Weakly-supervised Semantic Parsing

Tao Yu, Dragomir Radev PhD

Department of Computer Science, Yale University

LILY Lab

Introduction

Semantic parsers map natural language utterances to executable programs. The common approaches of training the parsers can be grouped into strongly supervised or weakly supervised. Supervised methods can have direct and detailed supervision using utterance-program pairs. However, labelling datasets is very costly. Training models based on indirect supervision of denotation only is a more attractive but also more challenging task. Moreover, weak supervision faces two main problems: large search space and spuriousness. In this poster, we focus on a new method introduced by (Guu et al., 2017) solving the spuriousness problem where an incorrect program can be executed to return a correct result, which is quite common in weak supervision. They propose a new learning algorithm that connects two common approaches to the problem, reinforcement learning (RL) and maximum marginal likelihood (MML). The new method combats spurious programs by introducing randomized exploration of RL into beam search traditionally employed in MML, which leads to more balanced exploration and gradients. They applied the method on a recent semantic parsing task and show significant gains on all subtasks.

Materials and Methods

In many semantic parsing tasks without direct supervision such as SCONE, a single action can be achieved by many different programs. For example, In Figure 1, the correct program captures the true meaning of the command. Without knowing the correct program, the model is likely to produce spurious programs that return the correct output accidentally but don't capture the true meaning of the utterances.

In order to tackle this problem, they first represent a program as a sequence in postfix notation and then formulate the task as a program sequence generation problem. Given input x = (u, w) where u is the utterance and w the target state, the model generates program tokens using a encoder-decoder attention model.

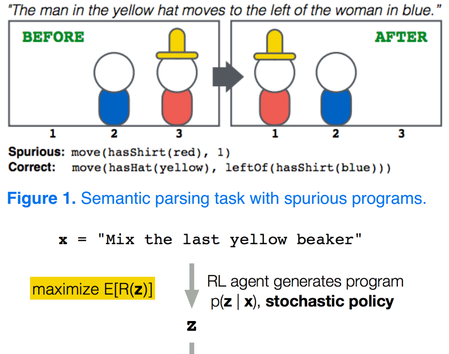

Figure 1. Semantic parsing task with spurious programs.

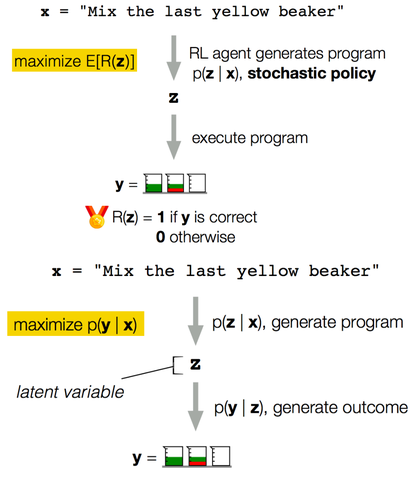

Figure 2. Two common approaches to the problem: reinforcement learning and maximum marginal likelihood

$$\nabla_\theta J = \sum_{(x,y)} \mathbb{E}_{\mathbf{z}\sim q}\left[R(\mathbf{z})\nabla \log p_\theta(\mathbf{z} \mid x)\right]$$
$$= \sum_{(x,y)} \sum_{\mathbf{z}} q(\mathbf{z}) R(\mathbf{z}) \nabla \log p_\theta(\mathbf{z} \mid x),$$

where $q(\mathbf{z})$ equals

$$q_{\mathrm{RL}}(\mathbf{z}) = p_\theta(\mathbf{z} \mid x) \quad \text{for } J_{\mathrm{RL}},$$
$$q_{\mathrm{MML}}(\mathbf{z}) = \frac{R(\mathbf{z}) p_\theta(\mathbf{z} \mid x)}{\sum_{\tilde{\mathbf{z}}} R(\tilde{\mathbf{z}}) p_\theta(\tilde{\mathbf{z}} \mid x)}$$
$$= p_\theta(\mathbf{z} \mid x, R(\mathbf{z}) \neq 0) \quad \text{for } J_{\mathrm{MML}}$$

Figure 3. The gradients of overall RL objective (expected reward) and MML's log marginal likelihood of the data.

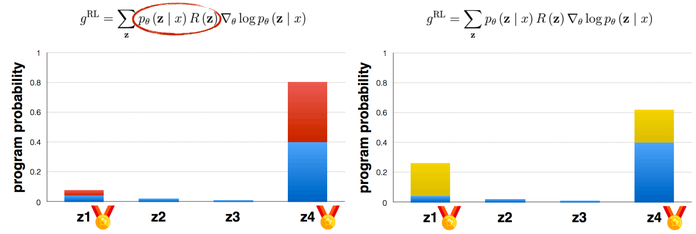

Figure 4. Program probability updates using direct update rule (left) and the meritocratic update rule (right).

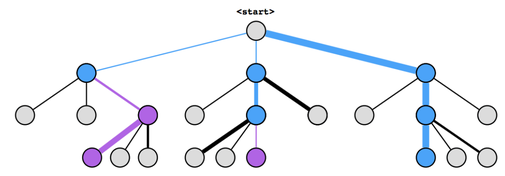

Figure 5. Randomized beam search

system	Alchemy 3utts	Alchemy 5utts	Tangrams 3utts	Tangrams 5utts	Scene 3utts	Scene 5utts
Long+16	56.8	52.3	64.9	27.6	23.2	14.7
REINFORCE	58.3	44.6	**68.5**	**37.3**	47.8	33.9
BS-MML	58.7	47.3	62.6	32.2	53.5	32.5
RandoMER	**66.9**	**52.9**	65.8	37.1	**64.8**	**46.2**

Table 1. Result comparison to prior work on the SCONE task

To train the model, the paper compares two learning frameworks, RL and MML, as shown in Figure 2. Both of them unweight the probability of a reward-earning predicted program z by gradient weight q(z) in Figure 3. Therefore, the model will be more likely to discover a spurious program during exploration if the current policy incorrectly assigns high probability to it. To solve this problem, they propose a meritocratic update rule so that any reward-earning program is up-weighted roughly equally.

In practice, the expectation of gradients is approximated by Monte Carlo integration for RL and numerical integration for MML. In particular, beam search is used to search states in MML. Traditional beam search suffers the problem of bias exploration of spurious programs. To alleviate this, they adapt the idea of random exploration from RL to beam search and propose a simple randomized beam search (Figure 5). Instead of selecting the B highest-scoring continuations at each time step, a random continuation is uniformly sampled with a certain probability as well. In this way the model has a chance to explore other programs even though that probability assigned by the current policy is low.

Results

Table 1 shows a comparison of the results of prior work on the SCONE task. The approach outperforms the previous state-of-art by significant margins. The paper also provides other results to show the detailed effects of randomized beam search and the meritocratic updates across all tasks.

Conclusion and Future Work

The approach introduced by this paper can be applied to many other semantic parsing tasks or even some related natural language processing fields when only indirect supervision is available. It solves the problem through more balanced exploration and gradients. We attempt to adapt this approach to two new tasks: seq2SQL without SQL labels and sequential question answering. In these two tasks, we have to convert natural language questions into SQL queries and execute them to return correct results. Because SQL is a well-formatted formal language and the datasets contain limited SQL structures (no "group by" , "join" and nested SQL), the search space is not very large but big enough to be complex. We still have to tackle the problem of spuriousness. Moreover, we will explore other approaches to the problem.

References

Paper: From Language to Programs: Bridging Reinforcement Learning and Maximum Marginal Likelihood (Guu et al., 2017)
Graphs are from the author's slides: kelvinguu.com/public/projects/Guu_Lang_to_Prog_ACL_2017_slides.pdf