Context Dependent Sequence to Sequence Model on SparC

Tao Chen, Tao Yu, Dragomir Radev

LILY Lab, Yale University

LILY Lab

Introduction

Querying a relational database is often challenging and a natural language interface has long been regarded by many as the most powerful database interface. The problem of mapping a natural language utterance into executable SQL queries (text2sql) has attracted increasing attention from the semantic parsing community by virtue of a continuous effort of dataset creation and the modeling innovation that follows it. While most of these focus on precisely mapping stand-alone utterances to SQL queries, generating SQL queries in a context-dependent scenario has been studied less. The most prominent context-dependent text-to-SQL task is ATIS, which is set in the flight-booking domain and contains only one database. The phrasing of such questions heavily depends on the interaction history. This requires a practical text-to-SQL system to effectively process context information to synthesize the correct SQL logic.

Dataset

To enable modeling advances in context-dependent semantic parsing, we introduce SParC (cross-domain Semantic Parsing in Context), an expert-labeled dataset which contains 4,298 coherent question sequences (over 12,000 questions paired with SQL queries) querying 200 complex databases in 138 different domains. The dataset is built on top of Spider, the largest cross-domain context-independent text-to-SQL dataset available in the field.

The large number of domains provide rich contextual phenomena and thematic relations between the questions, which general-purpose natural language interfaces to databases have to address. In addition, it enables us to test the generalization of the trained systems to unseen databases and domains.

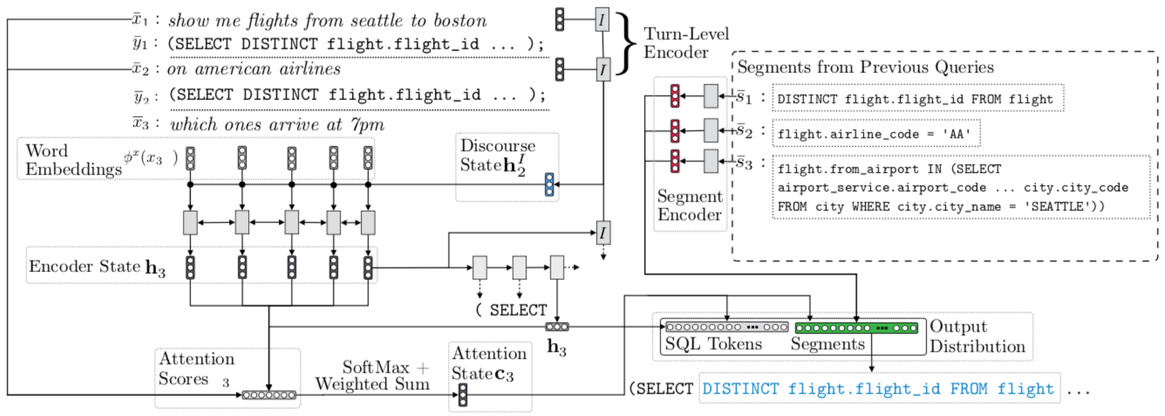

Figure 1. Illustration of the model architecture during the third decoding step while processing the instruction which ones arrive at 7pm from the interaction

Model	Question Match		Interaction Match	
	Dev	Test	Dev	Test
CD-Seq2Seq	17.1	18.3	**6.7**	**6.4**
SyntaxSQL-con	**18.5**	**20.2**	4.3	5.2
SyntaxSQL-inp	15.2	16.9	0.7	1.1

Table 1. Performance of various methods over all questions (question match) and all interactions (interaction match)

Turn #	CD-Seq2Seq	SyntaxSQL-con
1 (422)	31.4	38.6
2 (422)	12.1	11.6
3 (270)	7.8	3.7
$\geq$ 4 (89)	2.2	1.1

Table 2. Performance stratified by question turns on the development set. The performance of the two models decrease as the interaction continues.

Goal Difficulty	CD-Seq2Seq	SyntaxSQL-con
Easy (483)	35.1	**38.9**
Medium (441)	7.0	**7.3**
Hard (145)	**2.8**	1.4
Extra hard (134)	**0.8**	0.7

Table 3. Performance stratified by question difficulty on the development set. The performances of the two models decrease as questions are more difficult.

Thematic relation	CD-Seq2Seq	SyntaxSQL-con
Refinement	8.4	6.5
Theme-entity	13.5	10.2
Theme-property	9.0	7.8
Them./refine.-answer	12.3	20.4

Table 4. Performance stratified by thematic relations. The models perform best on the theme/refinement-answer relation, but do poorly on the refinement and theme-property relations.

Method

Context-dependent sequence-to-sequence (CD-Seq2Seq) was originally proposed for ATIS. To incorporate context within an interaction, it maintains and updates a discourse state encoder which encodes the history of previous utterances and is updated after each turn over the entire interaction, in addition to an utterance-level encoder for the current utterance. Also, positional encoding is used to take the position of each utterance relative to the current one into account.

Because the model was developed for ATIS, it does not take the database schema as input hence will only work for a single domain. For our task, encoding database schema (table and column names) is necessary for prediction across domains. Therefore, we adapt it to perform context-dependent SQL generation in multiple domains by modifying its encoders and decoders. We first add a database schema encoder to embed the column and table names by taking 300-dimensional GloVe pretrained word embeddings as a bag-of-words. At each generation step, the decoder selects between a SQL keyword or a column and table name from the corresponding database of the current question.

Conclusion

During this semester, I mainly focus on dialogue state tracking challenge and how to develop dialogue corpus for new dialogue system task. Encoder-decoder architecture with attention mechanism is proven to be effective in dialogue state tracking research. I surveyed and replicated several papers in this area. I firmly believe the SparC paper will be accepted by the ACL.

Acknowledgement

I would like to acknowledge Prof. Dragomir Radev for his supervision of my project. Also, I feel grateful to Tao Yu for his guidance and help in the through the progress.