PHOTON: A Robust Cross-Domain Text-to-SQL System

Jichuan Zeng[1*†] **Xi Victoria Lin**[2*] **Caiming Xiong**[2] **Richard Socher**[2]
Michael R. Lyu[1] **Irwin King**[1] **Steven C.H. Hoi**[2]
[1] The Chinese University of Hong Kong
[2] Salesforce Research
[1] {jczeng,lyu,king}@cse.cuhk.edu.hk
[2] {xilin,cxiong,rsocher,shoi}@salesforce.com

Abstract

Natural language interfaces to databases (NLIDB) democratize end user access to relational data. Due to fundamental differences between natural language communication and programming, it is common for end users to issue questions that are ambiguous to the system or fall outside the semantic scope of its underlying query language. We present PHOTON, a robust, modular, cross-domain NLIDB that can flag natural language input to which a SQL mapping cannot be immediately determined. PHOTON consists of a strong neural semantic parser (63.2% structure accuracy on the Spider dev benchmark), a human-in-the-loop question corrector, a SQL executor and a response generator. The question corrector is a discriminative neural sequence editor which detects confusion span(s) in the input question and suggests rephrasing until a translatable input is given by the user or a maximum number of iterations are conducted. Experiments on simulated data show that the proposed method effectively improves the robustness of text-to-SQL system against untranslatable user input. The live demo of our system is available at http://www.naturalsql.com.

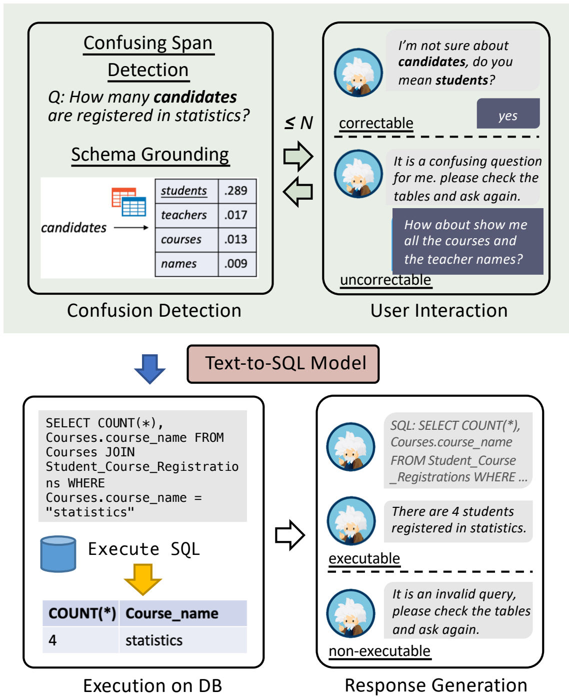


Figure 1: PHOTON workflow. The question corrector (upper block) detects the untranslatable questions from user input, scans the confusion span(s) that need clarification or correction. The accepted question is mapped into a SQL query through a text-to-SQL model, and finally the SQL execution results are returned to the user.

1 Introduction

Natural language interfaces to databases (Popescu et al., 2003; Li and Jagadish, 2014) democratize end user access to relational data and have attracted significant research attention for decades (Hemphill et al., 1990; Dahl et al., 1994; Zelle and Mooney, 1996; Popescu et al., 2003; Bertomeu et al., 2006; Zhong et al., 2017; Yu et al., 2018, 2019a). Most existing NLIDBs adopt a modular architecture consisting of rule-based natural language parsing, ambiguity detection and pragmatics modeling (Li and Jagadish, 2014; Setlur et al., 2016, 2019). While they have been shown effective in pilot study and production, rule-based approaches are limited in terms of coverage, scalability and naturalness – they are not robust against the diversity of human language expressions and are difficult to scale across domains.

Recent advances in neural natural language processing (Sutskever et al., 2014; Dong and Lapata, 2016; See et al., 2017; Liang et al., 2017; Lin et al., 2019; Bogin et al., 2019), pre-training (Devlin et al., 2019; Hwang et al., 2019), and the availability of large-scale supervised datasets (Zhong

* Equal contribution. Jichuan implemented the demo interaction flow and the neural question corrector. Victoria designed and implemented the neural semantic parser.

† Work done during internship at Salesforce Research.

et al., 2017; Finegan-Dollak et al., 2018; Yu et al., 2018, 2019b,a) enabled deep learning based approaches to significantly improve the state-of-the-art in nearly all subtasks of building an NLIDB. These include semantic parsing (Dong and Lapata, 2018; Zhang et al., 2019), ambiguity detection and confidence estimation (Dong et al., 2018; Yao et al., 2019), natural language response generation (Liu et al., 2019) and so on. Moreover, by jointly modeling the natural language question and database schema in the neural space, latest text-to-SQL semantic parsers can work cross domains (Yu et al., 2018; Zhang et al., 2019).

We present PHOTON, a modular, cross-domain NLIDB that adopts deep learning in its core components. PHOTON consists of (1) a neural semantic parser, (2) a human-in-the-loop question corrector, (3) a SQL query executor and (4) a natural language response generator. The *neural semantic parser* assumes limited DB content access due to data privacy concerns (§ **??**). It employs a BERT-based (Devlin et al., 2019) DB schema-aware question encoder and a pointer-generator decoder (See et al., 2017) with static SQL correctness check. It achieves competitive performance on the popular cross-domain text-to-SQL benchmark, Spider (Yu et al., 2018) (63.2% structure accuracy on the dev set based on the official evaluation).[1] The *question corrector* is a neural sequence editor which detects potential confusion span(s) in the input question and suggests possible corrections for the user to give feedback. When an input question is successfully translated into an executable SQL query, the *response generator* generates a natural language response conditioned on the output of the SQL *query executor*.

A pilot study with non-expert SQL users shows that the system effectively increases the flexibility of user's natural language expression and is easy to be adapted to unseen databases. Being able to detect and correct untranslatable questions reduces unexpected error cases during user interaction.

2 System Design

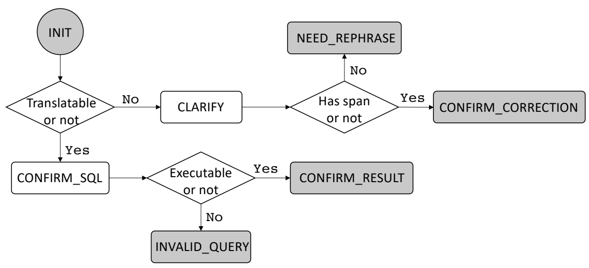
In this section, we will elaborate on the system design of PHOTON.

[1]We are continuously improving the performance of the neural semantic parser. Currently the semantic parser only accepts standalone question as input. We plan to also model the interaction context in future work.

2.1 Overview

Figure 1 shows the overall workflow of our system. PHOTON is an end-to-end system that takes a user question and database schema as input, and output the query result after executing the generated SQL on the database. PHOTON is a modular framework designed towards practical industrial applications. The core modules in PHOTON are the SQL parser and confusion detection mechanism. The SQL parser parses the input question and database schema, maps them into executable SQL query via an encoder-decoder framework. The confusion detection module identifies the untranslatable questions and captures the confusing span of the untranslatable question. The confusing tokens together with the context are fed into the auto-correction module to make a prediction of user attempted question.

To make it more applicable and accessible for user to query the database in a natural way, PHOTON also provides user interaction module enabling user to refine their queries in the interaction with the system. Response generation handles the output of the system by transducing the database-style query result into natural language or post warning when the query is non-executable on the database, making the system more user-friendly. Notice that the response generation module in the current version is implemented using a template-based approach and can be improved by using more advanced response generation models.



Response Template

CONFIRM_RESULT	"SQL: {PRED_SQL}. {NL_RESPONSE}"
CONFIRM_CORRECTION	"Sorry, {CONF_TOKENS} is confusing in our scenario, do you mean {CORR_TOKENS}?"
NEED_REPHRASE	"Sorry, it is a confusing question for me, please rephrase your question and ask again."
INVALID_QUERY	"Sorry, it is an invalidate query, please check the table names and associated fields of interest."

Figure 2: State transition map of interaction in PHOTON. States with darker background are the end states that can receive user reply, and switch to INIT state automatically. The bottom part is the system response templates in each end state.

2.2 User Interaction

Figure 2 illustrates the interaction process, which involves four types of response states: CONFIRM_RESULT, CONFIRM_CORRECTION, NEED_REPHRASE, and INVALID_QUERY. The set of response templates can be found at the bottom of Figure 2. When a user initiates the conversation by entering one query, PHOTON will first predict whether the query is translatable or not. If translatable, PHOTON generates the corresponding SQL command and checks the command's executability; otherwise, PHOTON will provide a correction strategy (i.e., CONFIRM_CORRECTION) based on the detected confusing span or ask the users to further rephrase the inquiry (i.e., NEED_REPHRASE) if no span is captured.

2.3 UI Design

Our system UI consists of three panels: chat window, schema viewer and results viewer.

- *Chat window*: This is a standard chat window that facilitates communication between the user and PHOTON. The user types the natural language input and the natural language responses of the system are displayed.
- *Schema viewer*: This view provides a graph visualization of the underlying relational DB schema. The panel is hideable and will not be shown in case the DB schema is confidential.
- *Result viewer*: This view displays the returned results of an executable SQL query mapped from a confirmed input question. The SQL query is formatted and displayed in the top for user verification. Multi-record results are presented as sub-tables. Result consists of a single table cell is presented as a 1-cell sub-table. If the result comes from an aggregation operation such as a counting, the data records supporting the calculation are also shown for explanability. Confidential DB records are hidden from the display and the user is informed of the number of hidden records.

2.4 Cross Domain

A relational DB for user queries should be set before usage. PHOTON consists of a collection of default databases and allows users to upload their own DBs for testing. Users can select which database they want to query by clicking the "Selected Database" drop down button.

References

Núria Bertomeu, Hans Uszkoreit, Anette Frank, Hans-Ulrich Krieger, and Brigitte Jörg. 2006. Contextual phenomena and thematic relations in database QA dialogues: results from a wizard-of-Oz experiment. In *Proceedings of the Interactive Question Answering Workshop at HLT-NAACL 2006*, pages 1–8, New York, NY, USA. Association for Computational Linguistics.

Ben Bogin, Jonathan Berant, and Matt Gardner. 2019. Representing schema structure with graph neural networks for text-to-sql parsing. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Vol. 1*, pages 4560–4565.

Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 5016–5026.

Deborah A. Dahl, Madeleine Bates, Michael Brown, William M. Fisher, Kate Hunicke-Smith, David S. Pallett, Christine Pao, Alexander I. Rudnicky, and Elizabeth Shriberg. 1994. Expanding the scope of the ATIS task: The ATIS-3 corpus. In *Human Language Technology, Proceedings of a Workshop held at Plainsboro, New Jerey, USA, March 8-11, 1994*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1*, pages 4171–4186.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Li Dong and Mirella Lapata. 2018. Coarse-to-fine decoding for neural semantic parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 731–742.

Li Dong, Chris Quirk, and Mirella Lapata. 2018. Confidence modeling for neural semantic parsing. In (Gurevych and Miyao, 2018), pages 743–753.

Catherine Finegan-Dollak, Jonathan K. Kummerfeld, Li Zhang, Karthik Ramanathan, Sesh Sadasivam, Rui Zhang, and Dragomir R. Radev. 2018. Improving text-to-sql evaluation methodology. In (Gurevych and Miyao, 2018), pages 351–360.

Iryna Gurevych and Yusuke Miyao, editors. 2018. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. Association for Computational Linguistics.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, USA, June 24-27, 1990*.

Wonseok Hwang, Jinyeung Yim, Seunghyun Park, and Minjoon Seo. 2019. A comprehensive exploration on wikisql with table-aware word contextualization. *CoRR*, abs/1902.01069.

Anna Korhonen, David R. Traum, and Lluís Màrquez, editors. 2019. *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Association for Computational Linguistics.

Fei Li and Hosagrahar Visvesvaraya Jagadish. 2014. Nalir: an interactive natural language interface for querying relational databases. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, pages 709–712. ACM.

Chen Liang, Jonathan Berant, Quoc V. Le, Kenneth D. Forbus, and Ni Lao. 2017. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 23–33. Association for Computational Linguistics.

Kevin Lin, Ben Bogin, Mark Neumann, Jonathan Berant, and Matt Gardner. 2019. Grammar-based neural text-to-sql generation. *CoRR*, abs/1905.13326.

Tianyu Liu, Fuli Luo, Pengcheng Yang, Wei Wu, Baobao Chang, and Zhifang Sui. 2019. Towards comprehensive description generation from factual attribute-value tables. In (Korhonen et al., 2019), pages 5985–5996.

Ana-Maria Popescu, Oren Etzioni, and Henry A. Kautz. 2003. Towards a theory of natural language interfaces to databases. In *Proceedings of the 8th International Conference on Intelligent User Interfaces, IUI 2003, Miami, FL, USA, January 12-15, 2003*, pages 149–157. ACM.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083.

Vidya Setlur, Sarah E. Battersby, Melanie Tory, Rich Gossweiler, and Angel X. Chang. 2016. Eviza: A natural language interface for visual analysis. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology, UIST 2016, Tokyo, Japan, October 16-19, 2016*, pages 365–377. ACM.

Vidya Setlur, Melanie Tory, and Alex Djalali. 2019. Inferencing underspecified natural language utterances in visual analysis. In *Proceedings of the 24th International Conference on Intelligent User Interfaces, IUI 2019, Marina del Ray, CA, USA, March 17-20, 2019*, pages 40–51. ACM.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Ziyu Yao, Yu Su, Huan Sun, and Wen-tau Yih. 2019. Model-based interactive semantic parsing: A unified framework and A text-to-sql case study. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 5446–5457. Association for Computational Linguistics.

Tao Yu, Rui Zhang, Heyang Er, Suyi Li, Eric Xue, Bo Pang, Xi Victoria Lin, Yi Chern Tan, Tianze Shi, Zihan Li, Youxuan Jiang, Michihiro Yasunaga, Sungrok Shim, Tao Chen, Alexander Richard Fabbri, Zifan Li, Luyao Chen, Yuwen Zhang, Shreya Dixit, Vincent Zhang, Caiming Xiong, Richard Socher, Walter S. Lasecki, and Dragomir R. Radev. 2019a. Cosql: A conversational text-to-sql challenge towards cross-domain natural language interfaces to databases. *CoRR*, abs/1909.05378.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir R. Radev. 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, Brussels, Belgium, Oct 31 - Nov 4, 2018*, pages 3911–3921.

Tao Yu, Rui Zhang, Michihiro Yasunaga, Yi Chern Tan, Xi Victoria Lin, Suyi Li, Heyang Er, Irene Li, Bo Pang, Tao Chen, Emily Ji, Shreya Dixit, David Proctor, Sungrok Shim, Jonathan Kraft, Vincent Zhang, Caiming Xiong, Richard Socher, and Dragomir R. Radev. 2019b. Sparc: Cross-domain semantic parsing in context. In (Korhonen et al., 2019), pages 4511–4523.

John M. Zelle and Raymond J. Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, IAAI 96, Portland, Oregon, USA, August 4-8, 1996, Volume 2.*, pages 1050–1055.

Rui Zhang, Tao Yu, Heyang Er, Sungrok Shim, Eric Xue, Xi Victoria Lin, Tianze Shi, Caiming Xiong, Richard Socher, and Dragomir R. Radev. 2019. Editing-based SQL query generation for cross-domain context-dependent questions. *CoRR*, abs/1909.00786.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. *CoRR*, abs/1709.00103.

A Appendices

A.1 Construct Untranslatable Questions

Table 1 shows a summary of different types of untranslatable questions based on analysis of CoSQL (Yu et al., 2019a) and Multi-WOZ (Budzianowski et al., 2018).

Table 2 shows examples of applying question-side and schema-side transformations to convert a translatable question from existing text-to-SQL datasets to an untranslatable question.

Reason	**Description**	**Example**
Underspecification	Input does not specify which data entries/attributes to query.	Q: What is the total? Schema:||Course_ID||Staring_Data||Course||…
Overspecification	Input asks for information that cannot be found in the DB.	Q: What is the name of the singer with the largest **net worth**? Schema: ||Singer_ID||Name||Birth_Year||Citizenship||
Ambiguity & Vagueness	Input contains ambiguous or vague expressions.	Q: Show me homes with **good** schools Schema: ||Address||Community||School Name||School Rating||
Beyond representation scope of SQL	Input asks for information that cannot be obtained by SQL logic.	Q: What is the **trend** of housing price this year? Schema: ||House ID||Location||Price||Number of amenties||
Not a query	Input is not a linguistically valid question.	Q: Cyrus teaches physics in department.
Others	Other cases that the question cannot be translated.	Q: How many **Russias** have Summer's transfer window? Schema: ||Name||Country||Type||Transfer Window||Transfer Fee||

Table 1: Types of untranslatable questions in text-to-SQL identified from manual analysis of CoSQL (Yu et al., 2019a) and Multi-WOZ (Budzianowski et al., 2018). A question span that is problematic for the translation is highlighted when applicable.

Transformation		**Original data**	**Transformed data**	**Confusing text span**
Question	Swap	Q1: How many *conductors* are there?	Q1: How many **soloists** are there ?	soloists
		S1: || Conductor_ID || Name || Age || Nationlity|| Year_of_Work ||		
		Q2: What are the maximum and minimum values of *area codes* ?	Q2: What are the maximum and minimum values of **types**?	types
		S2: || Vote_ID || Phone_Number || Area_Code || State || Created ||		
	Drop	Q1: How many *countries* exist?	Q1: How many are there?	WHOLE SENTENCE
		S1: || CoutryId || CountryName || Continent ||		
		Q2: What is the *official language* spoken in the country whose head of state is Beatrix?	Q2: What are the people in the country where Beatrix is located?	WHOLE SENTENCE
		S2: || CountryCode || HeadOfState || Captital || Language || IsOfficial || Percentage ||		
Schema Drop		Q1: How much *surface area* do the countries in the Carribean cover together?		surface area
		S1: || Name ||Continent || Region || *SurfaceArea* || Population || LifeExpectancy ||	S1: || Name ||Continent || Region || Population || LifeExpectancy ||	
		Q2: Find the name and *age* of the visitor who bought the most tickets at once.		age
		S2: ||Customer_ID||Name||Level_of_membership|| *Age* ||	S2: ||Customer_ID||Name||Level_of_membership||	

Table 2: Examples of question-side and schema-side transformations for generating training data for untranslatable question detection. Let Q denote the question and S denote the schema. For each transformation, we provide two examples, i.e., (Q1, S1) and (Q2, S2). The italic and bold fonts highlight phrases before and after transformations.