

MultiSpider: Towards Benchmarking Multilingual Text-to-SQL Semantic Parsing

Longxu Dou¹, Yan Gao², Mingyang Pan¹, Dingzirui Wang¹,
Wanxiang Che¹, Dechen Zhan¹, Jian-Guang Lou²

¹ Harbin Institute of Technology

² Microsoft Research Asia

{lxdou, mypan, dzrwang, car}@ir.hit.edu.cn, dechen@hit.edu.cn,
{yan.gao, jlou}@microsoft.com

Abstract

Text-to-SQL semantic parsing is an important NLP task, which greatly facilitates the interaction between users and the database and becomes the key component in many human-computer interaction systems. Much recent progress in text-to-SQL has been driven by large-scale datasets, but most of them are centered on English. In this work, we present MULTISPIDER, the *largest* multilingual text-to-SQL dataset which covers seven languages (English, German, French, Spanish, Japanese, Chinese, and Vietnamese). Upon MULTISPIDER, we further identify the lexical and structural challenges of text-to-SQL (caused by specific language properties and dialect sayings) and their intensity across different languages. Experimental results under three typical settings (zero-shot, monolingual and multilingual) reveal a 6.1% absolute drop in accuracy in non-English languages. Qualitative and quantitative analyses are conducted to understand the reason for the performance drop of each language. Besides the dataset, we also propose a simple schema augmentation framework SAVE (Schema-Augmentation-with-Verification), which significantly boosts the overall performance by about 1.8% and closes the 29.5% performance gap across languages.

1 Introduction

Text-to-SQL semantic parsing is the task of mapping natural language sentences into executable SQL database queries, which serves as an important component in many natural language interface systems such as question answering and task-oriented dialogue. Despite the substantial number of systems (Yin and Neubig 2018; Guo et al. 2019; Wang et al. 2020; Scholak, Schucher, and Bahdanau 2021) and benchmarks (Yu et al. 2018, 2019a,b; Guo et al. 2021) for text-to-SQL, most of them are predominantly built in English, excluding this powerful tool’s accessibility to non-English speakers. The reason for this limitation lies in the serious lack of high-quality multilingual text-to-SQL datasets.

Several works attempted to extend to new languages, but currently available multilingual text-to-SQL datasets only support four languages (English, Chinese, and Vietnamese and Portuguese) (Yu et al. 2018; Min and Zhang 2019; Tuan Nguyen, Dao, and Nguyen 2020; José and Cozman 2021), which hinders the study of multilingual text-to-SQL

Lang.	Question	Schema with Augmentation
English	Return the record companies of orchestras, sorted descending by the years in which they were founded.	year of founded {established year, the year of foundation}
German	Geben Sie die Plattenfirmen von Orchestern zurück, absteigend sortiert nach den Jahren ihrer Gründung.	Gründungsjahr {jahr der grundlage, jahr der gründung}
French	Listez les maisons de disques des orchestres, triées par ordre décroissant des années de leur création.	année de fondation {année de creation}
Spanish	¿Cuáles son las compañías discográficas de las orquestas en orden descendente de años de fundación?	año de fundación {Año Establecido, año de creación}
Chinese	返回按创立年份降序排列的乐团唱片公司的名称。	成立年份 {创立之年, 建立之年}
Japanese	創設年の降順でオーケストラのレコード会社を並べる。	創設年 {創業年, 設立年}
Vietnam	Liệt kê các công ty thu âm của các dàn nhạc theo thứ tự giảm dần về năm mà từng công ty được thành lập .	năm thành lập {năm sáng tạo}

↓

SELECT Record_Company FROM orchestra ORDER BY Year_of_Founded DESC

SELECT 唱片公司 FROM 管弦乐队 ORDER BY 成立年份 DESC

Figure 1: Examples of MULTISPIDER.

across a broad spectrum of language distances. Besides the language coverage, the existing multilingual datasets also suffer from the following limitations: (1) *low-quality*: unnatural or inaccurate translations; (2) *in-completed translation*: the database of Chinese-Spider and Portuguese-Spider are not translated and kept in English. These limitations will inevitably lead to a limited multilingual system. To advance multilingual text-to-SQL, in this paper, we present MULTISPIDER, the *largest* and *high-quality* multilingual text-to-SQL dataset, which covers seven main-stream languages (Sec 2.1). Figure 1 lists one example across seven languages including both question and schema. To ensure the dataset quality, we first identify five typical translation mistakes during constructing a multilingual text-to-SQL dataset (Sec 2.2), then we carefully organize the construction pipeline consisting of multi-round translation and validation (Sec 2.3). Most importantly, we take into account of the specific language properties to make the question more natural and realistic.

Besides high-quality, MULTISPIDER is quite challenging in multilingual text-to-SQL. Concretely, we explore the lex-

ical and structural challenge (Herzig and Berant 2018) of MULTISPIDER (Sec 2.4): (1) lexical challenge refers to mapping the entity mentions to schema alias (e.g., ‘record companies’ to RECORD_COMPANY); (2) structural challenge refers to mapping the intentions to SQL operators (e.g., ‘sorted descending’ to DESC). Experimental results and analysis demonstrate that (1) the specific language properties like Hiragana and Katakana (Japanese) and morphologically rich language (German and French) make the lexical challenge more difficult by expanding the syntactic difference between schema and tokens; (2) the dialect sayings require common-sense reasoning to address structural challenge (Figure 3).

To address the lexical challenges, we propose a simple data augmentation framework SAVE from the view of schema, which is more generic compared with the language-specific approaches (Min and Zhang 2019; Tuan Nguyen, Dao, and Nguyen 2020) (e.g. PhoBERT for Vietnamese (Nguyen and Nguyen 2020)). Concretely, SAVE consists of three steps (Sec 3.1): (1) conducting back-translation on contextualized schema using machine-translation; (2) extracting the the schema candidates; (3) measuring the semantic equivalency (Pi et al. 2022) with natural language inference model (NLI) to collect the suitable candidate. The quantitative and qualitative analysis prove that (1) the augmented schema including synonyms and morphological variants; (2) verification improve the accuracy of augmented data from 33.2% to 74.5% under human evaluation (Sec 3.2).

To examine the challenge of MULTISPIDER and verify the effectiveness of SAVE, we conduct extensive experiments (Sec 4) under representative settings (zero-shot transfer, monolingual and multilingual). Experimental results reveal the absolute drop of accuracy in non-English languages is about 6.1% on average, indicating the difference in language causes the performance gap. SAVE significantly boosts the overall performance by about 1.8%, reducing the performance gap by 29.5% across languages. We further study two research questions: *what causes the performance drop in non-English languages?* (Sec 5.1) and *how schema augmentation SAVE improves the model?* (Sec 5.2).

Our contributions can be summarized as follows:

- To our best knowledge, MULTISPIDER is the *largest* multilingual text-to-SQL semantic parsing dataset with seven languages.
- We further identify lexical challenge and structure challenge of multilingual text-to-SQL brought by specific language properties.
- We propose a simple-yet-effective data-augmentation method SAVE from the perspective of schema.
- Experimental results reveal that MULTISPIDER is indeed challenging and SAVE significantly boosts the overall performance by about 1.8%.

2 The MULTISPIDER Dataset

2.1 Dataset Collection and Statistic

We build MULTISPIDER based on Spider (Yu et al. 2018), a large-scale cross-database text-to-SQL dataset in English. Only 9691 questions and 5263 SQL queries over 166

Type	Schema	Mistake	Correction
Abbreviation	aid	援助 (assistance)	作者ID (ID of the author)
	did	做了 (done)	领域ID (ID of the domain)
Jargon	body builder	造车者 (carmakers)	健美运动员 (muscle-builder)
	snatch	抢夺 (wrest)	挺举 (weightlifting)
Polysemy	player	演员 (actor)	运动员 (athlete)
Inaccurate Translation (Question)			
Lexical	Spider: What capital is the largest in the us? (DB: Geo)		
	CSpider: 美国最大的资本是什么? (money)		
	MultiSpider: 美国最大的州会是什么? (metropolis)		
Structural	Spider: List names of conductors in descending order of years of work. SQL: SELECT Name FROM conductor ORDER BY Year_of_Work DESC		
	Google: コンダクターの名前と降順での勤務年数を示す? (List both name and year)		
	MultiSpider: 勤務年数の降順での指揮者の名前は? (List only name)		

Figure 2: Typical mistakes during the translation, due to the lack of context information and domain knowledge. The correct translation and their explanations from WordNet.

databases (train-set and dev-set) are publicly available. Thus we only translate those data.

Currently, there are two well-known extensions of Spider: (1) CSpider (Min and Zhang 2019) (Chinese, schema kept in English): we improve the existing translation and translate the schema as well. (2) VSpider (Tuan Nguyen, Dao, and Nguyen 2020) (Vietnamese): we re-partition the dataset to be the same as other languages for fair comparison. The mentioned value (e.g., location and name) in question are kept in English, to be consistent with the database content.

2.2 Challenge of Dataset Translation.

Based on our preliminary study, we summarize five typical mistakes during translating the text-to-SQL dataset including schema and question (Figure 2).

Challenge of Schema Translation Both insufficient context and domain knowledge make the schema translation challenges, including abbreviation, domain-specific jargon, and polysemy. For example, AID could be interpreted as ‘assistance’ or ‘id_of_the_author’ (Figure 2). We could disambiguate the meaning of schema headers by referring to its content value, neighbor headers, and involved question. Thus we can recognize AID as the abbreviation of ‘id_of_the_author’ by examining its value ‘0001’, neighbor ‘publisher’, and the question ‘Return the aid of the best paper?’.

Challenge of Question Translation We are facing two challenges here: (1) lexical challenge refers to the entity polysemy, such as the ‘capital’ in the case of Figure 2. It’s not easy to deduce the actual meaning of ‘capital’ (‘money’ or ‘metropolis’) simply based on the context of the question, but we could disambiguate its meaning by schema translation of ‘capital’ where domain knowledge is considered; (2) structural challenge points out that the complex logic or syntactic structure causes inaccurate translation. We propose to refer to the corresponding SQL query to validate the logic. For example, as shown in the last line of Figure 2, the machine translation might generate redundancy headers ‘year’.

Lexical Challenge	Explanation
Question: 有多少不同的获胜者都参加了 “wta championships”,并且都是左撇子? (How many different winners both participated in the WTA Championships and were left-handed?) Gold: SELECT count(DISTINCT winner_name) FROM matches WHERE tourney_name = 'WTA Championships' AND winner_hand = 'L'	Mention: 左撇子 Schema: 慣用手 (Slang)
Question: 最小バージョン番号とそのテンプレートタイプコードは? (What the smallest version number and its template type code?) Gold: SELECT min(Version_Number), template_type_code FROM Templates	Mention: バージョン番号 Schema: バージョナンバー (Hiragana and Katakana)
Question: 每个国家中的被最多人讲的主流语言是什么? (What is the language spoken by the largest percentage of people in each country?) Gold: SELECT Language, CountryCode, max(Percentage) FROM countrylanguage GROUP BY CountryCode	Mention: 主流 Schema: 百分比 (Semantic Match)
Structural Challenge	Explanation
Question: 按照从老到少的顺序输出老师的姓名? (List the names of teachers in ascending order of age.) Gold: SELECT Name FROM teacher ORDER BY Age ASC	Mention: 从老到少 Operator: ORDER BY Age ASC (Dialect)
Question: 成績証明書のリリースの最も早い日付は何ですか? 詳細を教えてください (What is the earliest date of a transcript release, and what details can you tell me?) Gold: SELECT transcript_date, other_details FROM Transcripts ORDER BY transcript_date ASC LIMIT 1	Mention: 最も早い日 Operator: ORDER BY Date ASC (Commonsense)

Figure 3: The lexical and structural challenge are further enhanced in MULTISPIDER due to specific language properties.

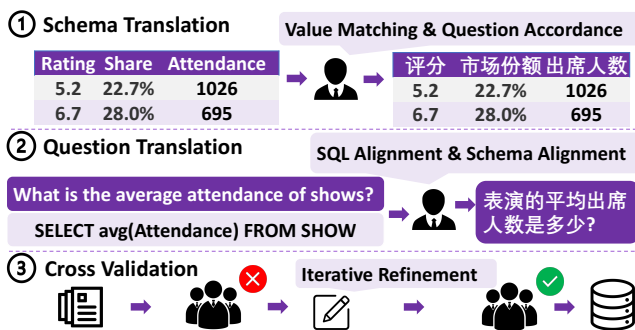


Figure 4: The translation pipeline of MULTISPIDER.

2.3 Translation Pipeline

Hiring Qualified Translators The translators are college students who majored in the target language¹. There are three students for each language (15 students in total) who are proficient in English (e.g. IELTS ≥ 7.0) and also meet the criteria: (1) language certificate of the target language, i.e. TEF/TCF for French; or (2) lived abroad for years.

Translation and Validation To be effective, we first use Google NMT to translate the Spider, then let each translator post-edit the translation individually. According to the preliminary study about translation mistakes in Sec 2.2, the translation pipeline is organized as three steps (Figure 4): (1) schema translation to let the translators leverage the content values of the corresponding schema, the neighbor headers, and the involved questions, to obtain sufficient context information of schema; (2) question translation by referring to the translated schema and translating the corresponding SQL simultaneously to valid the complex logic of the sentence; (3) cross validation to merge the annotated data through voting the best translations among three annotators.

¹The payment of translators is listed in Ethical Statement.

	DE	ES	FR	JA	ZH
Question	4,607	3,567	4,723	4,092	989
Column	1,248	682	1,382	1,601	1,469
Table	362	225	327	470	670

Table 1: The statistics of post-editing data for each language. ZH starts from CSpider while the others are translated from Google NMT.

2.4 Dataset Analysis

High Quality Although Google Translation reveals the excellent performance, the annotators further improve the data via post-editing about 37.1% questions and 27.3% schema as shown in Table 1. For each language, we spent more than 200 hours on translation and data review. In this way, the inter-agreement of annotators reaches 92.7%².

More Challenging Text-to-SQL usually tackles two kinds of challenges (Herzig and Berant 2018), i.e. the *structural challenge* (mapping the mentions to schema) and the *lexical challenge* (mapping the intentions to SQL operators). As illustrated in Figure 3, MULTISPIDER poses both the lexical and structural challenge in the context of multilingual: (1) the specific language properties like Hiragana and Katakana (Japanese) and morphologically rich language (German and French) would make the lexical challenge more difficult by expanding the syntactic difference between schema and tokens; (2) the translation question which involved dialect sayings requires further commonsense reasoning to address the structural challenge (Figure 3). Thus, MULTISPIDER is also challenging in multilingual besides high-quality.

²The inter-annotator agreement is calculated as the percentage of overlapping votes.

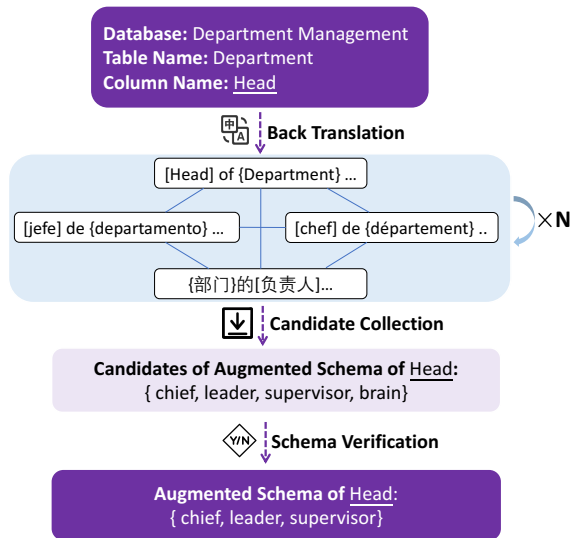


Figure 5: The pipeline of Schema Augmentation.

3 Schema Augmentation: SAVE

Lexical challenge becomes more severe in multilingual settings due to different language properties (Figure 3). To address this problem, we propose SAVE (Schema-Augmentation-with-Verification) to generate more schema variations, to improve the grounding ability of the parser³.

Specifically, we first adopt machine-translation to generate the synonym candidates of schemas by multi-rounds back-translation. Then we use natural-language-inference model to select the semantic equivalency candidates, via measuring the entailment scores between schema and candidate (to ensure the data quality). Eventually, the augmented schemas would be used to expand the training data.

3.1 Augmentation Pipeline

Back Translation generates the synonym candidates of schema (e.g. CHIEF and BRAIN are the candidates of HEAD). At first, to leverage the context of the schema for a better translation, we design a special template to insert the information of the database and the affiliated table like $[COLUMN]$ of $\{TABLE\}$ from $(DATABASE_NAME)$. Then we translate this template from the target language into K intermediate languages. To further improve the candidate diversity, N rounds of translation are conducted between intermediate language and target language alternatively. Finally, we obtain $K * N$ synonym candidates (duplicate exists) in the target language⁴.

³We choose schema augmentation rather than question augmentation since it's more efficient. The augment coverage of a single table modification includes all affiliated SQLs.

⁴The back-translation would run 3 turns among 11 languages (seven languages of MULTISPIDER plus Russian, Portuguese, Dutch, Swedish), i.e. $K = 11$, $N = 3$. The extra four languages are decided by their translation performance and the scale of their training corpus as reported in M2M100 paper.

Schema Verification We propose to measure the semantic equivalence between the original schema and the candidate synonym to collect the suitable candidates inspired by Pi et al. (2022). The main challenge in schema verification is to compute the similarity of contextualized schema (*head of department vs. brain of department*). Hill, Reichart, and Korhonen (2015) shows that natural language inference (NLI) model achieves promising performance (70% accuracy) than baselines (Word2Vec (Mikolov et al. 2013) and Glove (Pennington, Socher, and Manning 2014)) in computing semantic equivalency. Thus, NLI model is a good choice to collect the synonym of schema via enumerating the candidates. Concretely, we design a template to construct hypotheses and premise using schema and candidate as input. The template is $TABLE [COLUMN] (TYPE)$ which contextualizes the schema with table context. Finally, we compute the entailment scores from both directions (*premise to hypothesis* and *hypothesis to premise*), as the judgment of semantic similarity. If they are both above the threshold (0.68 for Chinese and 0.65 for others), we select this pair as augmented schema data.

3.2 Quality of Augmented Schema

To examine the effectiveness of schema verification, we conduct the human-evaluation of the augmented schema. Concretely, we sample 300 schemas from each language respectively. The accuracy (i.e., the percentage of semantic equivalent items) is about 74.5% with verification and drops drastically to 33.2% without verification.

3.3 Synthesising New Training Data

Text-to-SQL data example consists of three parts: question, schema and SQL. To expand the training corpus, for each data example, we randomly replace the schema items (e.g. COLUMN or TABLE) with the corresponding augmented schemas (e.g., replace HEAD with CHIEF in the above case) to compound the new training data examples. Consequently, we expand the training data by two to three times.

4 Experiments

4.1 Experimental Setup

Baseline Models We choose two types of representative models: (1) task-specific model RAT-SQL (Wang et al. 2020), equipped with pretrained multilingual encoder mBERT (Devlin et al. 2019) and XLM-Roberta-Large (Conneau et al. 2020); (2) pretrained multilingual encoder-decoder mBART (Liu et al. 2020) which is inspired by the recent work of Scholak, Schucher, and Bahdanau (2021) that reveals the excellent performance of pretrained encoder-decoder model⁵.

Evaluation Metric We report results using the same metrics as (Wang et al. 2020): exact match accuracy on all examples, as well as divided by difficulty levels determined by the official evaluation script (Yu et al. 2018).

Training with Augmented Data During the training phase, we first adopt the augmented data to warm up the model three epochs to alleviate the noise in augmented data, then fine-tune the model with original high-quality training data.

⁵Code available at <https://github.com/microsoft/ContextualSP>

Model	EN	DE	ES	FR	JA	ZH	VI	AVG(6 langs)
<i>Monolingual Training (only use target language training data)</i>								
mBART	57.3	39.7	41.3	37.5	45.7	55.0	42.2	43.6
mBART + SAVE	58.3	42.6	42.6	51.2	46.9	56.6	43.1	45.5 (+1.9%)
RAT-SQL + XLM-R	68.6	62.5	61.7	64.1	53.1	63.4	65.9	61.8
RAT-SQL + XLM-R + SAVE	68.8	63.9	62.7	65.7	54.3	66.2	66.1	63.2 (+1.4%)
<i>Multilingual Training (use training data from multiple languages)</i>								
mBART	58.3	42.7	45.9	42.9	52.2	57.8	43.2	47.5
mBART + SAVE	59.7	46.9	47.1	43.0	54.3	61.9	45.6	49.8 (+2.3%)
RAT-SQL + XLM-R	68.8	64.8	67.4	65.3	60.2	66.1	67.1	65.2
RAT-SQL + XLM-R + SAVE	70.8	66.7	69.3	67.5	61.6	67.3	67.8	66.7 (+1.5%)

Table 2: Exact-match Accuracy on MULTISPIDER for 7 languages. Notice that the AVG is calculated across 6 non-English languages to be comparable to English results. The performance boosts brought by SAVE are bolded.

4.2 Experimental Results

Follow the popular multilingual datasets MTOP (Li et al. 2021) and MultiATIS++ (Xu, Haider, and Mansour 2020), we conduct extensive experiments under three settings: zero-shot, monolingual and multilingual. The results demonstrate that (1) the absolute drop of accuracy in non-English languages is about 6.1% on average; (2) SAVE significantly improves the performance about 1.8% overall.

Model	DE	ES	FR	JA	ZH	VI
<i>Directly Predict</i>						
mBERT	50.9	52.2	50.7	43.1	49.6	45.3
XLM-R	57.6	60.8	59.1	48.3	55.5	56.5
<i>Translate-then-Predict</i>						
mBERT	49.6	51.2	47.6	39.1	46.7	43.3
XLM-R	58.8	57.2	58.7	46.3	55.3	53.8
<i>Translate-then-Train</i>						
mBERT	49.5	51.2	51.3	38.2	45.8	49.3
XLM-R	60.2	61.9	61.7	51.3	57.6	63.9

Table 3: Exact-match Accuracy under zero-shot settings.

Zero-shot Transfer Zero-shot transfer is a realistic scenario where only the English training dataset is available. We study three fine-grained zero-shot settings:

- **Directly Predict:** The parser is trained on English. During the inference, we directly predict with the question and schema in the target-language.
- **Translate-then-Predict:** The parser is trained on English. During the inference, we first translate the input question and schema from the target-language into English using Google NMT and then predict it.
- **Translate-then-Train:** We first translate the original English dataset into the target language, then train the parser on this machine-translated training dataset.

From Table 3, we observed that (1) the performance of zero-shot transfer largely depends on the choice of pre-trained encoder, where a better model enables better zero-shot transfer, i.e. XLM-R-Large beats mBERT by a large margin; (2) compared with translation-then-test, directly predict receives better performance about 1.6% since machine-translation might create mistakes, especially for schema translation; (3) with strong pretrained language model and machine-translation model, we could receive the promising results, which reveals that machine-translated data could be an economical proxy of human-translated data as Sherborne, Xu, and Lapata (2020).

Monolingual Training In this setting, the parser is trained on the human-translated training dataset in the target-language. From the results of the upper half of Table 2, we observed that (1) The performance of Japanese is significantly behind other languages. It’s mainly caused by Hiragana and Katakana, which will be further analyzed in Sec 5.1; (2) BART exhibits strong performance in English and Chinese compared with the task-specific model, indicating the potential growth of pretrained seq2seq model in text-to-SQL; (3) SAVE significantly improve the non-English languages (1.4%-1.9%) but raised less performance in English (0.2%-1.0%). We found that the most data pairs (schema and mention) in English are exactly/partly match (Gan et al. 2021), which is much easily than other languages so that it would benefit less from SAVE.

Multilingual Training In this setting, the parser is trained on the concatenation of training data from all languages. From the results of the bottom half of Table 2, we observed that (1) the multilingual training receives the best results overall. mBART and RAT-SQL receive a performance boost of about 3.9% from multilingual training in all languages; (2) English still benefits from multilingual training which is also proved by other multilingual datasets (Xu, Haider, and Mansour 2020; Li et al. 2021); (3) Notably, SAVE would improve the model further by 1.5%, indicating the effectiveness of data augmentation.

Lexical Mistake	Explanation
Question (ZH): 4缸以上の汽车数量是多少？ (What is the number of cars with more than 4 cylinders?) Gold: SELECT Count(*) FROM cars_data WHERE cylinders > 4 Pred: SELECT Count(*) FROM cars_data WHERE weight > 4	Mention: 4缸 Schema: 气缸数 (cylinders)
Question (JA): 「English」を話さず、政府の形態が「republic」でない国の国コードは何ですか？ (What are the codes of the countries that do not speak English and whose government forms are not Republic?) Gold: SELECT Code FROM country WHERE GovernmentForm != "Republic" EXCEPT SELECT CountryCode FROM countrylanguage WHERE LANGUAGE = "English" Pred: SELECT Code FROM country WHERE countrycode != "Republic" EXCEPT SELECT CountryCode FROM countrylanguage WHERE LANGUAGE = "English"	Mention: 政府の形態 Schema: 政府のフォーム (GovernmentForm)
Question (DE): Wie lauten Bevölkerung, Name und Führer des Landes mit der größten Fläche? (What are the population, name and leader of the country with the largest area?) Gold: SELECT Name , population , HeadOfState FROM country ORDER BY SurfaceArea DESC LIMIT 1 Pred: SELECT Name , population , GovernmentForm FROM country ORDER BY SurfaceArea DESC LIMIT 1	Mention: Führer des Landes Schema: Staatsoberhaupt (head_of_state)
Question (FR): Quel est le modèle de voiture avec le mpg le plus élevé? (What is the car model with the highest mpg?) Gold: SELECT model from car_names JOIN cars_data order by mpg DESC LIMIT 1 Pred: SELECT maker from car_names JOIN cars_data order by mpg DESC LIMIT 1	Mention: modèle Schema: maquette (model)
Structural Mistake	Explanation
Question (ZH): 最年轻的狗有多重？ (How much does the youngest dog weigh?) Gold: SELECT weight FROM Pets ORDER BY pet_age Asc LIMIT 1 Pred: SELECT weight FROM Pets ORDER BY pet_age Desc LIMIT 1	Mention: 年轻 SQL Operator: ORDER BY pet_age Asc
Question (JA): 最も燃費が良いのはどのモデルですか？すなわち、mpgが一番高い車種は何ですか？ (Which model saves the most gasoline? That is to say, have the maximum miles per gallon.) Gold: SELECT Model FROM car_names JOIN cars_data ORDER BY mpg DESC LIMIT 1 Pred: SELECT Model FROM car_names JOIN cars_data ORDER BY horsepower DESC LIMIT 1	Mention: 最も燃費が良い SQL Operator: ORDER BY mpg DESC
Question (ZH): 哪些城市有多于一个未满30岁的员工？ (Which cities do more than one employee under age 30 come from?) Gold: SELECT City FROM employee WHERE Age < 30 GROUP BY City HAVING Count(*) > 1 Pred: SELECT City FROM employee WHERE Age = 30 GROUP BY City HAVING Count(*) > 1	Mention: 未满30岁 SQL Operator: Age < 30

Figure 6: Case studies of non-English languages under two categories: lexical mistakes and structural mistakes.

5 Discussion and Analysis

5.1 What Causes the Performance Drop in Non-English Languages?

In this section, we conduct both qualitative analysis and quantitative analysis about the accuracy drop in non-English languages (Sec 4.2). Concretely, we conduct case studies (Figure 6) for incorrect SQL prediction in non-English compared to the correct SQL prediction in English. All these SQL are predicted by RAT-SQL+XML-R+SAVE under multilingual settings, which is the SOTA model in experiments (Table 2). Furthermore, we divide these bad cases into two categories: lexical mistakes and structural mistakes.

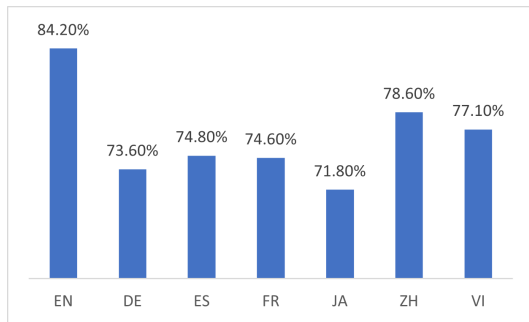


Figure 7: Fuzzy-match based schema-linking score.

Lexical Mistake refers that the schema has not been grounded in SQL, which is usually caused by the syntactic difference between schema and tokens also known as schema-linking problem (Wang et al. 2020; Lei et al. 2020). (1) Qualitative analysis (Figure 6) reveals that the specific language properties like Slang (Chinese), Hiragana and Katakana (Japanese) and morphologically rich language (German and French) would expand the syntactic difference between schema and tokens then make the lexical challenge more difficult. In comparison, the original English Spider employ the similar surface form between entity mention and schema. (2) Quantitative analysis (Figure 7) computes the fuzzy-match-based score between question and schema, which is usually employed by popular model (Wang et al. 2020; Guo et al. 2019), indicating that schema-linking becomes more challenging for non-English.

Structural Mistake refers to the incorrect prediction of SQL operators. The models are acquired to leverage the commonsense reasoning ability to match the SQL spans with intent mentions. However compared with the English, MULTISPIDER contains more dialect sayings in question annotation. In the last case of Figure 6, it's difficult to deduce the actual meanings of the expression 'Age < 30' in Chinese.

In summary, both *specific language properties* and the *dialect sayings* lead to the performance drop in non-English languages, which also makes MULTISPIDER more challenging in multilingual.

Schema	Synonyms
total spent	total expenditure total spending total consumption
收益	获利 利润 益处 收入
上級者	最高 高級者 優秀 トップ
Schema	Morphological Variants
donator name	name of the donor name of donor the donor name
销售额	销售 销售量 出货量 销售额的数量 销售金额
総乗客数	乗客の総数 乗客総数

Figure 8: Two categories of augmented schema.

5.2 How Schema Augmentation SAVE Improves the Model?

Sec 4.2 demonstrates that SAVE significantly improves the performance by about 1.8% overall in all languages across three settings. The performance gain of augmented data might come from two aspects: (1) addressing the lexical challenge by synthesizing more schema-token pairs; (2) improving the robustness of the text-to-SQL model through varies the schema input as studied by (Pi et al. 2022).

We conduct both qualitative and quantitative analysis on augmented schema to understand the reason of performance gain. For qualitative analysis, after conducting cases studies on seven languages, we roughly classify the augmented schema items into two categories (Figure 8): *synonyms* which is semantically identical with the original schema but with different lemmas (i.e. don't have string overlap); and *morphological variants* that changes the forms of schema syntactically. For quantitative analysis, we sample 500 schemas from each language respectively, and we found that (1) for DE and ES, the most augmented schema (over 70%) are morphological variants (2) for JA and ZH, it usually generate the synonyms.

6 Related Work

6.1 Multilingual Text-to-SQL Datasets

The recent development of text-to-SQL is largely driven by the large-scale annotation datasets. These corpora cover a wide range of settings: single-table (Zhong, Xiong, and Socher 2017), multi-table (Yu et al. 2018), multi-turn (Yu et al. 2019a,b). There are also a few non-English text-to-SQL datasets (Min and Zhang 2019; Tuan Nguyen, Dao, and Nguyen 2020; Guo et al. 2021; José and Cozman 2021).

However, all these multilingual text-to-SQL datasets only support three languages. The language coverage is limited compared with other multilingual datasets. For example, the multilingual task-oriented dialogue dataset MTOP (Li et al. 2021) and MultiATIS++ (Xu, Haider, and Mansour 2020) support six languages and nine languages respectively. Therefore, to advance the research on multilingual text-to-SQL, we propose MULTISPIDER covering seven mainstream languages and quite challenging.

6.2 Multilingual Text-to-SQL Systems

Driven by the large-scale English text-to-SQL dataset, many powerful task-specific model have been proposes for text-to-SQL, including effective input encoding (Wang et al. 2020), intermediate representation of SQL (Guo et al. 2019) and grammar-based decoding for valid SQL (Yin and Neubig 2018). Among a wide range of fancy models, RAT-SQL (Wang et al. 2020) is the most popular one which attracts a lot of attention from the research community and industry. Specifically, it adopts the relation-aware transformer to learn the joint representation of database and question, and achieves the promising results.

For non-English text-to-SQL, previous work (Min and Zhang 2019; Tuan Nguyen, Dao, and Nguyen 2020) typically adopts language-specific tokenizer or pretrained language model like PhoBERT for Vietnamese (Nguyen and Nguyen 2020), to extend the English parser for multilingual scenario. Therefore, we adopt the RAT-SQL with multilingual encoder like multilingual-BERT (Devlin et al. 2019) and XLM-R (Conneau et al. 2020) as our main baseline models.

Besides the task-specific approaches, there is also another research trend that using the pretrained encoder-decoder models to track with the text-to-SQL. It attempts to formula the text-to-SQL parsing tasks as seq2seq translation task. Recently, researchers have developed lots of powerful parsers (Scholak, Schucher, and Bahdanau 2021; Shin et al. 2021) built on the top of pretrained language models like BART (Liu et al. 2020) and T5 (Raffel et al. 2020). Thus, we attempt to choose mBART (Liu et al. 2020), a multilingual pretrained encoder-decoder model, as another baseline model.

7 Conclusion and Future Work

Most existing work on text-to-SQL are centered on English, excluding the powerful interaction technique's accessibility to non-English speakers. In this paper, we present the *largest* dataset MULTISPIDER covering seven mainstream languages to promote the research on multilingual text-to-SQL. We ensure the dataset quality by hiring sufficient qualified translators and multi-rounds checking. The results MULTISPIDER is natural, accurate and also challenging in terms of text-to-SQL. We further explore the lexical challenge and structural challenge in multilingual text-to-SQL and find that language-specific properties would make these two challenges more difficult. Therefore, we propose a simple and generic schema-augmentation method SAVE to expand the size of training data. Extensive experiments verify the effectiveness of SAVE, which boosts the model performance by about 1.8%. We propose a series of popular baseline methods and conduct extensive experiments on MULTISPIDER to encourage future research for multilingual text-to-SQL systems.

Future work would include (1) developing a multilingual text-to-SQL system and apply it in the real globalization scenario; (2) leveraging better pretrained model and advancing architecture design to address the lexical challenge and structure challenge in multilingual settings; (3) expanding SAVE to other table-related task (Wenhu Chen 2020) and further improve the schema verification accuracy.

Ethical Statement

This work presents MULTISPIDER, a free and open dataset for the research community to study the multilingual text-to-SQL problem. Data in MULTISPIDER are collected from Spider (Yu et al. 2018), a free and open cross-database English text-to-SQL dataset. We also collect data from the CSpider (Min and Zhang 2019) and VSpider (Tuan Nguyen, Dao, and Nguyen 2020), which are also free and open text-to-SQL dataset. To annotate the MULTISPIDER, we recruit 15 Chinese college students (8 females and 7 males). Each student is paid 2 yuan (\$0.3 USD) for translating the schema or questions. This compensation is determined according to prior work on similar dataset construction (Guo et al. 2021). Since all question sequences are collected against open-access databases, there is no privacy issue. The details of our data collection and characteristics are introduced in Section 2.

Acknowledgements

We thank all anonymous reviewers for their constructive comments. Wanxiang Che was supported via the grant 2020AAA0106501 and NSFC grants 62236004 and 61976072. Dechen Zhan is the corresponding author.

References

- Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.
- Gan, Y.; Chen, X.; Huang, Q.; Purver, M.; Woodward, J. R.; Xie, J.; and Huang, P. 2021. Towards Robustness of Text-to-SQL Models against Synonym Substitution. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2505–2515. Online: Association for Computational Linguistics.
- Guo, J.; Si, Z.; Wang, Y.; Liu, Q.; Fan, M.; Lou, J.-G.; Yang, Z.; and Liu, T. 2021. Chase: A Large-Scale and Pragmatic Chinese Dataset for Cross-Database Context-Dependent Text-to-SQL. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2316–2331.
- Guo, J.; Zhan, Z.; Gao, Y.; Xiao, Y.; Lou, J.-G.; Liu, T.; and Zhang, D. 2019. Towards Complex Text-to-SQL in Cross-Domain Database with Intermediate Representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4524–4535.
- Herzig, J.; and Berant, J. 2018. Decoupling Structure and Lexicon for Zero-Shot Semantic Parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1619–1629.
- Hill, F.; Reichart, R.; and Korhonen, A. 2015. SimLex-999: Evaluating Semantic Models With (Genuine) Similarity Estimation. *Computational Linguistics*, 665–695.
- José, M. A.; and Cozman, F. G. 2021. mRAT-SQL+GAP: A Portuguese Text-to-SQL Transformer. *CoRR*.
- Lei, W.; Wang, W.; Ma, Z.; Gan, T.; Lu, W.; Kan, M.-Y.; and Chua, T.-S. 2020. Re-examining the Role of Schema Linking in Text-to-SQL. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6943–6954.
- Li, H.; Arora, A.; Chen, S.; Gupta, A.; Gupta, S.; and Mehdad, Y. 2021. MTOP: A Comprehensive Multilingual Task-Oriented Semantic Parsing Benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2950–2962.
- Liu, Y.; Gu, J.; Goyal, N.; Li, X.; Edunov, S.; Ghazvininejad, M.; Lewis, M.; and Zettlemoyer, L. 2020. Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 726–742.
- Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient Estimation of Word Representations in Vector Space. *Computer Science*.
- Min, Y., Qingkai et aland Shi; and Zhang, Y. 2019. A Pilot Study for Chinese SQL Semantic Parsing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3652–3658.
- Nguyen, D. Q.; and Nguyen, A. T. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 1037–1042.
- Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.
- Pi, X.; Wang, B.; Gao, Y.; Guo, J.; Li, Z.; and Lou, J.-G. 2022. Towards Robustness of Text-to-SQL Models Against Natural and Realistic Adversarial Table Perturbation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140): 1–67.
- Scholak, T.; Schucher, N.; and Bahdanau, D. 2021. PICARD: Parsing Incrementally for Constrained Auto-Regressive Decoding from Language Models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 9895–9901.

Sherborne, T.; Xu, Y.; and Lapata, M. 2020. Bootstrapping a Crosslingual Semantic Parser. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 499–517.

Shin, R.; Lin, C.; Thomson, S.; Chen, C.; Roy, S.; Platanios, E. A.; Pauls, A.; Klein, D.; Eisner, J.; and Van Durme, B. 2021. Constrained Language Models Yield Few-Shot Semantic Parsers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Tuan Nguyen, A. e. a.; Dao, M. H.; and Nguyen, D. Q. 2020. A Pilot Study of Text-to-SQL Semantic Parsing for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4079–4085.

Wang, B.; Shin, R.; Liu, X.; Polozov, O.; and Richardson, M. 2020. RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7567–7578.

Wenhu Chen, e., Hongmin Wang. 2020. TabFact : A Large-scale Dataset for Table-based Fact Verification. In *ICLR*.

Xu, W.; Haider, B.; and Mansour, S. 2020. End-to-End Slot Alignment and Recognition for Cross-Lingual NLU. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5052–5063.

Yin, P.; and Neubig, G. 2018. TRANX: A Transition-based Neural Abstract Syntax Parser for Semantic Parsing and Code Generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 7–12. Brussels, Belgium: Association for Computational Linguistics.

Yu, T.; Zhang, R.; Er, H.; Li, S.; Xue, E.; Pang, B.; Lin, X. V.; Tan, Y. C.; Shi, T.; Li, Z.; Jiang, Y.; Yasunaga, M.; Shim, S.; Chen, T.; Fabbri, A.; Li, Z.; Chen, L.; Zhang, Y.; Dixit, S.; Zhang, V.; Xiong, C.; Socher, R.; Lasecki, W.; and Radev, D. 2019a. CoSQL: A Conversational Text-to-SQL Challenge Towards Cross-Domain Natural Language Interfaces to Databases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1962–1979.

Yu, T.; Zhang, R.; Yang, K.; Yasunaga, M.; Wang, D.; Li, Z.; Ma, J.; Li, I.; Yao, Q.; Roman, S.; Zhang, Z.; and Radev, D. 2018. 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*, 3911–3921.

Yu, T.; Zhang, R.; Yasunaga, M.; Tan, Y. C.; Lin, X. V.; Li, S.; Er, H.; Li, I.; Pang, B.; Chen, T.; Ji, E.; Dixit, S.; Proctor, D.; Shim, S.; Kraft, J.; Zhang, V.; Xiong, C.; Socher, R.; and Radev, D. 2019b. SPaC: Cross-Domain Semantic Parsing in Context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4511–4523.

Zhong, V.; Xiong, C.; and Socher, R. 2017. Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning. *CoRR*.