Computer Fraud and Security
ISSN (online): 1873-7056

A Transfer Learning Based Approach to Automatic SQL Translation

Wensheng Tang[1], Tian Tian[2,*], Hongcheng Jiang[3], Zesan Liu[4], Hong Lin[2], Yifan Hu[3]
[1]*Marketing Department of State Grid Corporation of China, Beijing 100100, China*
[2]*Beijing China-Power Information Technology Co., Ltd., Beijing 100192, China*
[3]*Information Service Center, Info and Comm Branch of State Grid Zhejiang Electric Power Co., Ltd. Hangzhou 310000, Zhejiang, China*
[4]*Information Industry Research Institute, State Grid Info & Telecom Group Co., Ltd, Beijing 100052, China*
**Corresponding Author.*

Abstract:
Limitations of Oracle database in business support, development, operation, and maintenance hinder marketing 2.0's growth. Translating Oracle SQL statements to domestic databases is vital. A neural machine translation model based on migration learning resolves the lack of parallel corpus between Oracle and domestic databases. Encoder and decoder parameters of both databases' models are initialized using a trained Oracle database-side encoder and domestic database-side decoder. Fine-tuning and optimization produce high-quality parameters, improving translation performance. Results showed that the bilingual evaluation understudy (BLEU) of model reached 31.20 and an execution accuracy value of 84.3, outperforming the Transformer model by 13.28 and 13.3, respectively. Demonstrate support for database migration.

Keywords: transfer learning, oracle database, SQL, attention mechanisms, Seq2Seq

INTRODUCTION

The Oracle database has been extensively utilized within the information systems of the State Grid. However, the evaluation of its practical application in the context of the Marketing 2.0 system has revealed significant limitations in terms of business support, system development, and operational maintenance. These limitations pose substantial obstacles to the advancement of Marketing 2.0. Concurrently, domestic databases, particularly distributed systems such as OceanBase, are maturing rapidly and offer potential solutions to the shortcomings of Oracle in this domain. Furthermore, the trend towards database localization is becoming increasingly pronounced both domestically and internationally, underscoring its importance. Therefore, the development of an automated SQL translation methodology for domestic database migration is essential. Such a method would enable seamless integration of various domestic databases into existing business center services, allowing for a transition that does not disrupt business logic or the normal operation of the current Marketing 2.0 system.

Text-Structured Query Language (SQL) translation and SQL-text translation technologies have advanced significantly, however, the neural machine translation (NMT) [1] from Oracle Database to domestic databases remains limited due to the small size of the parallel corpus and unsatisfactory translation performance. This limitation represents a significant bottleneck for the development of Marketing 2.0. Current mainstream research in machine translation utilizes end-to-end NMT based on the codec model, where an encoder transforms the original language text into a fixed-size semantic description and a decoder generates the corresponding target translation. Although codec-based NMTs exhibit strong performance in resource-rich language pairs, their effectiveness diminishes in low-resource settings, often performing worse than traditional statistical machine translation (SMT) [2]. Currently, pivot languages and migration learning are effective approaches to address the low performance of NMT in low-resource scenarios. Utiyama et al. [3] invented a translation method that relies on pivot languages, which uses resource-rich pivot languages as a link connecting the source and target languages and exploits the existence of parallel corpora of source and goal language pivot languages before training the two models separately. The advantage of this approach is that effective translation from origin to object language can be achieved using the pivot language even in low-resource scenarios where large-scale bilingual parallel corpora are lacking; however, the direct use of pivot languages as an intermediate bridge for translation results in the accumulation of errors. Unlike the previously mentioned methods, transfer learning [4] can effectively improve the number of parameters of a language model and make the model smaller. Cheng team [5] designed a transfer learning approach for pivot languages, which takes into account the relationships and differences between the three languages in model training by training the two translation models together and having them share parameters while training, which greatly reduces the parameter size. However, the step-by-step training process from the origin language to the fulcrum language and then from the fulcrum language to the final language lacks the guidance of a bilingual parallel corpus, which leads to the noise phenomenon caused by the multilingual input; moreover, the above method focuses on the improvement of the model's parameters in low-resource scenarios and does not improve the individual encoder or decoder.

In the realm of NMT for low-resource scenarios, database translation NMT presents significant challenges due to the limited availability of training data [6-7], However, the abundance of parallel corpora between Oracle database texts and national databases offers a unique opportunity to leverage migration learning approaches to enhance translation performance. This paper introduces a novel Database Translation NMT (DBNMT) model, which applies migration learning techniques to improve NMT performance for database translation tasks. Our approach incorporates the concept of pivot languages to bridge the gap caused by discrepancies in SQL usage. Specifically, we propose using a text-based pivot language to mitigate these differences. Initially, we train the encoder and decoder components using large-scale parallel corpora derived from Oracle database texts and national databases. This pre-training initializes the parameters of our NMT model, setting a solid foundation for subsequent fine-tuning. We then fine-tune the model with the smaller parallel corpora specific to Oracle and national databases to enhance translation accuracy further. Our method demonstrates a promising strategy for addressing the challenges of database translation in low-resource contexts, leveraging available data efficiently and effectively to achieve improved translation results.

In summary, we propose a migration learning method based on Seq2Seq network, which achieves excellent performance in bilingual evaluation understudy (BLEU) and execution accuracy (EX) evaluation metrics, realizes the ‘senseless’ replacement of SQL between databases, and provides the possibility of migrating between Oracle database and domestic databases.

TECHNOLOGICAL BASE

Text-SQL and SQL-Text

Text-to-SQL query generation is a critical subtask within semantic parsing, aimed at converting natural language text into formal representations such as SQL queries, logical forms, or code. This task addresses a fundamental challenge in improving database accessibility by eliminating the need for specialized SQL knowledge. The literature on Text-to-SQL generation often employs a framework based on SQL structure sketches, which consist of multiple slots that are filled through various merging techniques. This approach typically involves pointing and copying mechanisms to accommodate a diverse range of queries. Such methods are instrumental in generating precise SQL queries from natural language inputs, thereby facilitating more efficient and accessible database interactions. Resdsql [8] uses decoupled schema linking and skeleton parsing to implement text-to-sql conversions The Seq2SQ [9] approach is a framework based on augmented pointer networks that take SQL questions, table schemas, and keywords as inputs and primarily utilize the unique structure of SQL to prune the output space of the target query. By fine-tuning the SQL-specific design and domain knowledge, the medium-sized model has yielded good results. It has been the best approach in a long time.

SQL-to-Text aims to automatically generate human-readable descriptions that interpret SQL queries. This task is crucial for natural language interfaces to databases, as it facilitates understanding of complex SQL queries by non-expert users. Initially, rule-based and template-based methods were employed for SQL-to-Text generation. While these approaches require substantial manual effort to create rules or templates, they often produce rigid and formulaic text that lacks the natural flow of human language. IDP-Seq2Seq [10] is in practice that users of the model usually have a greater need for it to be able to handle complex SQL queries and provide correct translations, whereas the coverage of complex SQL in the training data may be low. To address this problem, a sequence-to-sequence (Seq2Seq) network is proposed to jointly model SQL queries and common language. The model learns the basics required for the current application scenario from simple SQL and is capable of combinatorial generalization by restructuring the basics to understand the query correctly when faced with complex SQL.

Similarities and Differences

Oracle SQL and domestic database SQL have some of their own characteristics, but because both are developed on top of the standard SQL syntax, they are still largely the same. [11] The two databases are largely the same because they are developed on top of the standard SQL syntax. In the translation process, the focus will be on different parts of the specific differences between the two.

The difference in data types such as Oracle in the numeric field unified numeric to define, the difference is the parameters behind. In the domestic database SQL, the number can have int, _double, float, decimal, numeric, and so on. Some common functions on the different functions such as in Oracle, replace the null value of the column function for all and in the domestic database SQL for nonnull and case conversion function, the former cause and lease the latter for the upper and lower, similar to such differences there are a lot of.

Some keywords or operators have different meanings. For example, in Oracle "(+)" can mean join operation, but in the domestic database SQL does not have the symbol, which must be changed to the corresponding join command. Another example is that in Oracle "rownum" can be used to hoof pick rows, while in the domestic database SQL does not have this keyword, you can only use the top part of the function instead of it, otherwise it can only be rewritten as a complex function. There are many such examples.

The programmable SQL language is different. There are many differences everywhere, from the declaration of variables, to the Data Definition Language (DDL) that controls the logic of the program, to the library functions that can be called, to the exception handling, and so on. Therefore, the translation between statements is the most difficult.

NMT Based on Transfer Learning

In NMT the encoder first processes and encodes the input sentences to produce vector representations. The decoder then uses these vectors to generate the corresponding sequence in the target language. Lakew et al. [12] proposed the use of dynamic word lists to increase the accuracy and convergence speed of the models by migrating model parameters from initial language pairs to new language pairs. Hill et al. [13] demonstrated that word vector representations obtained from an NMT encoder outperform those obtained from a monolingual (e.g., language modeling) encoder on a semantic similarity task. McCann et al. [14] utilized the attention mechanism within NMT models to contextualize word vectors, thereby enhancing both the robustness and usability of the model. Their approach demonstrates that pre-training NMT on a large corpus and subsequently transferring the learned parameters to tasks with limited data not only accelerates the training process but also improves output accuracy.

Database Interpretation Route

This method implements the functionality of the escape module, which is the middleware for SOA databases. In order to better combine information, process management components can be flexibly added to the middleware, which in turn logically manages and controls event processing and extends the whole middleware function. At the same time, the data service management of SOA database is the core content of the middleware. In the specific design, according to the data service using mechanism of UDDI, data service management is applied to extend the UDDI query function and realise the data service release and registration. As for the management of services, it is mainly through the two programmes of grid-based data service management and external database management. In external database management, each node is comprehensively analysed and data services are cached in each node, so that query functions can be achieved using the nodes and data binding can be carried out effectively. In this way, database access is dispersed among different nodes, thus buffering the access traffic of a single node, resolving the bottleneck effect during peak access periods, and ensuring the smooth flow of information.

On the basis of researching the current use of Oracle database, extract the Oracle syntax set (a collection of database syntax elements); take this as the benchmark, compare and analyze with the mainstream domestic databases; construct the maximum transferable syntax difference index system between domestic databases and Oracle; based on this syntax difference index system, design the access method of Oracle to domestic databases automatically. Based on this grammatical difference index system, we design the access method of Oracle to domestic database automatically, and realize the mutual translation and transfer between Oracle and domestic database, and the whole process is shown in Figure 1 below.

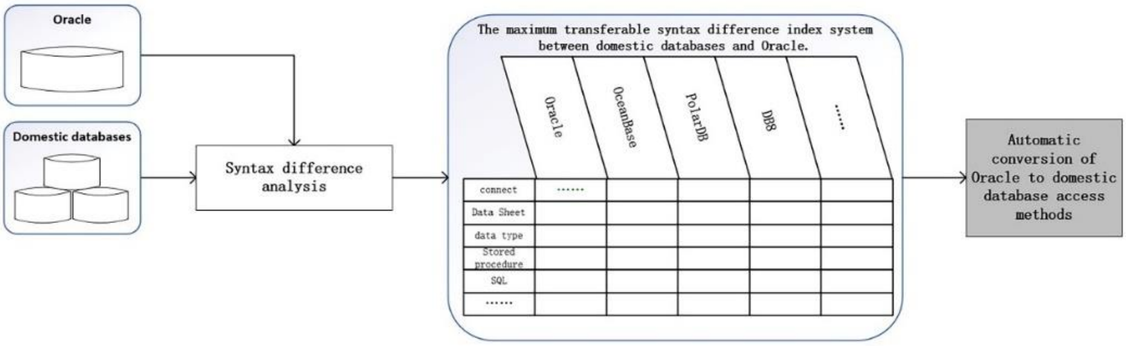

Figure 1. Schematic diagram of the intertranslation process between Oracle and domestic databases

DBNMT MODEL

In NMT, the model initially represents the source sentence as a fixed-length vector. However, this vector may not fully capture the sentence's nuances. Attention-based NMT improves this representation by dynamically retrieving information about the source language words relevant to each target word during translation. This mechanism significantly enhances the model's expressive capability and accuracy. We proposed attention mechanism based Seq2Seq [15] basis to train the translation model of Chinese-English and English-Vietnamese, and the training process is shown in Figure 2. Initially, two pre-trained models (A and B) are developed: one trained on a large corpus of domestic database SQL statement-text resources, and the other on a large corpus of text-Oracle SQL resources. Subsequently, during the training of the Oracle SQL- DBNMT model, the encoder parameters from the Oracle SQL-text model are used to initialize the parameters of the Oracle SQL-Domestic Database SQL translation model. Similarly, the decoder parameters from the Domestic Database SQL-text model are used to initialize the correspdbnmtonding parameters in the Oracle SQL-Domestic Database SQL translation model. Finally, the DBNMT model is fine-tuned using the initialized parameters and the Oracle SQL-Domestic Database SQL resource corpus to obtain the final model.

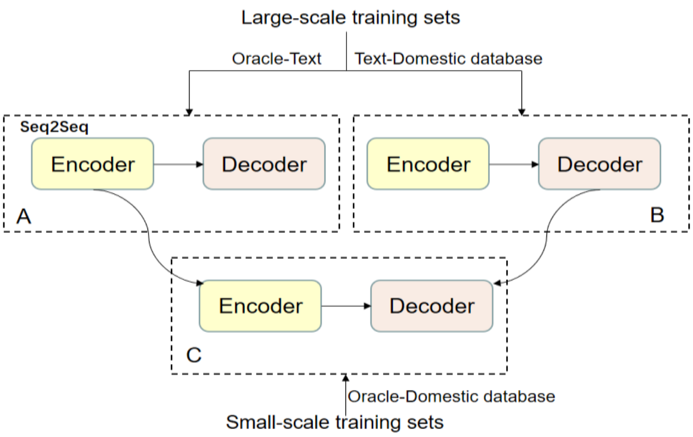

Figure 2. Flowchart of DBNMT training

In contrast to the method proposed by Zoph et al., we initialize the parameters by leveraging the encoder parameters from the SQL side of the Oracle database in the Oracle database SQL-text model, and use the parameters of the decoder from the SQL side of the domestic database in the text-domestic database SQL model for initializing both the encoder and decoder of the translation model. Subsequently, we fine-tune the model using a small-scale bilingual parallel corpus from Oracle and domestic databases to obtain the SQL DBNMT model. To enhance the relevance and integration between the pre-training components and to optimize the initialized parameters for fine-tuning, we expand the training set prior to experimentation. Initially, we back-translate text from the existing Oracle database-text and text-domestic database training sets, utilizing a large corpus of Oracle

SQL-text resources to train the Oracle SQL-Domestic database SQL translation model. This model is then used for back-translation of text within the text-domestic database SQL resource corpus. Data augmentation techniques are employed to create an expanded Oracle SQL-text-Domestic database SQL trilingual resource corpus, thereby improving model accuracy and mitigating potential errors.

SQL Translation Seq2Seq Network Structure

The Seq2Seq (Attention) model is shown in Figure 3, where an Attention layer is added on top of Seq2Seq. The Encoder and Decoder layers use Recurrent Neural Networks (RNN). In this paper, the encoder layer is a single-layer unidirectional GRU, and the Decoder layer is a single-layer unidirectional GRU. intuitively, the Attention layer is in fact similar to a very complex fully connected neural network.

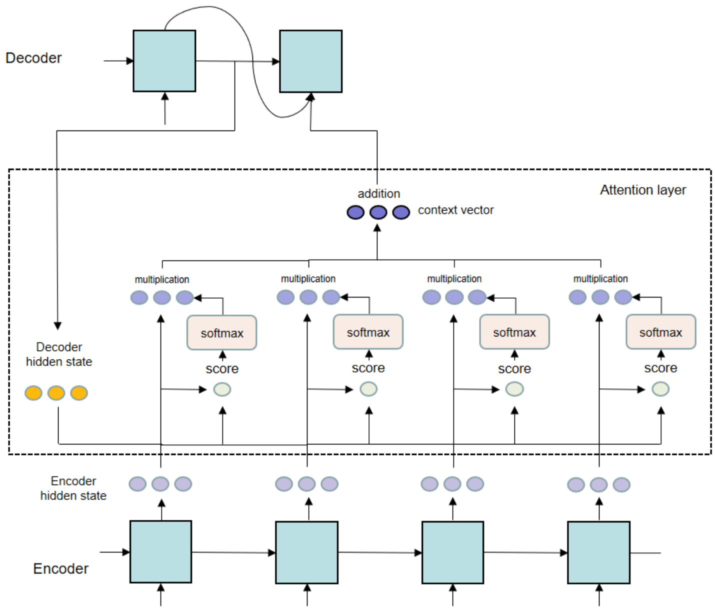

Figure 3. Seq2Seq network structure

Encoder layer input is x vector, Encoder layer is responsible for processing the input data can well capture the features of short sentences, the output sequence is [h1, h2...hm], the features are compressed to the last moment of the RNN hidden layer output is processed into s0. The attention layer processes the output of the coding layer based on the generation of [a1, a2..... am] to process the output of the encoding layer. Processes the output of the attention layer, solves for ci, and decodes it to obtain si. We use RNNs to implement the encoder and decoder of the Seq2Seq model with 128 neurons per layer.

Attention Span

In order to increase the superiority of Seq2Seq, the attention mechanism is introduced to improve the stability of the results and to let the model know in which corresponding part of the sentence to extract features. The attention mechanism allows the neural network to selectively focus on important information and ignore irrelevant information. Its core content is based on query, key and value, through the attention function after the output value of the weighted sum. In the calculation, a set of query, key, and value are generally combined into matrices Q, K, and V. The formula for the attention function is shown in equation (1).

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\big(\mathrm{sim}(Q, K)\big)V \tag{1}$$

Transformer uses scaled dot product attention, the structure of which is shown in Figure 4, and the attention scoring function uses a dot product function. In order to avoid the value of the dot product being too large, after calculating the attention scores of Q and K divide to perform scaling.

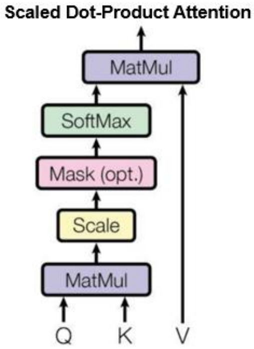

Figure 4. Scaling dot product attention

The formula for scaling dot product attention is shown in equation (2).

$$\mathrm{Attention}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)\mathrm{V} \tag{2}$$

For self-attention, Q, K, and V are all obtained from the same input sequence X (a, ..., an) multiplied by three weight matrices Wq, Wk, and Wv respectively by a linear transformation as in Eqs. (3), (4), and (5).

$$\mathrm{Q} = W_q \cdot [a_1 a_2 \dots a_i] \tag{3}$$

$$\mathrm{K} = W_k \cdot [a_1 a_2 \dots a_i] \tag{4}$$

$$\mathrm{V} = W_v \cdot [a_1 a_2 \dots a_i] \tag{5}$$

Transformer [16] model introduces a multi-head self-attention mechanism. This allows the model to learn information from different subspaces and enhances the model's expressive ability. The long self-attention mechanism is formulated as:

$$\mathrm{MultiHead}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \mathrm{Concat}(head_1, head_2, \dots, head_h)W^o \tag{6}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{7}$$

Loss Function

In this model, since the output layer adopts the Softmax activation function, the probability distribution of different classifications at each position in the sequence is obtained, so this paper adopts CrossEntropyLoss as the loss function, i.e.,cross-entropy, which mainly portrays the distance between the probability distribution of the actual output and the probability distribution of the desired output, assuming that there are predicted matching distributions of the same variable x q (x) and the target distribution p(x) two probability distributions, then the relative entropy between the two can be defined by the following:

$$D_{KL}(p \parallel q) = \sum_{N}^{i=1} p(x_i)\, log\left(\frac{p(x_i)}{q(x_i)}\right) \tag{8}$$

EXPERIMENTAL ANALYSIS

Data Sets

The dataset used in this paper is the Oracle database and domestic database SQL statement resource corpus maintained by the State Grid Cloud Platform, with 100,000 SQL corpora, including 13,000 test corpora and 10,000 validation corpora, which basically cover most of the daily SQL statements; 500,000 Oracle-text resource corpora, including 30,000 test corpora and 20,000 validation corpora; and 300,000 pairs of text-domestic database SQL corpora, including 20,000 test corpora and 10,000 validation corpora. 3000 pairs of validation corpora; and 300,000 sets of text-domestic database SQL corpora, including 20,000 sets of test corpora and 10,000 sets of validation corpora. Specific parallel corpus of Oracle database and domestic database SQL statements are divided as shown in Table 1.

Table 1. Data set

Set up	Number of SQL statements
Training	100000
Validation	3000
Test	13,000

Evaluation Indicators

We used commonly used evaluation metrics to test our approach: EX and BLEU [17]. EX is a measure of whether the result of the execution of the SQL statement matches the true reality (GT). The value of BLEU is taken between 0 and 1, and it evaluates the similarity of the generated SQL statement to the actual target SQL statement. The formula for BLEU is shown below:

$$P_n = \frac{\sum_{i}^{E}\sum_{k}^{K}\min(h_k(c_i), \min_{j\in M} h_k(s_{i,j}))}{\sum_{i}^{E}\sum_{k}^{K}\min(h_k(c_i))} \tag{9}$$

where: s_j represents the human translation, c_i represents the translated translation, and K denotes the Kth phrase.

$$BP = \begin{cases} 1 & \text{if l c} > \text{1 s} \\ e^{1-\frac{l_s}{l_c}} & \text{if 1 c} <= \text{l s} \end{cases} \tag{10}$$

BP stands for the length penalty factor. lc stands for denoting the translated length, ls stands for the effective length of correct sentence.

$$BLEU = BP \times \exp(\sum_{N}^{n=1} W_n \, log(P_n)) \tag{11}$$

While this paper provides EX evaluation, this paper considers BLUE to be a more reliable evaluation metric due to the fact that EX contains false positives, such as the existence of multiple correct SQL statements for a query. This paper follows the official evaluation protocol of Spider2.

Experimental Settings

The experiments were run on the operating system Ubuntu 18.04, using the Python 3.8 programming language and the PyTorch 1.9.0 deep learning framework. For all SQL language pairs, the Adam optimizer was used to optimize the model parameters, the $\beta_1 = 0.9$, the $\beta_2 = 0.97$.$\epsilon = 10^{-9}$. For the training of the base model, small batches were used, with each batch containing approximately 32K target language tokens and 150K training steps. For the large model, this paper uses larger batches, each containing about 460K tokens. When using large batches this paper follows the use of a cosine learning rate scheme, specifically, the learning rate ramps up linearly to le-3 in the first 10K steps and then decays at a cosine rate to le-7 in a single cycle. The large model was trained on large batches for 30K steps. To obtain the final model this paper averages 5 nearby checkpoints, which were updated at an interval of 1000 updates and 500 updates for the base and large models, respectively. For the decay schedule and base model, α=0.1, β=37.5K; for the large model, α=0.1, β=22.5K. All models were performed on 4 Nvidia GeForce 3090 GPUs for all experiments.

Comparative Tests

The model used in this paper is the DBNMT translation model based on migration learning, to test the reliability and usefulness of the model, experiments are conducted using the same dataset on the following classical models, in addition to this paper also compares the experimental results of the literature mentioned in the related studies.

(1) The seq2seq model with multi-head self-attention [18].

(2) A seq2seq model constructed as a two-layer LSTM + Attention [19].

(3) Transformer of the baseline [16] model.

(4) Resource-poor translation with cued learning. [20]

(5) Transformer+Augment modeling [21]. This model is based on the Transformer model of data augmentation.

(6) SFT for Improved Translation [22]. This model proposes a semi-precise and high-volume training method.

(7) Semantic Information Sharing for SQL Translation Transformer Model, which proposes a semantic sharing method based on the high relevance between before and after translation of SQL statements.

Table 2. Experimental results

Method	BLEU	EX
The seq2seq model with multi-head self-attention	4.89	30.3
LSTM+Attention	15.23	67.0
Base Transformer	17.92	71.0
Cued learning translation	26.77	82.7
Transformer+Augment	27.89	82.8
Scale-Trans	28.13	84.1
ours	31.06	83.8

The experimental results of each model are shown in Table 2. From the results of the experiments, it can be seen that the two-layer LSTM+Attention model has a higher BLEU value of 10.34 than that of the Transformer and seq2seq model, which proves that the introduction of a deeper network and the Attention mechanism can bring a great improvement to the model. The machine-turnover model of the benchmark Transformer has a higher BLEU value of 2.69 than that of the two-layer LSTM+Attention model, and the EX improves by 4. This is because the Transformer has a stronger sequence modeling ability and global information perception ability compared to the LSTM, and the Transformer is better able to access to key information of the sentence and is semantically better in expression is also stronger. The DBNMT model based on migration learning has a higher BLEU value of 13.14 and an improved EX of 12.8 than the benchmark Transformer, and a higher BLEU of 4.29 and an improved EX of 1.1 than Seq2Seq. The reason for such a large improvement is that migration learning can effectively avoid the shortcomings of the parallel corpus of SQL statements in the Oracle database and the SQL statements of domestic databases. Parallel corpus is less defective. In addition, the effect of the model proposed is more outstanding than that of the models proposed in other literature.

Ablation Study

To rigorously evaluate the impact of each architectural component, we conducted a series of ablation studies on our top-performing models. Table 3 provides a comparative analysis of three critical design choices within our architecture. Specifically, we investigate the effects of omitting the table and column selection components from the encoder and the implications of encoding the dependency graph problem.

Table 3. Ablation studies results

Method	EM Accuracy (%)
DBNMT	77.6±0.65
DBNMT w/o dependency grap	76.9±0.61
DBNMT w/o Part-of-Speech graph	75.5±0.36
DBNMT w/o table + column selection	73.2±0.25

Table 3 indicates that the table and column selection component has the most significant influence on the architecture. Its elimination causes a drop in Exact Match (EM) accuracy from 77.6% to 73.2%, resulting in a 4.4% absolute performance decrease. We speculate that this mechanism introduces schema-question association, which is critical in the text-to-SQL field. Thus, without schema linking, the joint contextualisation of the question and schema is absent, significantly raising the task's difficulty.

We explored the rewriting strategy, as shown in Figure 5(a), which is a statement fragment that caused the execution failure in a homegrown database. Domestic databases do not support the use of nested aggregation functions in window analyses, such as sum (sum (ws_sales_price)), sum on the inner level to calculate the sum of columns within the window, and sum on the outer level to calculate the sum of columns within a group. The statement can be rewritten by nesting subqueries so that it can be executed on a national database, as shown in Figure 5(b). The rewriting strategy of DBNMT is extensible, and new rewriting strategies are summarised based on failure examples and added to the strategy configuration file.

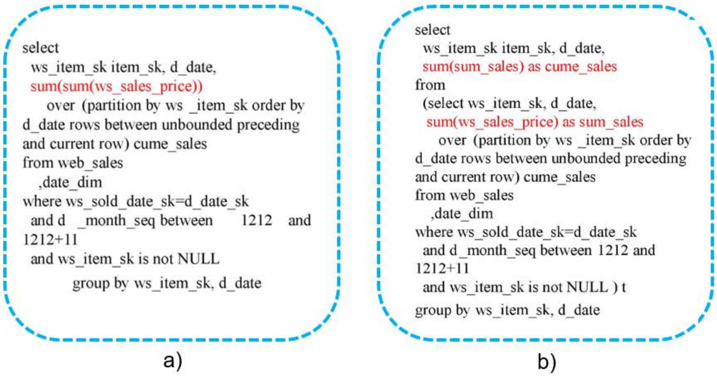

Figure 5. Statement fragment rewriting

To examine how well Transformer and Convolutional Neural Networks (CNN) encoders fare in handling this task. Table 4 shows the results of Transformer and CNN encoder processing this task using the train dataset and the test dataset. However, both of them perform worse than the original LSTM encoder. Nevertheless, Transformer showed a slight improvement over the CNN encoder. Notably, while LSTM and CNN could overfit the training dataset, the Transformer encoder did not exhibit this issue.

Table 4. Test results

Method	Train accuracy	Test accuracy
LSTM encoder	97.6%	90.4%
Transformer encoder	87.2%	86.9%
CNN encoder	99.1%	85.2%

CONCLUSIONS

The Seq2Seq machine translation model based on migration learning proposed in this paper outperforms the benchmark Transformer model as well as other models mentioned in related studies in the translation task of SQL language. The method in this paper can train the initialization parameters of the encoder and decoder of Oracle SQL-Native Database SQL NMT using Oracle SQL-Text and Text-Native Database SQL large-scale precisions and obtain the DBNMT model by fine-tuning the training with a small Oracle-Native Database corpus, which can improve the Oracle SQL-Domestic Database SQL NMT performance in low resource scenarios. Through migration learning, the model in this paper can better understand the semantic relationship between Oracle database SQL statements and domestic database SQL statements, thus improving the accuracy and fluency of translation. We continue to explore the large-scale Oracle SQL-domestic database SQL corpus for training in the subsequent research and integrate the language knowledge obtained from the training into the DBNMT model construction to improve translation accuracy.

ACKNOWLEDGMENT

This work is supported by the Science and Technology Project of State Grid Corporation of China (No. 5700-202341299A-1-1-ZN).

REFERENCES

[1] Felix Stahlberg. Neural machine translation: A review. Journal of Artificial Intelligence Research, 2020, 69: 343-418.

[2] Subrota Kumar Mondal, Haoxi Zhang, HM Dipu Kabir, et al. Machine translation and its evaluation: a study. Artificial Intelligence Review, 2023, 56(9): 10137-10226.

[3] Masao Utiyama, Eiichiro Sumita. Machine translation from Japanese and French to Vietnamese, the difference among language families, 2015 International Conference on Asian Language Processing (ialp): Ieee, 2015: 17-20.

[4] Yutong Chen, Fangyun Wei, Xiao Sun, et al. A simple multi-modality transfer learning baseline for sign language translation, Proceedings of the Ieee/cvf Conference on Computer Vision and Pattern Recognition, 2022: 5120-5130.

[5] Yong Cheng, Yong Cheng. Joint training for pivot-based neural machine translation. Joint Training for Neural Machine Translation, 2019: 41-54.

[6] Surangika Ranathunga, En-Shiun Annie Lee, Marjana Prifti Skenduli, et al. Neural machine translation for low-resource languages: A survey. Acm Computing Surveys, 2023, 55(11): 1-37.

[7] Divya Kumari, Asif Ekbal, Rejwanul Haque, et al. Reinforced nmt for sentiment and content preservation in low-resource scenario. Transactions on Asian and Low-resource Language Information Processing, 2021, 20(4): 1-27.

[8] Haoyang Li, Jing Zhang, Cuiping Li, et al. Resdsql: Decoupling schema linking and skeleton parsing for text-to-sql, Proceedings of the Aaai Conference on Artificial Intelligence, 2023: 13067-13075.

[9] George Katsogiannis-Meimarakis, Georgia Koutrika. A survey on deep learning approaches for text-to-SQL. The Vldb Journal, 2023, 32(4): 905-936.

[10] Yi-Jun Tang, Yi-He Pang, Bin Liu. IDP-Seq2Seq: identification of intrinsically disordered regions based on sequence to sequence learning. Bioinformatics, 2020, 36(21): 5177-5186.

[11] Salil Bharany, Kiranbir Kaur, Safaa Eltayeb Mohamed Eltaher, et al. A Comparative Study of Cloud Data Portability Frameworks for Analyzing Object to NoSQL Database Mapping from ONDM's Perspective. International Journal of Advanced Computer Science and Applications, 2023, 14(10).

[12] Raj Dabre, Chenhui Chu, Anoop Kunchukuttan. A survey of multilingual neural machine translation. Acm Computing Surveys (csur), 2020, 53(5): 1-38.

[13] Felix Hill, Kyunghyun Cho, Sébastien Jean, et al. The representational geometry of word meanings acquired by neural machine translation models. Machine Translation, 2017, 31: 3-18.

[14] Bryan McCann, James Bradbury, Caiming Xiong, et al. Learned in translation: Contextualized word vectors. Advances in Neural Information Processing Systems, 2017, 30.

[15] Ilya Sutskever, Oriol Vinyals, Quoc V Le. Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems, 2014, 27.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. Attention is all you need. Advances in Neural Information Processing Systems, 2017, 30.

[17] Chris Callison-Burch, Miles Osborne, Philipp Koehn. Re-evaluating the role of BLEU in machine translation research, 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006: 249-256.

[18] Elozino Egonmwan, Yllias Chali. Transformer and seq2seq model for paraphrase generation, Proceedings of the 3rd Workshop on Neural Generation and Translation, 2019: 249-255.

[19] Hao Huang, Jiannan Mao, Weike Lu, et al. DEASeq2Seq: An attention based sequence to sequence model for short-term metro passenger flow prediction within decomposition-ensemble strategy. Transportation Research Part C: Emerging Technologies, 2023, 146: 103965.

[20] Zihui Gu, Ju Fan, Nan Tang, et al. Few-shot text-to-sql translation using structure and content prompt learning. Proceedings of the Acm on Management of Data, 2023, 1(2): 1-28.

[21] Laurent Sartran, Samuel Barrett, Adhiguna Kuncoro, et al. Transformer grammars: Augmenting transformer language models with syntactic inductive biases at scale. Transactions of the Association for Computational Linguistics, 2022, 10: 1423-1439.

[22] Puneet Kumar Ojha, Abhishek Gautam, Ankit Agrahari, et al. SFT for Improved Text-to-SQL Translation. International Journal of Intelligent Systems and Applications in Engineering, 2024, 12(17s): 700-705.