62.24	78.76/75.32	62.99/60.01	62.54/60.07	65.54/53.32	68.48/65.84
MLM + CTL + Adv	77.04/73.21	75.45/71.59	76.45/73.21	78.65/74.67	63.18/69.94	63.84/61.64	66.20/63.75	69.99/67.55
MLM + CS	81.22/77.59	80.27/76.78	81.18/78.40	80.74/77.27	70.07/67.25	70.63/68.01	72.14/69.47	73.09/70.31
Adapter + MLM + CS	78.77/75.00	79.35/75.97	76.49/73.21	80.68/76.78	66.00/63.33	67.10/64.35	68.45/65.77	69.60/66.82

Table 10. Performance comparison of LLaMA-3 models across different parameter sizes and domains in the DomainTQA dataset (F1/EM).

Domain	Train Dataset	Size	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Public Administrative Document Tables	KorQuAD 2.0	1B	63.85/38.52	72.52/57.01	65.93/44.57	60.03/44.98	70.49/56.71	59.81/39.28	61.82/41.88
	DomainTQA	1B	58.71/38.63	71.34/54.94	64.09/43.31	63.76/47.57	75.68/65.69	58.71/38.63	60.31/40.83
	KorQuAD 2.0	3B	72.24/54.42	81.24/72.00	75.08/61.85	73.95/63.02	81.39/70.75	72.39/56.49	72.95/58.35
	DomainTQA	3B	71.42/49.31	78.71/67.14	73.05/57.70	73.01/57.68	83.71/75.59	68.44/51.54	69.05/53.02
	KorQuAD 2.0	8B	78.69/60.90	85.04/75.11	79.17/65.20	76.81/65.29	85.22/77.20	74.63/56.81	75.89/61.90
Legal Document Tables	DomainTQA	8B	76.16/59.12	81.41/71.53	75.61/61.05	76.45/65.69	83.52/75.36	72.93/58.60	74.05/60.45
	KorQuAD 2.0	1B	28.66/13.95	44.20/24.75	43.29/19.39	38.00/20.46	46.69/23.49	38.79/17.87	45.31/21.94
	DomainTQA	1B	62.93/22.09	63.45/35.92	63.28/26.58	63.30/27.55	71.33/41.73	61.30/27.99	61.61/28.81
	KorQuAD 2.0	3B	50.21/28.23	60.31/40.61	56.02/33.77	55.66/31.35	57.89/30.44	52.01/28.33	58.47/34.67
	DomainTQA	3B	74.08/38.53	76.72/55.66	74.86/39.63	73.10/41.91	77.97/46.21	71.09/35.91	74.37/46.06
Product Specification Tables	KorQuAD 2.0	8B	73.63/42.02	82.69/63.43	71.46/44.81	67.92/40.75	77.25/46.83	72.18/37.77	74.24/48.57
	DomainTQA	8B	75.80/44.51	83.47/62.94	77.65/46.15	76.23/50.49	78.96/51.31	72.87/39.62	75.44/49.24
	KorQuAD 2.0	1B	45.88/22.65	50.95/28.75	42.14/20.92	53.66/30.62	56.85/36.97	40.02/18.02	49.99/27.65
	DomainTQA	1B	53.52/29.78	56.64/31.62	52.51/27.81	55.67/31.62	63.89/40.66	48.08/24.15	51.53/26.40
	KorQuAD 2.0	3B	48.38/30.03	50.70/31.00	42.36/23.68	63.19/34.50	60.71/46.06	44.23/24.40	45.01/24.40
	DomainTQA	3B	63.14/42.92	65.11/47.25	61.81/45.36	71.99/56.75	77.52/61.67	60.38/42.42	61.71/42.80
	KorQuAD 2.0	8B	64.30/45.43	77.33/63.50	63.33/45.98	76.73/63.12	76.83/64.37	63.57/47.68	65.46/49.93
	DomainTQA	8B	66.73/47.93	72.32/55.50	66.46/50.00	75.12/58.50	78.18/61.67	67.37/49.06	67.11/49.56

In contrast, the Legal Document and Product Specification domains contain a large number of specialized terms and table formats that differ noticeably from those found in KorQuAD. In these domains, models trained only on the original Korean DomainTQA data achieved higher performance for all languages, including Korean itself, even though the amount of DomainTQA training data was much smaller than the translated KorQuAD 2.0 data. This suggests that even when trained in a single language, the ability to understand and reason over domain-specific structures and terminology can transfer effectively across languages. The difference between the two training settings was most pronounced for the smaller 1B and 3B models, whereas the 8B model showed almost no performance gap.

Tables 11 and 12 present the performance of the Polyglot-1.3B model on KorQuAD and DomainTQA. In Table 11, the model trained only with Korean data shows reasonably strong performance for Korean and English, but its performance drops sharply for all other languages. When multilingual queries or multilingual datasets are added during training, the model does improve, yet the performance gap between Korean and the other target languages remains much larger than what we observed in the LLaMA experiments.

Table 11. Performance comparison of Polyglot-1.3B across different training data configurations (F1/EM).

Train Dataset	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Korean-Only	16.64/ 2.33	42.35/15.68	27.53/4.57	23.06/ 7.60	52.02/32.56	15.45/ 4.38	25.51/ 5.04
Multi-lang Query	22.32/ 7.26	48.94/23.97	36.41/13.00	30.92/12.64	54.08/36.20	23.54/ 9.72	37.57/15.14
Multi-lang Dataset	30.43/12.19	52.61/29.44	42.09/21.11	39.45/21.17	55.98/38.18	27.79/13.69	47.30/25.86

Table 12. Performance of Polyglot-1.3B across different domain-specific dataset (F1/EM).

Domain	Train Dataset	Arabic	English	Indonesian	Japanese	Korean	Thai	Vietnamese
Public Administrative Document Tables	KorQuAD 2.0	31.87/ 6.12	53.16/20.40	38.19/10.15	32.58/11.48	48.64/27.94	28.04/ 3.47	40.04/10.87
	DomainTQA	10.32/ 1.02	42.51/ 8.13	24.07/ 2.29	42.51/ 8.13	37.56/16.94	6.38/ 2.08	27.42/ 2.63
Legal Document Tables	KorQuAD 2.0	19.57/ 7.45	27.48/ 7.05	25.28/ 6.62	17.58/ 6.62	31.46/13.33	17.77/ 7.51	27.72/ 7.12
	DomainTQA	2.70/ 0.03	42.43/ 5.24	14.58/ 0.08	20.29/ 4.85	48.26/15.19	2.86/ 1.15	22.83/ 3.13
Product Specification Tables	KorQuAD 2.0	33.39/ 0.00	51.70/11.11	34.65/ 0.00	45.38/20.00	47.10/20.89	52.63/ 0.00	46.08/20.00
	DomainTQA	0.00/ 0.00	36.94/ 6.94	22.00/ 4.00	31.98/ 0.00	49.59/23.88	0.00/ 0.00	20.12/ 0.00

A similar trend appears in Table 12. For the Legal Document and Product Specification domains, Polyglot-1.3B performs well only in Korean, while its accuracy for the other languages remains extremely low. This finding is notably different from the LLaMA results, where models trained only on the Korean DomainTQA data performed better than those trained on multilingual KorQuAD for these domains. For Polyglot, the model trained on Korean DomainTQA data does not outperform the multilingual KorQuAD trained model for any of the non-Korean languages.

These observations highlight an important point. When an LLM has limited cross-lingual generalization ability, as in the case of Polyglot-1.3B, performance improvements through dataset design alone become difficult. Meaningful gains require either training directly in the target languages or applying an explicit cross-lingual adaptation strategy. Without such support, the model struggles to transfer domain knowledge or reasoning skills beyond the languages it was primarily trained on.

6. Ablation Study

We carried out additional experiments to examine the impact of table-based pre-training and cross-lingual adaptation techniques on performance in settings where translated table context is not available. Furthermore, we conducted supplementary analyses to determine which input representations are most effective when applying cross-lingual adaptation methods.

Table 9 reports the performance across different numbers of training instances as translated table data in each language is incrementally incorporated into the training set. When 5000 translated table instances were added, the performance difference between models employing adaptation methods and the vanilla baseline became negligible. This suggests that adaptation techniques are particularly beneficial in low-resource scenarios, where only a small amount of translated table data—or solely original-language table data—is available. Additionally, the performance gap between models trained with and without translated data was further reduced when additional table-based pre-training was applied.

Table 13 presents the performance variations observed when different input representations are used for the adaptation methods. Models trained with adversarial learning exhibited inconsistent performance depending on the specific model configuration. In contrast, models employing contrastive learning consistently achieved the best results when using a representation vector obtained by mean-pooling over all table token representations.

Table 13. Performance with hidden representation input in cross-lingual adaptation methods (F1/EM) [32]. Bold values indicate the best performance among the evaluated models.

		MLM		MLM + CS	
		CTL	ADV	CTL	ADV
Korean	None	72.22/68.34		81.22/77.59	
	[CLS]	75.70/71.91	74.19/69.96	79.17/75.64	80.95/77.11
	Answer Cell	75.95/72.24	75.61/71.91	79.67/75.64	81.40/78.08
	Header Cell	75.95/72.24	73.21/69.31	79.30/75.97	81/30/77.92
	Header + Answer Cell	76.05/72.88	74.43/71.10	77.61/74.51	82.03/78.40
	All Table Cell	76.64/72.72	75.88/72.24	81.98/78.40	81.89/78.24
Others	None	61.46/59.02		70.07/67.25	
	[CLS]	61.24/59.06	62.99/60.01	65.03/63.09	69.56/67.10
	Answer Cell	63.24/60.69	62.57/60.51	68.47/65.88	68.90/66.06
	Header Cell	60.93/58.55	62.17/59.76	65.04/62.68	70.70/67.94
	Header + Answer Cell	59.59/57.41	60.85/58.57	64.44/62.01	69.41/66.70
	All Table Cell	64.01/61.29	62.70/60.42	69.61/67.10	68.71/66.16

7. Limitations

In this study, we conducted multilingual table question answering experiments primarily on the KorQuAD 2.0 dataset. This choice was motivated by practical considerations, as the native language of the researchers responsible for verifying translated data was Korean, enabling reliable quality control during dataset construction and evaluation. While this setup allowed us to carefully analyze cross-lingual generalization under controlled conditions, it also limits the diversity of table structures and linguistic phenomena covered in the experiments. Since many widely used table QA benchmarks are originally published in English, extending our evaluation to English-centric datasets remains an important direction for future work.

Another limitation arises from computational constraints during pre-training. All table-specific pre-training experiments were conducted using RTX-3090 GPUs, which imposed strict limitations on batch size, sequence length, and the total number of training steps. To ensure stable and reproducible training under this hardware configuration, we restricted the size of the pre-training corpora for both masked language modeling and cell selection objectives. As a result, the pre-training data used in our experiments represents only a subset of the available resources, and larger-scale pre-training may lead to different performance trends, particularly for multilingual and domain-specific generalization.

Our experimental results further indicate that while adversarial training consistently improves performance in cell-matching tasks and contrastive learning is particularly effective for reasoning-oriented questions, applying these adaptation techniques on top of models that have already undergone cell selection pre-training can lead to performance degradation. This suggests that certain adaptation objectives may interfere with strong structural priors learned during table-specific pre-training, especially when training capacity is constrained. Investigating more flexible or configurable training strategies—such as scaling pre-training data, decoupling adaptation stages, or leveraging more powerful hardware—remains an important avenue for future research.

A further limitation concerns the applicability of the proposed cross-lingual adaptation methods to different model architectures. The contrastive learning and adversarial training strategies introduced in this work are designed for encoder-based models, where explicit sentence- or cell-level representations can be directly aligned across languages. In contrast, applying these techniques to decoder-only large language models is non-trivial, as such

models do not expose intermediate representations in a manner that readily supports representation-level alignment or adversarial objectives.

As a result, the LLM-based experiments in this study do not incorporate the proposed cross-lingual adaptation mechanisms, and instead rely on the inherent multilingual generalization capabilities of decoder-only models. While our results show that modern LLMs exhibit strong robustness to cross-lingual table reasoning even without explicit adaptation, this architectural gap limits a direct comparison between encoder-based adaptation methods and decoder-based training paradigms. Developing cross-lingual adaptation techniques that are compatible with decoder-only architectures remains an important direction for future research, particularly for enabling more principled multilingual alignment in large language models.

We note that the variation in improvement magnitude across languages is not fully explained within the scope of this study. While linguistic characteristics such as morphological structure or script differences may contribute to cross-lingual performance variation, we do not provide a dedicated analysis to isolate or quantify their effects. As a result, our discussion focuses on empirical observations rather than attributing performance differences to specific linguistic causes. A more detailed analysis of how linguistic properties interact with cross-lingual adaptation in table question answering, such as the role of tokenization, morphology, or script characteristics, is left for future work.

8. Conclusions

In this work, we investigated the construction of multilingual table question answering systems in settings where only limited resources are available. Since translating large table datasets is both costly and labor-intensive, we proposed an efficient strategy that translates only the question text while keeping the original table content intact. This approach leverages the language-agnostic nature of tabular structures, in which the meaning of a cell is largely independent of language-specific syntax or word order, and allows for scalable multilingual dataset creation without compromising table structure.

To reduce the performance gap that arises when models are trained on untranslated tables paired with translated queries, we introduced cross-lingual adaptation techniques based on contrastive learning and adversarial training. These two methods contributed in complementary ways. Adversarial learning promoted the acquisition of language-invariant representations and improved performance in cell matching questions. Contrastive learning encouraged alignment of semantically equivalent queries written in different languages and yielded notable gains in reasoning oriented questions.

We additionally explored how table-based pre-training affects multilingual generalization. Pre-training on multilingual Wikipedia tables using a masked language modeling objective and on SQL-derived data using a cell selection objective substantially improved performance across all target languages. Among the two strategies, cell selection-based pre-training produced the largest improvements, demonstrating that structural supervision from SQL executions contributes strongly to multilingual table reasoning even when the pre-training data are written in English. However, once the model was pre-trained with strong table-specific objectives, the benefits of applying cross-lingual adaptation diminished, indicating that adaptation objectives and table-based pre-training may interact in non-trivial ways.

Beyond encoder-based models, we evaluated decoder-only large language models under the same multilingual TQA conditions. LLaMA-3 models exhibited strong cross-lingual generalization, achieving competitive or superior performance even when trained solely on Korean table data. The performance differences among training configurations, such as using translated queries or fully translated datasets, were considerably smaller than those

observed with encoder models. This suggests that modern large language models possess robust multilingual reasoning capabilities that extend naturally to structured tabular data. Experiments on domain-specific datasets further revealed that domain knowledge learned from Korean table data transfers effectively to other target languages, even without explicit multilingual supervision.

Overall, our findings demonstrate that multilingual table question answering can be effectively achieved without fully translating table datasets. Selective question translation, cross-lingual adaptation, and table-based pre-training provide a practical solution for encoder-based architectures in low-resource environments. At the same time, decoder-only large language models present a powerful alternative due to their inherent multilingual robustness and their ability to generalize across domains and languages. Future work will extend our experiments to a broader range of multilingual benchmarks and explore adaptation techniques suitable for decoder-only architectures, with the goal of further improving multilingual table reasoning in low-resource or partially translated settings.

Author Contributions: Conceptualization, S.C.; Methodology, S.C.; Software, S.C.; Validation, S.C., M.K., H.-L.K. and J.-H.L.; Formal analysis, S.C., M.K., H.-L.K. and J.-H.L.; Investigation, S.C., M.K., H.-L.K. and J.-H.L.; Resources, S.C.; Data curation, S.C.; Writing—original draft preparation, S.C., M.K. and H.-L.K.; Writing—review and editing, S.C.; Visualization, S.C.; Supervision, J.-H.L., H.-C.K. and S.-J.L.; Project administration, J.-H.L., H.-C.K. and S.-J.L.; Funding acquisition, J.-H.L., H.-C.K. and S.-J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Commercialization Promotion Agency for R&D Outcomes(COMPA) funded by the Ministry of Science and ICT(MSIT).(RS-2025-25416427, University Technology Management Promotion Project(TLO Innovation Type)_NATIONAL KOREA MARITIME & OCEAN UNIVERSITY). This work was supported by Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. RS-2024-00338140, Development of learning and utilization technology to reflect sustainability of generative language models and up-to-dateness over time).

Data Availability Statement: The source data used for translation in this study are licensed under CC BY-ND 2.0 KR. Because the dataset used in our experiments contains modified/derived versions of the original materials, it cannot be redistributed under the terms of the license; therefore, the processed data used in this study are not publicly available. However, the data construction procedure relies only on publicly available tools and language models, and we provide sufficient methodological details to enable full reproduction of the dataset and experiments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Clark, J.; Choi, E.; Collins, M.; Garrette, D.; Kwiatkowski, T.; Nikolaev, V.; Palomaki, J. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 454–470.
2. Pasupat, P.; Liang, P. Compositional Semantic Parsing on Semi-Structured Tables. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Beijing, China, 26–31 July 2015.
3. Zhong, V.; Xiong, C.; Socher, R. Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning. *arXiv* **2017**, arXiv:1709.00103.
4. Yu, T.; Zhang, R.; Yang, K.; Yasunaga, M.; Wang, D.; Li, Z.; Ma, J.; Li, I.; Yao, Q.; Roman, S.; et al. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3911–3921.
5. Iyyer, M.; tau Yih, W.; Chang, M.W. Search-based Neural Structured Learning for Sequential Question Answering. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017.
6. Dou, L.; Gao, Y.; Pan, M.; Wang, D.; Che, W.; Zhan, D.; Lou, J.G. MultiSpider: Towards Benchmarking Multilingual Text-to-SQL Semantic Parsing. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 22 February–1 March 2022.
7. Nguyen, A.G.T.; Dao, M.; Nguyen, D.Q. A Pilot Study of Text-to-SQL Semantic Parsing for Vietnamese. *arXiv* **2020**, arXiv:2010.01891.

8. Shu, D.; Yang, J.; Wu, Z.; Wu, X.; Cheng, X.; Guan, X.; Wang, Y.; Wu, P.; Yang, T.; Zhu, H.; et al. M3TQA: Massively Multilingual Multitask Table Question Answering. *arXiv* **2025**, arXiv:2508.16265.
9. Touvron, H.; Martin, L.; Stone, K.R.; Albert, P.; Almahairi, A.; Babaei, Y.; Bayazitov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv* **2023**, arXiv:2307.09288.
10. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V.S. Domain-Adversarial Training of Neural Networks. *J. Mach. Learn. Res.* **2015**, *17*, 1–35.
11. Lee, S.; Kim, D.; Park, J. Domain-agnostic Question-Answering with Adversarial Training. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 3–7 November 2019.
12. Wang, H.; Gan, Z.; Liu, X.; Liu, J.; Gao, J.; Wang, H. Adversarial Domain Adaptation for Machine Reading Comprehension. *arXiv* **2019**, arXiv:1908.09209.
13. Bai, Z.; Wang, B.; Wang, Z.; Yuan, C.; Wang, X. Domain adaptive multi-task transformer for low-resource machine reading comprehension. *Neurocomputing* **2022**, *509*, 46–55.
14. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G.E. A Simple Framework for Contrastive Learning of Visual Representations. *arXiv* **2020**, arXiv:2002.05709.
15. Kim, D.; Park, S.; Kim, J.; Lee, J. SelfReg: Self-supervised Contrastive Regularization for Domain Generalization. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9599–9608.
16. Yang, J.; Duan, J.; Tran, S.; Xu, Y.; Chanda, S.; Chen, L.; Zeng, B.; Chilimbi, T.M.; Huang, J. Vision-Language Pre-Training with Triple Contrastive Learning. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 15650–15659.
17. Chen, B.; Guo, W.; Gu, B.; Liu, Q.; Wang, Y. Multi-Level Contrastive Learning for Cross-Lingual Alignment. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 22–27 May 2022; pp. 7947–7951.
18. Wang, Y.S.; Wu, A.; Neubig, G. English Contrastive Learning Can Learn Universal Cross-lingual Sentence Embeddings. *arXiv* **2022**, arXiv:2211.06127.
19. Han, X.; Luo, Y.; Chen, W.; Liu, Z.; Sun, M.; Botong, Z.; Fei, H.; Zheng, S. Cross-Lingual Contrastive Learning for Fine-Grained Entity Typing for Low-Resource Languages. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022.
20. Yin, P.; Neubig, G.; tau Yih, W.; Riedel, S. TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data. *arXiv* **2020**, arXiv:2005.08314.
21. Wang, Z.; Dong, H.; Jia, R.; Li, J.; Fu, Z.; Han, S.; Zhang, D. TUTA: Tree-based Transformers for Generally Structured Table Pre-training. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, New York, NY, USA, 21 August 2020.
22. Liu, Q.; Chen, B.; Guo, J.; Ziyadi, M.; Lin, Z.; Chen, W.; Lou, J.G. TAPEX: Table Pre-training via Learning a Neural SQL Executor. In Proceedings of the International Conference on Learning Representations, Virtual Event, 25–29 April 2022.
23. Herzig, J.; Nowak, P.K.; Müller, T.; Piccinno, F.; Eisenschlos, J. TaPas: Weakly Supervised Table Parsing via Pre-training. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; Jurafsky, D., Chai, J., Schluter, N., Tetreault, J., Eds.; pp. 4320–4333. <https://doi.org/10.18653/v1/2020.acl-main.398>.
24. Wang, R.; Tang, D.; Duan, N.; Wei, Z.; Huang, X.; Ji, J.; Cao, G.; Jiang, D.; Zhou, M. K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Online Event, 1–6 August 2021.
25. Kamath, G.T.A.; Ferret, J.; Pathak, S.; Vieillard, N.; Merhej, R.; Perrin, S.; Matejovicova, T.; Ram'e, A.; Rivière, M.; Rouillard, L.; et al. Gemma 3 Technical Report. *arXiv* **2025**, arXiv:2503.19786.
26. Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. The Llama 3 Herd of Models. *arXiv* **2024**, arXiv:2407.21783.
27. Riviere, G.T.M.; Pathak, S.; Sessa, P.G.; Hardin, C.; Bhupatiraju, S.; Hussenot, L.; Mesnard, T.; Shahriari, B.; Ram'e, A.; Ferret, J.; et al. Gemma 2: Improving Open Language Models at a Practical Size. *arXiv* **2024**, arXiv:2408.00118.
28. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
29. Ainslie, J.; Lee-Thorp, J.; de Jong, M.; Zemlyanskiy, Y.; Lebron, F.; Sanghai, S. GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–10 December 2023; Bouamor, H., Pino, J., Bali, K., Eds.; pp. 4895–4901. <https://doi.org/10.18653/v1/2023.emnlp-main.298>.
30. Ko, H.A.; Yang, K.; Ryu, M.; Choi, T.; Yang, S.; Hyun, J.; Park, S.Y.; Park, K. A Technical Report for Polyglot-Ko: Open-Source Large-Scale Korean Language Models. *arXiv* **2023**, arXiv:2306.02254.

31. Kim, H.L.; Cho, S.; Shin, J.; Kwon, H.C. Evaluating the Generalization Performance of Korean Table Machine Reading Comprehension across Domain-Specific Evaluation Datasets. In Proceedings of the Korea Software Congress 2022 (KSC 2022), Jeju, Republic of Korea, 20–23 December 2022; pp. 365–367.
32. Cho, S.; Kim, M.; Kim, H.L.; Lee, J.H.; Kwon, H.C. Multilingual Table Question Answering for Low-Resource Languages via Selective Question Translation and Cross-Lingual Adaptation. In Proceedings of the 2025 IEEE International Conference on Big Data and Smart Computing (BigComp), Kota Kinabalu, Malaysia, 9–12 February 2025; pp. 264–272.
33. Kim, Y.; Lim, S.; Lee, H.; Park, S.; Kim, M. KorQuAD 2.0: Korean QA Dataset for Web Document Machine Comprehension. In Proceedings of the Annual Conference on Human and Language Technology, Minneapolis, MN, USA, 2–7 June 2019.
34. Wang, K.; Zhu, J.; Ren, M.; Liu, Z.; Li, S.; Zhang, Z.; Zhang, C.; Wu, X.; Zhan, Q.; Liu, Q.; et al. A Survey on Data Synthesis and Augmentation for Large Language Models. *arXiv* **2024**, arXiv:2410.12896.
35. Nadăș, M.; Dioșan, L.; Tomescu, A. Synthetic Data Generation Using Large Language Models: Advances in Text and Code. *IEEE Access* **2025**, *13*, 134615–134633.
36. Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L.Y.; Edunov, S.; Chen, D.; tau Yih, W. Dense Passage Retrieval for Open-Domain Question Answering. *arXiv* **2020**, arXiv:2004.04906.
37. Cho, S.; Park, S.; Kim, H.L.; Lee, J.H.; Shin, J.; chul Kwon, H. Multi-Paragraph Machine Reading Comprehension with Hybrid Reader over Tables and Text. *Appl. Artif. Intell.* **2024**, *38*, 2367820.
38. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised Cross-lingual Representation Learning at Scale. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019.
39. Rajpurkar, P.; Zhang, J.; Lopyrev, K.; Liang, P. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.