Weakly Supervised Multi-task Learning for Semantic Parsing

Bo Shao^{1,2*} , **Yeyun Gong**^{2} , **Junwei Bao**^{2} , **Jianshu Ji**^{3} , **Guihong Cao**^{3} , **Xiaola Lin**^{1} and **Nan Duan**^{2}

^{1}School of Data and Computer Science, Sun Yat-sen University
^{2}Microsoft Research Asia
^{3}Microsoft AI and Research, Redmond WA, USA
shaobo2@mail2.sysu.edu.cn, {yegong, nanduan, jianshuj, gucao}@microsoft.com,
baojunwei001@gmail.com, linxl@mail.sysu.edu.cn

Abstract

Semantic parsing is a challenging and important task which aims to convert a natural language sentence to a logical form. Existing neural semantic parsing methods mainly use <question, logical form> (Q-L) pairs to train a sequence-to-sequence model. However, the amount of existing Q-L labeled data is limited and hard to obtain. We propose an effective method which substantially utilizes labeling information from other tasks to enhance the training of a semantic parser. We design a multi-task learning model to train question type classification, entity mention detection together with question semantic parsing using a shared encoder. We propose a weakly supervised learning method to enhance our multi-task learning model with paraphrase data, based on the idea that the paraphrased questions should have the same logical form and question type information. Finally, we integrate the weakly supervised multi-task learning method to an encoder-decoder framework. Experiments on a newly constructed dataset and ComplexWebQuestions show that our proposed method outperforms state-of-the-art methods which demonstrates the effectiveness and robustness of our method.

1 Introduction

The goal of semantic parsing is to convert a natural language sentence to executable logical form [Zettlemoyer and Collins, 2007; Jia and Liang, 2016]. These methods are typically in one of two research categories: grammar based semantic parsing methods [Lee *et al.*, 2016; Zhang *et al.*, 2017; Zettlemoyer and Collins, 2012] and neural semantic parsing methods [Dong and Lapata, 2018; Jia and Liang, 2016]. All these methods only leverage the annotation corpus which contains natural language sentence and logical form pairs. There are 3 issues with these methods, including: 1) how to accurately predict the logical form template. 2) how to improve the performance of entity mention generation in logical form. and 3) how to handle diverse question expressions for the same logical form. To tackle these challenges, we focus on using additional supervised signals from other tasks to help to train the semantic parser. Table 1 shows some examples of a corpus for different tasks. For the challenge of logical form template prediction, in the semantic parsing data, each logical form can be abstracted to a logical form template, such as the logical form $\lambda x.people.person.date\_of\_birth(\text{Obama}, x)$ can be abstracted to template $\lambda x.predicate(entity, x)$ and each type of question has one type of logical form template. Intuitively, question type information is beneficial to logical form template prediction. For the challenge of entity mention generation, we use supervised information of mention tags to help our semantic parsing model learning. For the challenge of diverse question expressions of the same logical form, the paraphrase data containing pairs of sentences' equivalent semantics are used in our model. Using the supervised signal in paraphrase data is helpful for improving the generalization ability of the semantic parsing model.

In order to use these supervised information from other tasks to enhance the original task, multi-task learning is one of the choices which has been successfully used in various tasks [Guo *et al.*, 2018; Lin *et al.*, 2018]. In this work, we design a multi-task learning method which integrate question type and entity mention tag information into the question semantic parsing model. Our semantic parsing model shares an encoder layer with auxiliary tasks (question type classification and entity mention detection) and uses different decoders for different tasks. Through this method, our model learns an encoder from three different kinds of supervised information. With the paraphrase data, we propose a weakly supervised learning method to enhance our multi-task learning model. We assume that our model should give the same output for each pair of sentences in the paraphrase data, such as "When was Alexander born?" and "The birthday of Alexander" have the same logical form:"$\lambda x.people.person.date\_of\_birth(\text{Alexander}, x)$". Based on this assumption, we propose a weakly supervised learning method, which incorporates a consistency loss.

To train a semantic parsing model, training data containing a set of <sentence, logical form> pairs which typically require experts to annotate is important. However, it is hard to collect enough data for training a semantic parser due to the cost of annotation. Existing datasets including complex questions are relatively small, such as Free917 [Cai and Yates, 2013], QALD-6 [Unger *et al.*, 2016] and LC-QuaD [Trivedi

[*] Work done while this author was visiting Microsoft Research.

Task	Natural Sentence	Supervised Label
Semantic Parsing	When was Obama born	$\lambda x.people.person.date\_of\_birth$(Obama, x)
	When was Obama's daughter born	$\lambda x \exists y.people.person.children$(Obama, y) and $people.person.date\_of\_birth(y, x)$
Classification	Who is the director of Inception	Single-relation
	Who is the son of director of Tom Hanks	Multi-hop
Mention Detection	When was Obama's daughter born	O O B O O
	Which film is directed by Tom Hanks	O O O O O B I
Paraphrase	How long is the flight from Singapore to Syndney	Distance of flight from Singapore to Syndney
	When was Alexander born	The birthday of Alexander

Table 1: Examples for different tasks. Semantic Parsing is a corpus containing sentence and logical form pairs. Classification is a corpus which can be used in question classification. Mention Detection is a corpus which contains a tag for each word in the sentences, "B" represents beginning,"I" represents inside, "O" represents outside. Paraphrase is a corpus which contains pairs of sentences with equivalence semantics

et al., 2017]. In ComplexWebQuestions [Talmor and Berant, 2018], a relatively larger dataset containing 34,689 examples, there are limited 4 types of questions in it. Therefore, constructing a large scale and more complex labeled dataset for semantic parsing is considerably meaningful. In this paper we construct a large scale semantic parsing dataset which contains more than 50,000 <question, logical form> pairs over 9 types of questions.

The main contributions of this paper are: 1) We design a multi-task learning method, which integrates the question type and entity mention tag information into the semantic parser. 2) We propose a weakly supervised learning method to enhance our multi-task learning model using paraphrase data. 3) We construct a relatively large scale semantic parsing dataset to advance the semantic parsing research. We will release the dataset together with our code[1]. 4) Experiments on two semantic parsing datasets show the effectiveness and robustness of our proposed method.

2 Model

First, we introduce the baseline model and then we show our multi-task architecture. Finally, we introduce our weakly supervised multi-task learning for semantic parsing.

2.1 Baseline Model

The baseline model is fed with the <question,logical form> pairs (Q, L). The question $Q = [x_1, x_2 ... x_{|Q|}], x_i \in V_q$ is used to generate logical form $L = [y_1, y_2, ... y_{|L|}], y_i \in V_l$. We define V_q as the source vocabulary and V_l as the target vocabulary. Our baseline model aims to estimating $P(L|Q)$, and the conditional probability can be formulated as follows:

$$p(L|Q) = \prod_{t=1}^{|L|} p(y_t | y_{t-1}, ..., y_1, Q) \tag{1}$$

The model is based on the sequence-to-sequence model with a bidirectional GRU-RNN [Cho *et al.*, 2014] layer as encoder and a unidirectional GRU-RNN layer as decoder in our baseline model, which has been applied in semantic parsing tasks in [Dong and Lapata, 2016; Rabinovich *et al.*, 2017]. Our model adopts the attention mechanism[Bahdanau *et al.*, 2014] and copy mechanism in [Gu *et al.*, 2016; See *et al.*, 2017] to deal with out-of-vocabulary (OOV) words. We define $h = [h_1, h_2 ... h_{|Q|}]$ as the hidden states of encoder and $s = [s_1, s_2 ... s_{|L|}]$ as the hidden states of decoder. And let c_t be the context vector computed from a weighted combination of encoder hidden states h, which is same as basic sequence-to-sequence model in [Dong and Lapata, 2016; See *et al.*, 2017]. The probability distribution P_g^t over the target vocabulary V_l can be computed as follows:

$$P_g^t(y_t) = sfm(W_p(W[s_t; c_t] + b) + b_p) \tag{2}$$

where W_p, W, b, b_p are trainable parameters, $sfm(.)$ is the softmax function.

Specially, to tackle out-of-vocabulary words, we incorporate the same copy mechanism as [Gu *et al.*, 2016; See *et al.*, 2017] in our decoder. Attention score a_i is used as probability distribution of copy mechanism from input question. The final distribution at time step t is computed as:

$$\begin{aligned} g_c =& \sigma(W^*[c_t; s_t] + b^*) \\ P_v^t(y_t) =& (1 - g_c) P_g^t(y_t) + g_c \sum_{i: x_i = y_t} a_i^t \end{aligned} \tag{3}$$

where W^*, b^* are trainable parameters, σ is a non-linear function, the gate g_c is used to decide whether y_t should be copied from the input question or generated from the target vocabulary V_l.

Finally, we compute the overall loss function in all steps:

$$loss_{s2s} = \frac{\sum_{t=0}^{|L|} -log P_v^t(y_t)}{|L|} \tag{4}$$

2.2 Multi-task Learning Architecture

In the basic sequence-to-sequence model, the logical form is generated according to the question context and the model is only trained with $<Q, L>$ pairs. However, the $<Q, L>$ pairs are hard to annotate. To overcome the limitations of training data, we incorporate multi-task learning architecture to improve the performance of our model with supervised information of other tasks.

In this work, we leverage question type classification and entity mention detection as auxiliary tasks to boost our semantic parsing model. We define $T \in \{1, 2, ..., n\}$ to denote the index of question types and $M = \{m_1, ..., m_{|Q|}\}$, where $m_i \in \{B, I, O\}$, to denote the mention tags. The multi-task learning model is aimed at estimating the conditional probability $p(L, T, M|Q)$. We decompose $p(L, T, M|Q)$ into 3 distributions, $p(L|Q)$, $p(T|Q)$, and $p(M|Q)$.

We have introduced $p(L|Q)$ in previous section. In this section, we will introduce the two auxiliary tasks, which are aimed at estimating $p(T|Q)$ and $p(M|Q)$.

[1] https://github.com/shaoboly/wsmtl

Question Classification

The task of question classification is to classify the question to n types T such as Multi-hop, Multi-constraint .etc. It can be formulated as a conditional probability $p(T|Q)$. To estimate $p(T|Q)$, We share the encoder with baseline method and add a classification layer. The input of classification layer is the final hidden state $h_{|Q|}$ of shared encoder and the output is question type distribution P_{type}. At training step, the probability distribution P_{type} over n question types and the cross-entropy loss of classification $loss_c$ can be calculated as follows:

$$\begin{aligned} P_{type} &= sfm(W_s h_{|Q|} + b_s) \\ loss_c &= -\sum_{d=1}^{n} [T = d] \log P_{type}(T) \end{aligned} \tag{5}$$

where W_s, b_s are trainable parameters, $P_{type}(T)$ denotes the conditional probability $p(T|Q)$, sfm(.) denotes the softmax function.

Mention Detection

The task of entity mention detection is to detect the mention words in the question, which is important to help mention words generation in logical form. In our baseline model, the mention words mainly rely on a copy mechanism. However, it is difficult to copy the mention words for complex questions which contain multiple mention spans. We leverage the mention tags to boost the copy mechanism in our model.

The probability of mention tags generation can be factorized as:

$$p(M|Q) = \prod_{i=1}^{|Q|} p(m_i|Q) \tag{6}$$

To estimate $p(M|Q)$ We add an RNN layer with GRU unit. The inputs of it are the hidden vectors h of encoder, and outputs are “B”,“I”,“O” tags corresponding to each word in question. The hidden state h'_i and mention tags distribution of ith step in the RNN layer can be formulated as:

$$\begin{aligned} h'_i &= GRU(h_i, h'_{i-1}) \\ P^i_{tag} &= sfm(W_m h'_i + b_m) \end{aligned} \tag{7}$$

where w_m,b_m are trainable parameters, sfm(.) is the softmax function, and P^i_{tag} is the distribution of tags of the ith token and $P^i_{tag}(m_i)$ denotes $p(m_i|Q)$ in Eq. 6.

Then the loss function of mention detection task can be calculated as follows:

$$loss_{tag} = \frac{\sum_{i=0}^{|Q|} -logP^i_{tag}(m_i)}{|Q|} \tag{8}$$

where $|Q|$ is the length of input question.

Loss Function

The multi-task model is trained with semantic parsing, question classification and semantic parsing jointly. The overall loss function is weighted sum of the three loss as:

$$loss_{mtl} = loss_{s2s} + \alpha loss_{type} + \beta loss_{tag} \tag{9}$$

where $\alpha, \beta \in [0, 1]$ are hyper-parameters, which are used to balance the weight of three tasks.

2.3 Weakly Supervised Multi-task Learning

In this section, we present our weakly supervised method using the paraphrase data to boost the multi-task learning model. Each instance of paraphrased data contains a pair of questions $<Q^1$,$Q^2>$ which are meaning-equivalent. In our assumption, the question Q^2 should have the same logical form and question type with Q^1, since Q^2 is annotated by the root question Q^1. We use $p(L, T, M^1|Q^1)$ and $p(L, T, M^2|Q^2)$ to represent the conditional probabilities given questions Q^1 and Q^2 respectively. Our goal is to minimize the relative distances between $p(L, T, M^1|Q^1)$ and $p(L, T, M^2|Q^2)$ and ignore the influence of $p(M|Q)$. We use Kullback-Leibler divergence [Kullback, 1997] to measure the distance and only use $p(L|Q^i)$ and $p(T|Q^i)$ to compute its divergence, which can be formulated as:

$$\begin{aligned} &D_{KL}(p(L, T, M^1|Q^1)||p(L, T, M^1|Q^2)) \\ \propto &D_{KL}(p(L|Q^1)||p(L|Q^2)) + D_{KL}(p(T|Q^1)||p(T|Q^2)) \end{aligned} \tag{10}$$

We first introduce the computation of $D_{KL}(p(L|Q^1)||p(L|Q^2))$. Given a question Q^1 and its annotated paraphrase question Q^2, we formulated the computation of the KL divergence:

$$\begin{aligned} &D_{KL}(p(L|Q^1)||p(L|Q^2)) \\ &= \sum_{L \in S(L)} p(L|Q^1) log(p(L|Q^1)/p(L|Q^2)) \end{aligned} \tag{11}$$

where $S(L)$ represents the set of all possible logical forms with the input question Q^1. Since Q^2 is annotated by Q^1, we can consider $p(L|Q^1)$ as constant target distribution and compute the partial derivative as follows:

$$\begin{aligned} &\frac{\partial D_{KL}(p(L|Q^1)||p(L|Q^2))}{\partial \theta} \\ &= -\sum_{L \in S(L)} p(L|Q^1) \frac{log(p(L|Q^2))}{\partial \theta} \\ &= -\mathbf{E}_{L \sim p(L|Q^1)} \frac{log(p(L|Q^2))}{\partial \theta} \end{aligned} \tag{12}$$

where θ represent the parameters in our model and the expectation $-\mathbf{E}_{L \sim p(L|Q^1)}$ can be approximated by samples from $p(L|Q^1)$. Thus minimizing the KL divergence is equal to maximize the log-likehood on samples from the decoding result from Q^1 with the input of Q^2.

In each training step, we first generate the candidates by our model with the input samples of Q^1. Then we make Q^2 with each corresponding candidate L' as a pseudo pair. We optimize our model using these pseudo pairs same as training data. We optimize the parameters of our model based on Eq. 4 and compute the consistency loss as $loss^l_{para}$.

For question type classification, we compute distributions P^1_{type} and P^2_{type} over n types for questions Q^1 and Q^2. Then we compute the KL divergence $D(p(T|Q^1)||p(T|Q^2))$ as loss $loss^{type}_{para}$ which can be calculated as follows:

$$loss^{type}_{para} = \frac{1}{n} \sum_{d=1}^{n} \log P^1_{type}(d) \log(\frac{P^1_{type}(d)}{P^2_{type}(d)}) \tag{13}$$

Finally, joint with the loss of multi-task learning $loss_{mtl}$, we achieve the loss for our weakly supervised multi-task learning model:

$$loss_{ws} = loss_{mtl} + \lambda_1 loss^{l}_{para} + \lambda_2 loss^{type}_{para} \qquad (14)$$

where $\lambda_1, \lambda_2 \in [0, 1]$ are hyper-parameters, which are used to balance the weight of the multi-task learning and weakly supervised learning in our model.

3 Dataset

3.1 Dataset Construction

In this paper, we construct a Large Scale semantic Parsing Dataset(LSParD) based on a knowledge base (KB), we use Freebase for our dataset. It contains a set of nodes and edges, which are always represented by triple $\{s, p, o\}$. Each triple denotes two nodes, a subject entity s, an object entity o and the directed edge p between them as a predicate. A kind of special KB node called CVT in KB, which is a compound type, is used to denote events.

In our LSParD, some primary functions such as *Argmax, Argmin,Argmore, Argless, Max, Min* are defined to denote basic functional natural language expressions, such as ”larger than”, ”the longest”, and so on. Then we collect our dataset by crowd sourcing which contains five steps. First, we collect the connected triples which has one same node on the knowledge graph. Second, we annotate a question for each triple. Third, we automatically generate complex questions for the connected triples through the simple question of each triple using a template. Fourth, the workers paraphrase the questions generated from template. Finally, 3 other workers verify the quality of paraphrasing.

Type	Example
Single-Relation	when was Steve Jobs born
CVT	who played deputy Ferguson in Project Viper
Multi-Hop	which film was written by the director of Wonder
Multi-Constraint	what movie was produced by Milan Cheylov that Samantha Follows acted in
Multi-Choice	which was invented by Steve Jobs, Alt code or iPhone
Yes/No	was the iPod invented by Steve Jobs
Superlative	what is the longest road in the world
Aggregation	how many children does Bill Gates have
Comparison	which country has more than 100 million people

Table 2: Examples for each question type in our dataset

We roughly divide questions in our dataset into 9 categories. “Single-Relation” are questions which ask for a single relation which consist of one entity and one predicate. “CVT” are questions which involve a node with a compound type and connected with multiple entities through multiple predicates. “Multi-Hop” are questions that can be transformed into a path which contains multiple entities linked by predicates. “Multi-Constraint” are questions which ask for the answer with multiple constraints by entities and predicates. “Multi-Choice” are questions which involve a set of candidate answers to choose from. “Yes/No” are questions that ask about the existence of a relation and expect a Yes or No answer. “Superlative” denotes an operation on a subgraph containing a set of entities which belong to a special type and are linked by the same comparable predicate. The operation can be $ArgMax$, $ArgMin$, Max or Min. “Aggregation” denotes the questions asked about the number of all connected entities linked to the giving entity with the same predicate. “Comparison” are questions about a set of entities linked with the same predicate to the same object, in which the predicate is comparable. These questions ask for the entities which satisfy more or less (denoted by $ArgMore$ or $ArgLess$ in the logical forms) than a given threshold.

3.2 Dataset Statistic

Questions in our dataset are annotated with λ-Calculus. Table 3 shows statistics of our dataset. From the table, we observe that our dataset contains 51,164 $<$ question, logical form $>$ pairs and 9 types of questions which is a large scale and complex semantic parsing dataset. Existing large scale semantic parsing datasets, LC-QuAD [Trivedi *et al.*, 2017] has 5,000 questions and it is annotated with SPARQL queries based on DBpedia. ComplexWebQuestion(CWQ) contains 34,689 examples and 4 types of complex questions with SPARQL queries based on Freebase. WikiSQL [Zhong *et al.*, 2017] has 80,654 questions and it is annotated with SQL which can be executed on tables. This work mainly focuses on constructing a semantic parsing dataset based on knowledge graph such as Freebase and DBpedia. Our dataset is the largest in this type of datasets as far as we know. We will publish this dataset with more detailed instructions.

Type	Question	LFP	Entity
Single-Relation	28,776	621	13,367
CVT	5,115	437	4,045
Multi-Hop	7,452	689	1,950
Multi-Constraint	2,601	235	2,962
Multi-Choice	1,344	448	2,376
Yes/No	2,688	448	2,387
Superlative	1,013	179	326
Aggregation	1,818	256	922
Comparison	357	48	219
Statistic	**51,164**	**3,361**	**23,144**

Table 3: Statistics of our dataset. “Question” represents the number of questions. “LFP” denotes the number of logical form patterns. “Entity” represents the number of entities in the dataset

4 Experiment

We evaluate our method on two datasets, our LSParD and ComplexWebQuestion (CWQ).

4.1 Experimental Setup

Preprocessing

For weakly supervised learning, we use WikiAnswers paraphrase corpus [Fader *et al.*, 2013] in our model. The dataset contains 18M of question-paraphrase pairs and 2.4M distinct questions. We build a search engine using BM25 to select the most relevant 220,000 question pairs with the questions in the semantic parsing corpus and use these paraphrase data as our weakly supervised learning data.

For ComplexWebQuestion (CWQ), since the dataset only provides the ID of entity without its name and mention words in the question. We find the entity names in CWQ from Freebase. Some entity names are not contained in questions(19.7% of the dataset), since questions have been paraphrased according to author of the dataset. We will release the preprocessed dataset together with our code and LSParD.

Configuration

We use Glove word embeddings [Pennington *et al.*, 2014] as our pre-trained word embeddings. We set the dropout rate to 0.5. The dimension of all hidden vectors and word embedding is 300. Word vocabulary and embedding are not shared between encoder and decoder. In all our experiment, α and β are set to 0.5, λ_1 and λ_2 are set to 0.1.

We compare several previous works in our experiment, including SEQ2SEQ and SEQ2TREE [Dong and Lapata, 2016], PointerGenerator [See *et al.*, 2017] and Coarse2fine [Dong and Lapata, 2018]. We directly train their model on LSParD and CWQ with their released code. In baseline methods SEQ2SEQ and SEQ2TREE, we use the same method to pre-process the training data as they do to replace the entities as a placeholder in the logical forms and questions. For evaluation, we use the accuracy of exactly match to evaluate the performance of our model.

4.2 Experimental Results

Method	LSParD	CWQ
SEQ2SEQ [Dong and Lapata, 2016]	33.5	43.7
SEQ2TREE [Dong and Lapata, 2016]	33.1	44.1
Coarse2fine [Dong and Lapata, 2018]	52.3	48.5
PointerGenerator [See *et al.*, 2017]	51.2	47.9
Baseline	52.1	49.1
Multi-Task-Learning (MTL)	54.1	51.0
w/o Mention Detection	53.3	49.8
w/o Question Classification	53.6	50.3
Weakly-Supervised MTL (WS-MTL)	**56.8**	**52.7**
w/o Mention Detection	55.1	51.7
w/o Question Classification	54.7	52.2

Table 4: The accuracy of our methods with previous work on LSParD and CWQ

Table 4 shows the comparisons of the proposed method with the state-of-the-art methods and variants of the proposed model. "Baseline" represents the single task model based on the basic architecture. "MTL" denotes the multi-task learning method we propose in this paper. "WS-MTL" is the weakly supervised multi-task learning method. From the results, we observe that the method "WS-MTL" proposed in this paper achieves the best performance. Compared to the state-of-the-art method "PointerGenerator", our model achieves around 11% relative improvement on the LSParD and around 10% relative improvement on the CWQ, which illustrates the effectiveness and robustness of the proposed model. Comparing the results of "MTL" and "Baseline", we observe that "MTL" achieves improvement on both datasets, which demonstrates that the question type and mention tag information used in our "MTL" are beneficial to our semantic parsing. Comparing the results "MTL" and "WS-MTL", we see that the weakly supervised learning method we used to enhance the "MTL" achieves further improvement. To evaluate the influence of each auxiliary task, we conduct an ablation test, such as "MTL" without "Mention Detection" or "Question Classification". The bottom of Table 4 lists the results of ablation test. The results show that each auxiliary task is helpful to improve the performance of our semantic parsing task in both "MTL" and "WS-MTL".

4.3 Auxiliary Task Results

We also analyze the performance on auxiliary tasks, and find that the auxiliary tasks also improve compared with training the task itself alone. And the weakly supervised learning can also boost performance.

Question Classification

Table 5 shows the performance of question type classification on LSParD and CWQ. "BiLSTM-C" denotes the method that only use the shared encoder and classification layer in our model. From the results, we observe that the "MTL" achieves better performance than "BiLSTM-C". We believe that the supervised information of semantic parsing and mention tag are also beneficial to question type classification. Furthermore, we observe that "WS-MTL" achieves the best performance on question classification which illustrates weakly supervised learning not only improves the performance of semantic parsing, but also boost question classification. We compare the results between LSParD and CWQ for the same model. We see that the accuracy on LSParD is lower than CWQ, the reason is LSParD has more types of questions than CWQ. The results show that our model is more effective when the task is more complicated.

	LSParD(ACC)	CWQ(ACC)
BiLSTM-C	78.6	86.4
MTL	82.5	87.1
WS-MTL	84.8	89.1

Table 5: Performance of question type classification

Mention Detection

Table 6 shows the performance of mention detection task on LSParD and CWD. "BiLSTM-D" denotes the methods that only use the shared encoder and mention detection layer in our model. From the results of "BiLSTM-D" and "MTL", we observe that semantic parsing and question type information is helpful in improving the performance of mention detection. Comparing the results of "MTL" and "WS-MTL", it shows the same trend with question classification task which illustrates the robustness of weakly supervised learning method.

	LSParD(ACC)	CWQ(ACC)
BiLSTM-D	77.4	76.5
MTL	79.6	77.9
WS-MTL	80.7	78.6

Table 6: Performance of mention detection

4.4 Discussion and Analysis

We conduct an experiment to evaluate the consistence of logical forms generated from our model for paraphrase data. We use BLEU-4 [Papineni *et al.*, 2002] as evaluation metric to measure the consistence of our model. We randomly select 20,000 pairs of paraphrase dataset as test set for consistence evaluation. Table 7 shows the performance of our model. The results show that our model achieves significant improvement on BLEU-4 score which illustrates our model can generate more similar logical forms for question pairs in paraphrase data. Comparing the results of "Baseline" and "MTL", we find that "MTL" weakly improves performance. While our "WS-MTL" method achieves around 43.6% relative improvement over BLEU-4, the results demonstrates the ability of our model to handle various expression of questions.

	LSParD(BLEU-4)	CWQ(BLEU-4)
Baseline	61.2	70.1
MTL	63.9	71.7
WS-MTL	87.9	89.6

Table 7: Performance of model robustness

As show in Figure 1, we train the models with different sizes of training data. Data sizes ranges from 20% to 100%. We find that our method achieves improvement among various sizes of training data. Furthermore, our method achieves more improvements for low training data. It demonstrates that our method effectively uses supervised information of other tasks to help semantic parsing.

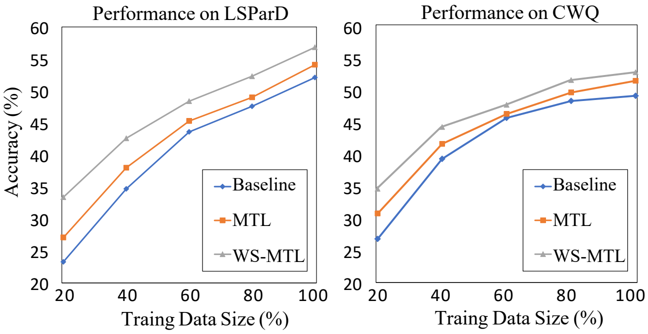

Figure 1: Performance of various amounts of training data in LSParD and CWQ

We further analyze the performance of each type of questions on LSParD in Table 8. We replace the entity and predicate in the logical form to measure its Pattern accuracy. The results show that the question type information helps the model for logical form pattern generation, especially for the types which contain small scale training data, such as Superlative, Aggregation, and Comparative. Comparing the results of entity generation, we find that our model with mention supervised information improves the performance of entity generation effectively. Finally the results of the whole logical form generation show that our model achieves improvements for all types of questions which illustrates the effectiveness and robustness of the proposed method.

	Pattern		Entity		LF	
Question Type	**Our**	**B**	**Our**	**B**	**Our**	**B**
Single-Relation	99.3	98.5	98.3	98.1	83.4	82.1
CVT	92.5	91.2	78.4	75.8	40.2	38.2
Multi-Hop	95.2	94.8	93.3	91.3	74.0	68.7
Multi-Constraint	95.4	94.1	74.5	71.7	47.8	42.1
Multi-Choice	79.6	79.8	61.2	51.8	31.5	24.2
Yes/No	85.7	83.7	68.4	64.8	46.6	43.9
Superlative	78.2	69.7	77.7	77.6	68.6	60.1
Aggregation	83.5	74.4	86.1	83.7	51.5	38.5
Comparative	82.7	78.6	90.8	85.7	71.4	59.2
Overall	91.1	89.2	82.5	79.7	56.8	52.1

Table 8: Accuracy for different types of questions on different aspects. "Pattern" represents the accuracy of logical form patterns. "Entity" represents the accuracy of entity prediction. "LF" denotes the accuracy of complete logical form generation. "Our" refers to our model "WS-MTL" and "B" refers to the "baseline" model

5 Related Work

Semantic parsing, as an important task of natural language understanding, has been paid a lot of attention. Many semantic parsers [Kwiatkowski *et al.*, 2011; Berant *et al.*, 2013; Dong and Lapata, 2018] are learned based on labeled <sentence, logical form> pairs which are hard to obtain. To address this problem, multi-task [Peng *et al.*, 2017],transfer learning [Fan *et al.*, 2017], paraphrasing [Su and Yan, 2017] and weakly supervision approaches [Goldman *et al.*, 2018; Cheng and Lapata, 2018] are proposed to train semantic parsers. In this work, we follow the multi-task learning mechanism. Multi-task learning has been used in various natural language processing tasks, such as text classification [Liu *et al.*, 2017], reading comprehension [Wang *et al.*, 2018] and summarization [Guo *et al.*, 2018]. [Guo *et al.*, 2018; Wang *et al.*, 2018] use shared layers to incorporate auxiliary tasks. [Liu *et al.*, 2017] use adversarial learning to find common information from different tasks. We propose a weakly supervised multi-task learning model for semantic parsing.

6 Conclusion

In this paper, we propose a weakly supervised multi-task learning method for semantic parsing. We design a multi-task architecture which incorporates question type and mention tag information into the semantic parser. To further improve model performance and consistence, we propose a weakly supervised method to enhance our multi-task learning model. The experiments on two datasets show that our multi-task learning architecture significantly improves performance of baseline method. Furthermore, the weakly supervised learning mechanism we propose effectively improves the performance of the multi-task learning model.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grants No.U1711263.

References

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[Berant *et al.*, 2013] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *EMNLP*, 2013.

[Cai and Yates, 2013] Qingqing Cai and Alexander Yates. Large-scale semantic parsing via schema matching and lexicon extension. In *ACL*, 2013.

[Cheng and Lapata, 2018] Jianpeng Cheng and Mirella Lapata. Weakly-supervised neural semantic parsing with a generative ranker. *arXiv preprint arXiv:1808.07625*, 2018.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[Dong and Lapata, 2016] Li Dong and Mirella Lapata. Language to logical form with neural attention. In *ACL*, 2016.

[Dong and Lapata, 2018] Li Dong and Mirella Lapata. Coarse-to-fine decoding for neural semantic parsing. *arXiv preprint arXiv:1805.04793*, 2018.

[Fader *et al.*, 2013] Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. Paraphrase-driven learning for open question answering. In *ACL*, 2013.

[Fan *et al.*, 2017] Xing Fan, Emilio Monti, Lambert Mathias, and Markus Dreyer. Transfer learning for neural semantic parsing. *arXiv preprint arXiv:1706.04326*, 2017.

[Goldman *et al.*, 2018] Omer Goldman, Veronica Latcinnik, Ehud Nave, Amir Globerson, and Jonathan Berant. Weakly supervised semantic parsing with abstract examples. In *ACL*, 2018.

[Gu *et al.*, 2016] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*, 2016.

[Guo *et al.*, 2018] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. Soft layer-specific multi-task summarization with entailment and question generation. *arXiv preprint arXiv:1805.11004*, 2018.

[Jia and Liang, 2016] Robin Jia and Percy Liang. Data recombination for neural semantic parsing. In *ACL*, 2016.

[Kullback, 1997] Solomon Kullback. *Information theory and statistics*. Courier Corporation, 1997.

[Kwiatkowski *et al.*, 2011] Tom Kwiatkowski, Luke Zettlemoyer, Sharon Goldwater, and Mark Steedman. Lexical generalization in ccg grammar induction for semantic parsing. In *EMNLP*, 2011.

[Lee *et al.*, 2016] Kenton Lee, Mike Lewis, and Luke Zettlemoyer. Global neural ccg parsing with optimality guarantees. *arXiv preprint arXiv:1607.01432*, 2016.

[Lin *et al.*, 2018] Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. A multi-lingual multi-task architecture for low-resource sequence labeling. In *ACL*, 2018.

[Liu *et al.*, 2017] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742*, 2017.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318, 2002.

[Peng *et al.*, 2017] Hao Peng, Sam Thomson, and Noah A Smith. Deep multitask learning for semantic dependency parsing. *arXiv preprint arXiv:1704.06855*, 2017.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[Rabinovich *et al.*, 2017] Maxim Rabinovich, Mitchell Stern, and Dan Klein. Abstract syntax networks for code generation and semantic parsing. *arXiv preprint arXiv:1704.07535*, 2017.

[See *et al.*, 2017] Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

[Su and Yan, 2017] Yu Su and Xifeng Yan. Cross-domain semantic parsing via paraphrasing. *arXiv preprint arXiv:1704.05974*, 2017.

[Talmor and Berant, 2018] Alon Talmor and Jonathan Berant. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*, 2018.

[Trivedi *et al.*, 2017] Priyansh Trivedi, Gaurav Maheshwari, Mohnish Dubey, and Jens Lehmann. Lc-quad: A corpus for complex question answering over knowledge graphs. In *International Semantic Web Conference*, pages 210–218. Springer, 2017.

[Unger *et al.*, 2016] Christina Unger, Axel-Cyrille Ngonga Ngomo, and Elena Cabrio. 6th open challenge on question answering over linked data (qald-6). In *Semantic Web Evaluation Challenge*, pages 171–177. Springer, 2016.

[Wang *et al.*, 2018] Yizhong Wang, Kai Liu, Jing Liu, Wei He, Yajuan Lyu, Hua Wu, Sujian Li, and Haifeng Wang. Multi-passage machine reading comprehension with cross-passage answer verification. *arXiv preprint arXiv:1805.02220*, 2018.

[Zettlemoyer and Collins, 2007] Luke Zettlemoyer and Michael Collins. Online learning of relaxed ccg grammars for parsing to logical form. In *EMNLP-CoNLL*, 2007.

[Zettlemoyer and Collins, 2012] Luke S Zettlemoyer and Michael Collins. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. *arXiv preprint arXiv:1207.1420*, 2012.

[Zhang *et al.*, 2017] Yuchen Zhang, Panupong Pasupat, and Percy Liang. Macro grammars and holistic triggering for efficient semantic parsing. *arXiv preprint arXiv:1707.07806*, 2017.

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