ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200350

SETNet: A Novel Semi-Supervised Approach for Semantic Parsing

Xiaolu Wang * and **Haifeng Sun** and **Qi Qi** and **Jingyu Wang** [1]

Abstract. In this work, we study on semi-supervised semantic parsing under a multi-task learning framework to alleviate limited performance caused by limited annotated data. Two novel strategies are proposed to leverage unlabeled natural language utterances. The first one takes entity predicate sequences as training targets to enhance representation learning. The second one extends Mean Teacher to seq2seq model and generates more target-side data to improve the generalizability of decoder network. Different from original Mean Teacher, our strategy produces hard targets for the student decoder and update the decoder weights instead of the whole model. Experiments demonstrate that our proposed methods significantly outperform the supervised baseline and achieve more impressive improvement than previous methods.

1 Introduction

Semantic parsing aims to map natural language utterances into machine executable meaning representations [12] (*e.g.* abstract meaning representation or logical forms). It constitutes a key technology for achieving the long-term goal of being capable of understanding natural language in artificial intelligence, with a wide range of applications in NLP tasks including robot controlling, question answering, database queries, etc. In this work, we study on logical forms. Fig 1 shows an example of natural language utterance, corresponding logical forms and its denotation.

Recent researches have demonstrated the effectiveness of sequence-to-sequence (seq2seq) based model on semantic parsing [7, 12, 23]. The key to fully explore advantage of such sequence-to-sequence based models is abundant natural language utterances and corresponding logical forms [26]. However, logical forms are diverse in different application scenarios (*e.g.* lambda-calculus and SQL). The annotations of a particular scenarios are difficult to reuse in other scenes, which makes annotation processing more expensive. Namely, the limited amount of annotated data has become a huge obstacle of seq2seq-based semantic parsing applications.

One solution to tackle the data-hungry challenge in seq2seq-based model is to exploit unannotated utterances in semi-supervised learning. This can be done for the reason that natural language utterances are large-scale and readily-available in most domains, meeting demands for semi-supervised learning. Unlabeled utterances can improve model generalizability and ease over-fitting caused by limited labeled training set. Many recent researches leverage unannotated utterances to learn the information common in statistics, which provides model with general representations rather than ones targeted towards a particular task like semantic parsing, thus their contribution to the task is limited.

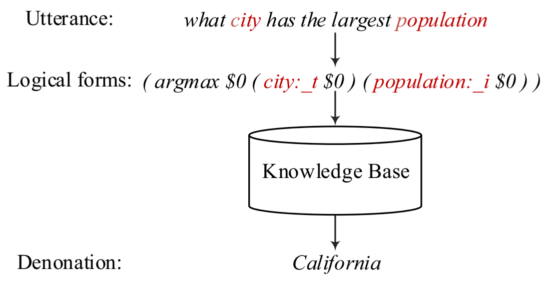
Utterance:
what city has the largest population
Logical forms:
(argmax $0 (city:_t $0) (population:_i $0))
Knowledge Base
Denonation:
California

Figure 1: An example of natural language utterance, corresponding logical form and denotation on GEO.

Another one is multi-task learning (MTL), which generally alleviates demands for annotated data by transferring knowledge from related auxiliary tasks to the main task. In the generic MTL architectures for sequence modeling, lower layers are shared for transferring common knowledge between tasks, and higher layers are independent and remained for various task. While in the case of low resource of annotated data, the generalization performance of higher layers is still weak.

In this work, we combine ideas of the two solutions and propose *SETNet*, a semi-supervised approach with two novel strategies leveraging unlabeled utterances in a MTL framework. The encoder is shared all the time. Different from generic MTL, SETNet benefits from the source-side information to improve the generality of encoder as well as decoder network.

Specifically, the first strategy, *entity lexicon learning*, provides an independent and related auxiliary task. It works by strengthening the encoder network and obtaining high quality representation, directing at semantic parsing. Semantic parsing is an exactly matching task, meaning that miss or mismatch of key information in representation will result in wrong predictions. The sequence autoencoder [6] can enhance representation learning with a training objective of minimizing reconstructing error, whereas it pays more attention to common information in many utterances than key information in a specific utterance. As an improvement, the training objective of our strategy is to maximize the log likelihood of entity predicate sequences (*e.g.* ’`city:_t population:_i`’) given unlabeled utterances (*e.g.* ‘*what **city** has the largest population*’).

The second strategy, *target generation*, attempts to improve generality of decoder network by generating more target-side data

[1] Beijing University of Posts and Telecommunications, China, email: {wangxiaolu, sunhaifeng_1, qiqi, wangjingyu}@ebupt.com;
* Corresponding author: wangxiaolu@ebupt.com

for semantic parsing training. The effectiveness of this strategy will depend on quality of generated data. We are inspired by the Mean Teacher [24] in Image Recognition, which tracks an exponential moving average of the whole model (student) weights as teacher to construct better predictions, and calculates distance between teacher and student output distributions as cost. Different from the original Mean Teacher, our strategy produces hard targets for original student decoder by running the teacher decoder and update the decoder weights in each training step instead of the whole model. When the student improves and updates the weights, the teacher improves in turn so that it can continually produce better target-side data.

Overall, the main contributions in this work are:

- We improve semantic parsing by enhancing not only the encoder representation, but also the decoder generality with extra unlabeled utterances in a novel multi-task learning framework.
- We alleviate missing and mismatching cases of key information and reduces wrong predictions by assigning entity lexicon learning as auxiliary task.
- We provide a novel application of the Mean Teacher framework on the task of semantic parsing and modify it for extending to sequence modelling.

We perform experiments in GEO, ATIS and JOBS. The results demonstrate our proposed methods significantly outperform supervised baseline. Furthermore, improvement is more impressive than previous methods.

2 Related work

Our work involves following threads of research. In this section, we sum up approaches from previous works.

Semantic Parsing Early work on semantic parsing [28, 16] relies on complex rules and pre-defined features. Recent years, the researchers follow trends of deep learning and apply the sequence to sequence model, attention mechanism [7], copy mechanism [12] and transfer learning [9], which have made progress in field of NLP and can be generally adapted to different domains and meaning representations.

The issue of limited labeled training data has received considerable critical attention and been addressed in the following works. [1] utilizes the denotations of meaning representations as indirect supervision in weakly-supervised learning; data recombination is induced in research of [12]. Another direction of research is domain adaptation, from multiple knowledge-bases [9] to multilingual sources [23], aiming at transferring information learned from related source domains to target domain by sharing parameters and capturing common linguistic features. Moreover, several attempts have been made in semi-supervised learning. [13] trains Support Vector Machine classifiers for every production in the meaning representation grammar. [2] applies dual learning in semantic parsing. [26] employs VAEs. [15] applies self-training to bootstrap an existing parser for AMR parsing. The semi-supervised learning with multi-task learning setup proposed here offers an alternative solution to this issue.

Multi-task learning for NLP Exploiting the shared information between different but related tasks, multi-task learning for NLP has been very popular since it was proposed by [5]. It has been used in various NLP applications, such as machine translation, machine reading comprehension and text classification. The majority of these multi-task architecture generally consists of lower layers shared across all tasks and top layers which are task-specific. Another line of work considers arranging tasks in linguistically-motivated hierarchies [10]. In this work, we follow the line of sharing parameters in lower layers and propose an auxiliary task, which provides information targeting towards semantic parsing, mitigating missing and mismatched cases of key information.

Semi-Supervised Learning for NLP In the application of NLP, semi-supervised learning has made fast progress with significant performance made by self-training [25], language modeling [6], autoencoder [3] and variational autoencoder [26]. Specifically, [21] attempts to enhance the decoder network model by incorporating the target-side monolingual data so as to boost the translation fluency. Work of [3] proposes to reconstruct the monolingual corpora using an auto-encoder for training NMT models. In [22], the weights of the encoder and decoder of a seq2seq model are initialized with the pre-trained weights of two language models and then fine-tuned with labeled data. Such methods can be easily applied to tasks with rich recourse on both source-side and target-side data. As for semantic parsing, we seek solutions with extra unlabeled source-side data in this work.

3 Methods

3.1 Overview

In this task, there are lots of natural language utterances $x^1, x^2, ..., x^I$. Each x^i is constructed by a sequence of natural language words $x^i = [x_1, ..., x_m]$. Some of the utterances has annotate with corresponding logic forms $y^1, y^2, ..., y^J$, where $J < I$ and y^j is constructed by a sequence of logic tokens $y^i = [y_1, ..., y_n]$. The labeled utterances and its logic form construct a labeled semantic parsing corpus $L = \{[x^n, y^n]\}_{n=1}^{N}$. The rest unlabeled utterances construct unlabeled corpus $UL = \{[x^m]\}_{m=1}^{M}$. Our goal is to enhance semantic parsing performance by fully exploring UL.

As is shown in Fig 2, we employee a multi-task learning framework to solve this task. It consists of one shared encoder and three attentional decoders, including semantic parsing decoder (SPDecoder), entity lexicon learning decoder (ELLDecoder) and target generation decoder (TGDecoder).

On the labeled corpus L, we train on the shared encoder and SPDecoder for the main task, semantic parsing. Once trained, we just use this part to do inference.

On the unlabeled corpus UL, we propose two strategies to improve the encoder and decoder network respectively. The first one enhances representation of the shared encoder by assigning entity predicates sequences as training targets in an auxiliary task, which we refer to as *entity lexicon learning*. The second one improves decoder network by extending Mean Teacher algorithm to generate more stable and better target for semantic parsing training, which we refer to as *target generation*.

3.2 Semantic parsing

We view the main task, semantic parsing, as a sequence transduction problem. We model it on the attention-based Encoder-Decoder architectures (i.e. Seq2seq) which have been successfully applied in neural semantic parsing. We implement the encoder as a bidirectional RNN and the decoder as another RNN, both with long short-term memory [11] units. The encoder encodes natural language sequence into a fixed-dimensional context vector as the representation, then the decoder generates logical forms based on the context vector with attention mechanism [19].

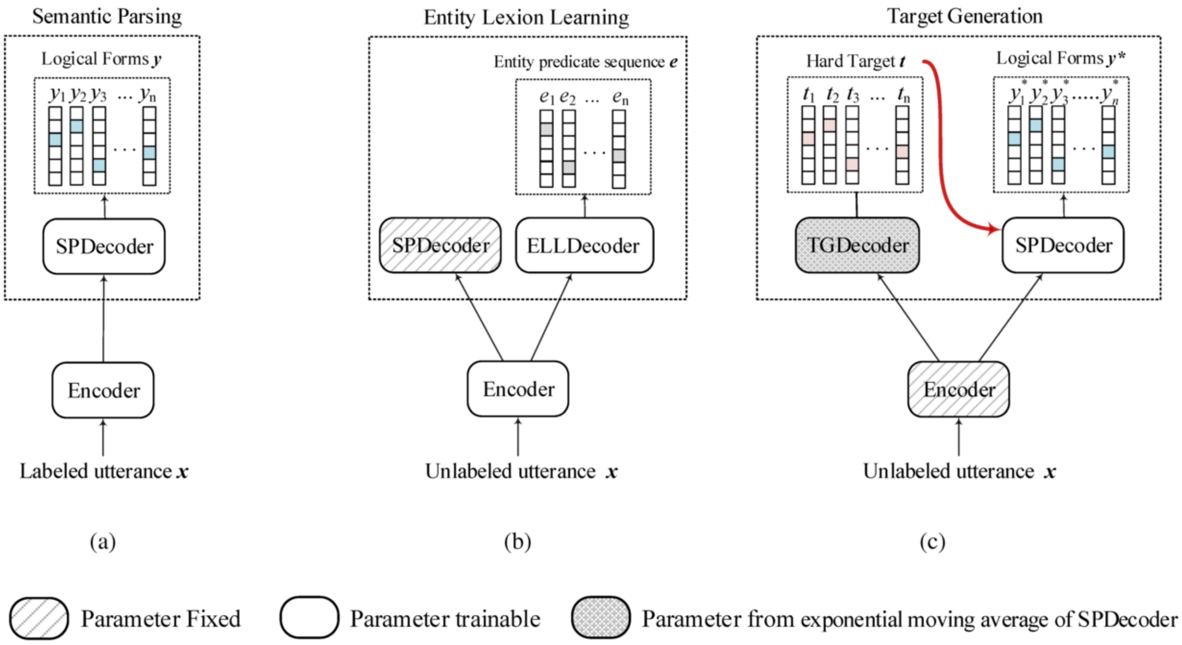

Figure 2: An overview of SETNet. (a) shows semantic parsing training on labeled dataset, (b) and (c) respectively illustrate our proposed strategies, *entity lexicon learning* and *target generation*, training on unlabeled dataset. y_i, t_i and y_i^* are logical token ids, while e_i indicates entity predicate id. They are all mapped from target vocabulary

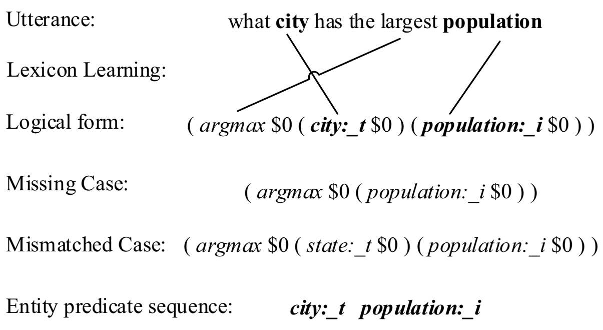
Utterance:
what city has the largest population
Lexicon Learning:
Logical form:
(argmax $0 (city:_t $0) (population:_i $0))
Missing Case:
(argmax $0 (population:_i $0))
Mismatched Case:
(argmax $0 (state:_t $0) (population:_i $0))
Entity predicate sequence:
city:_t population:_i

Figure 3: An example of exact match in semantic parsing, missing and mismatched cases, and assigned entity predicates sequence. The mappings between utterance and logical form represent lexicon learning.

3.3 Entity Lexicon Learning

To most unsupervised learning algorithm, such as sequence autoencoder [6], the motivation is given all consideration to the integrity of input and expects to provide the decoder with more comprehensive information. However, it focuses on more common in statistics but useless information instead of entity keywords.

Lexicon learning aims to learn the mapping from natural language words to predicates in pre-defined knowledge base, which is fundamental for exact match of semantic parsing. In this case, we focus on entity lexicon learning, which we define as capturing the **entity-type words** in the natural language utterance and mapping into corresponding **entity predicates** (*e.g.*, “city” :: `city:_t`). Different from other types of lexicons (*e.g.*, “largest” :: `argmax`), entity lexicons are pre-specific and rather static in a specific domain and can be easily learned using entity linking techniques instead of manual annotation. We could benefit this setting by two factors. First, it aims to force shared encoder's representations to pay more attention to the information targeting towards semantic parsing, mitigating missing and mismatched cases of key information. Second, the target outputs are logical tokens instead of natural language words, thus ELLDecoder can share the same embedding with other decoders.

As shown in Fig 2b, given the unlabeled corpus $UL = \{[x^m]\}_{m=1}^{M}$, we preprocess the corpus by identifying predifined entity-type words in knowledgebase and pair each sample $x = [x_1, ..., x_m]$ with ordered entity predicate sequence $e = [e_1, ..., e_n]$, where e_i refers to entity predicate ids mapped from V_t. As illustrated in Fig 3, natural language utterance ‘what *city* has the largest *population*’ is tagged with ‘`city:_t population:_i`’. The shared encoder computes representation of x, then the ELLDecoder predicts e from the representation.

On this strategy, we regard Entity Lexicon Learning as a standard and independent task. We freeze the parameters of other parts and just optimize the shared encoder and ELLDecoder by maximizing the log likelihood of e.

3.4 Target Generation

The key idea of Target Generation is that if logical form labels of good quality could be generated for the unlabeled utterances, the decoder network will be enhanced. Effectiveness of this strategy will highly depend on the quality of generated data. Targets computed from an exponential moving average of the model parameters tend to be better than ones generated by the original model [24].

As is shown in Fig 2c, our strategy follows three steps. First, we construct a teacher TGDecoder by tracking exponential moving average of SPDecoder parameters. We update parameters of TGDecoder as following:

$$\theta_{TGD}^{t} = \lambda\theta_{TGD}^{t-1} + (1-\lambda)\,\theta_{SPD}^{t} \tag{1}$$

where θ_{TGD}^{t} and θ_{SPD}^{t} refer to parameters of TGDecoder and SPDecoder at step t, λ is the decay parameter which controls the speed of model updating and its value usually approaches 1. This step is conducted when θ_{SPD} updates. Second, TGDecoder acts as a teacher and generates complete logical form targets. Since teacher forcing can't be implemented on account of the absence of ground truth at each time step, TGDecoder predicts hard target $t = [t_1, ..., t_n]$ with the unlabeled utterance $x = [x_1, ..., x_m]$ in every training step, where t_i is logical token id mapped from V_t. Finally, SPDecoder acts as a student at the same training step and learns from the teacher TGDecoder by generating logical forms $y^* = [y_1^*, ..., y_n^*]$ regarding t as ground truth, where y_i^* is logical token id mapped from V_t. Note that we apply dropout both in the second and the third steps.

Compared with constructing synthetic labeled data with self-training model [25], our strategy generates new labels for unlabeled utterances per epoch, and the quality of labels improve as the model updates.

4 Training Setting

In this part, we discuss the training objective of each task and how they combine in semi-supervised setting.

Training on Semantic Parsing Given a labeled corpus $L = \{[x^n, y^n]\}_{n=1}^{N}$, semantic parsing is trained in fully supervised manner. The loss function is as following:

$$L_{SP}(\theta_E, \theta_{SPD}, \theta_{ELLD}) = \frac{1}{N}\sum_{n=1}^{N} -logP(y^n | x^n, \theta_E, \theta_{SPD}) \tag{2}$$

where θ_E is a set of the shared encoder parameters, θ_{ELLD} and θ_{SPD} indicate parameters of ELLDecoder and SPDecoder.

Training on Entity Lexicon Learning Let $UL^{'} = \{[x^m, k^m]\}_{m=1}^{M}$ denote the dataset which is assigned with ordered entity predicate sequence for unlabeled data, the loss function is factorized as:

$$L_{ELL}(\theta_E, \theta_{SPD}, \theta_{ELLD}) = \frac{1}{M}\sum_{n=1}^{N} -logP(k^m | x^m, \theta_E, \theta_{ELLD}) \tag{3}$$

Training on Target Generation Let t^m denote output from running beam search with TGDecoder given x^m in $UL = \{[x^m]\}_{m=1}^{M}$, the loss function is formalized as:

$$L_{TG}(\theta_E, \theta_{SPD}, \theta_{ELLD}) = \omega(t) \cdot \frac{1}{M}\sum_{n=1}^{N} -logP(t^m | x^m, \theta_{SPD}) \tag{4}$$

where $\omega(t)$ is unsupervised loss weighting function for balancing the preference for this task. In the beginning, we loosen the constraint between the averaging model and the original model before meaningful prediction is obtained. In this stage, the function begins with zero and ramps up at a slow speed. After several training epochs, the training targets obtained by the averaging model can be expected to be significant and better than original model so that the function stays a larger value. When the network tends to stabilize, the advantage of training targets is not as obvious as before and sometimes maybe inferior compared with model prediction, then the function ramps down. Similar to [17], we choose a Gaussian curve to describe the dynamical weighting function. In this training procedure, we fix parameters of the shared encoder and don't do backpropagation through it.

Algorithm 1: Training Procedures

```
input  : Dataset $L, UL^{'}, UL$, Parameter $\theta_E, \theta_{SPD}, \theta_{ELLD}$
output: Parameter $\theta_E, \theta_{SPD}, \theta_{ELLD}$
$mode$ = 0
// for choose the auxiliary task
repeat
    for $epoch = 0, F$ do
        for $(x^n, y^n) \in L$ do
            Compute $L_{SP}$ by Equation (2)
            Update $\theta_E, \theta_{SPD}$ by gradient descent on $L_{SP}$
            Update $\theta_{TGD}$ by Equation (1)
    if $mode\%2 == 0$ then
        for $(x^m, k^m) \in UL^{'}$ do
            Compute $L_{ELL}$ by Equation (3)
            Update $\theta_E, \theta_{ELLD}$ by gradient descent on $L_{ELL}$
    else
        for $(x^m) \in UL$ do
            Compute $L_{TG}$ by Equation (4)
            Update $\theta_{SPD}$ by gradient descent on $L_{TG}$
            Update $\theta_{TGD}$ by Equation (1)
    $mode$++
    // change the auxiliary task
until model converge;
```

Training Procedure As shown in Algorithm 1, we train the model alternately on the main task semantic parsing and two auxiliary tasks, switching modes over at a particular frequency F. More specifically, after F times of semantic parsing training epochs, the model switches to one of the proposed tasks for one epoch. We only add one auxiliary task at a time and continue this process until the model converges.

5 Experiment

5.1 Dataset

Our model was evaluated on the following datasets.

GEO This dataset is a frequently used semantic parsing benchmark, including 880 queries to a U.S. geography database and their corresponding logical forms. The logical form takes the form of lambda-calculus expressions. We follow the practice in [27] and divide the training set into 600 examples for training and 280 examples for testing.

ATIS This dataset is about queries to a flight booking system, consisting of 5,410 examples. The dataset is split into 4434 examples for training, 448 examples for testing and 528 examples for developing. The logical forms also use lambda-calculus.

JOBS This dataset is about queries to a database of job listings. Questions are paired with Prolog-style queries. 640 examples are split into 500 training and 140 test instances.

In particular, we use argument identification [7] to replace entities and numbers with type names (*e.g.* company, degree, language, platform, location and job area in JOBS) and unique IDs. Moreover, we develop a simple procedure for identifying entity-type words and assigning entity predicate sequences to unlabeled natural language utterances, which compose the training set for Entity Lexicon Learning.

5.2 Setup

We select samples of the original dataset at random as the labeled training set and establish the unlabeled set with all natural language utterances in domain, which refers to the method in [26] to simulate the semi-supervised learning scenario.

Hyper-parameters of the supervised part follow the settings in [7]. In all experiments, we used pre-trained GloVe vectors with dimension 300. We use the RMSProp algorithm to train the model as [7] does. For semi-supervised setting, alternating frequency F is chosen from 1 to 15. Exponential moving average decay value λ is chosen from the set $\{0.9, 0.99, 0.999\}$. As previous semantic parsing researches did, the accuracy metric we use refers to the percentage of exact correct parsed outputs.

Our experiment consists of the following aspects. First, we compare our proposed model (SETNet) in semi-supervised manner using all the training data as the labeled set and unlabeled set with prior work in the literature. We conduct experiments on seq2seq model (SUP-SEQ2SEQ) described in Subsection 3.2 as our baselines. Then we compare SUP-SEQ2SEQ with SETNet and two ablation variants, *i.e.*, only with Entity Lexicon Learning (SETNet.ELL) and only with Target Generation (SETNet.TG), under the settings of different labeled sizes. Second, we discuss comparison between our proposed method with previous semi-supervised models. The semi-supervised frameworks proposed by previous works are various while not all can be applied to semantic parsing. We implement self-training (SELF-TRAIN, [29]), auto-encoder (AUTO-ENCODER) and cross-view training (CVT, [4]), and then train them with the same labeled data as our method. The AUTO-ENCODER is a standard MTL model, which has the same structure as SETNet.ELL and just replaces the target output with the input sequence itself, while SELF-TRAIN is the algorithm which learns from the model itself by following steps: 1) training a model with labeled examples until converging 2) assigning labels to unlabeled examples 3) mixing the generated data with original labeled data and training them as normal supervised training. As for CVT, we follow the design for seq2seq learning in [4]. In addition, we compare with results presented in SEQ4+, SEQ4- [14] on GEO, and STRUCTVAE [26] on ATIS. Since their results are all based on randomly sampled labeled data, we pay more attention to the improvement upon the same baseline model with the same amount of labeled data in these groups of comparison. Then we analyze how the proposed strategies work and explore impact of different hyper-parameters.

5.3 Comparison with previous work

The comparisons on GEO, ATIS and JOBS are listed on Tab 1-4.

Table 1: Previous supervised methods.

Method	ATIS	GEO	JOBS
ZC05 [27]	-	79.3	79.3
ZC07 [28]	84.6	86.1	-
UBL [16]	71.4	87.9	-
DCS+L [18]	-	87.9	90.7
TISP [30]	84.2	**88.9**	85.0
SEQ2SEQ [7]	84.2	84.6	87.1
SEQ2TREE [7]	84.6	87.1	90.0
ASN [20]	85.3	85.7	**91.4**
COARSE2FINE [8]	**87.7**	88.2	-
SETNet	87.2	85.4	89.3

Comparison with Supervised Methods As is shown in Tab 1, our method is competitive with previous neural network based methods on three datasets. Among all datasets, SETNet performs significantly better than SEQ2SEQ[7] by 3.0% on ATIS, 0.8% on GEO, and 2.2% on JOBS.

Comparison with Baseline First, SETNet outperforms the baseline all the time when there are extra unlabeled data disjoint with the labeled set. Using all labeled data in the dataset, we find that SETNet still has contributions to semantic parsing task. SETNet outperforms the baseline by 1.0% on ATIS, 0.8% on GEO and 0.7% on JOBS. In addition, it should be noted that the improvement is remarkable in the case of a small amount of labeled data. The best improvements achieved are respective 11.1% on GEO with 150 labeled examples, 5.8% on ATIS with 500 labeled examples and 19.3% on JOBS with 50 labeled examples. After that, the improvement falls steadily as the size of labeled data increases. This suggests that our method performs better in the setting of a small amount of labeled data and a relatively larger amount of unlabeled data. Among the three datasets, we find that the improvement in JOBS is more remarkable than GEO and ATIS. On JOBS, SETNet with 50 labeled examples even performs better than fully supervised SUP-SEQ2SEQ with 100 labeled examples.

Comparison with Semi-supervised Methods Among all 13 settings, where labeled size is smaller than size of the whole dataset, our method surpasses the three semi-supervised methods by 12 settings in comparison with SELF-TRAIN, 13 settings for in comparison with AUTO-ENCODER and 13 settings in comparison with CVT. On GEO dataset, SETNet improves more on baseline than SEQ4+ in all 6 settings, while than SEQ4- in 5 of 6 settings. On ATIS dataset, the improvement in SETNet is larger than in STRUCTVAE-SEQ in 3 settings of all 5 settings.

5.4 Performance of proposed strategies

We explore how proposed strategies work respectively through the training curves.

Study on Entity Lexicon Learning By comparing AUTO-ENCODER with the ablation variant SETNet.ELL, we find that AUTO-ENCODER gets a small boost than SETNet.ELL. In Fig 4, we define the error rate as the proportion of wrong predictions caused by miss or mismatch of entity predicates. The examples are shown in Fig 3. We find that in the training process, SETNet.ELL has the stronger ability to capture key information in utterances and complete the expression of logical forms, while SUP-SEQ2SEQ converges faster than the others. By contrast, AUTO-ENCODER is even harmful to address key information.

Table 2: Evaluation result on ATIS w.r.t. the size of labeled training data. The numbers in brackets represent the improvements relative to the performance of supervised baseline.

Method	**ATIS**				
	500	**1000**	**2000**	**3000**	**4434**
SUP-SEQ2SEQ	58.2	74.8	79.5	84.2	86.2
SELF-TRAIN	57.1 (−1.1)	75.0 (+0.2)	82.4 (+2.9)	84.7 (+0.5)	-
AUTO-ENCODER	60.7 (+2.5)	75.2 (+0.4)	80.0 (+0.5)	84.3 (+0.1)	-
CVT	60.1 (+1.9)	75.9 (+1.1)	79.3 (−0.2)	80.0 (−1.2)	-
SETNet	64.0 (+5.8)	79.2 (+4.4)	82.0 (**+2.5**)	85.3 (+1.1)	87.2 (+1.0)
- SETNet.TG	61.6 (+3.4)	76.1 (+1.3)	81.5 (+2.0)	83.7 (−0.5)	87.5 (**+1.3**)
- SETNet.ELL	61.5 (+3.3)	77.8 (+3.0)	81.4 (+1.9)	85.5 (**+1.3**)	86.1 (−0.1)
Previous Method					
SUP_SEQ2SEQ [26]	47.3	62.5	73.9	80.6	84.6
Structvae-SEQ [26]	55.6 (**+8.3**)	73.1 (**+10.6**)	74.8(+0.9)	81.3(+0.7)	84.2 (−0.4)

Table 3: Evaluation result on GEO w.r.t. the size of labeled training data. The numbers in brackets represent the improvements relative to the performance of supervised baseline.

Method	**GEO**					
	30	**60**	**150**	**300**	**450**	**600**
SUP-SEQ2SEQ	18.2	43.7	60.0	75.7	80.7	84.6
SELF-TRAIN	19.1 (+0.9)	43.8 (+0.1)	59.6 (−0.4)	74.9 (−0.8)	80.4 (−0.3)	-
AUTO-ENCODER	19.6 (+1.4)	45.1 (+1.4)	60.7 (+0.7)	76.1 (+0.4)	81.7 (+1.0)	-
CVT	18.6 (+0.4)	42.4 (−1.3)	60.7 (+0.7)	75.9 (+0.2)	80.4 (−0.3)	-
SETNet	26.0 (+7.8)	49.1 (**+5.4**)	71.1 (**+11.1**)	82.1 (**+6.4**)	83.3 (**+2.6**)	85.4 (**+0.8**)
- SETNet.TG	20.8 (+2.6)	46.1 (+2.4)	61.4 (+1.4)	77.9 (+2.2)	82.1 (+1.4)	85.4 (+0.8)
- SETNet.ELL	26.2 (+8.0)	47.7 (+4.0)	69.9 (+9.9)	79.3 (+3.6)	81.1 (+0.4)	84.6 (+0.0)
Previous Method						
SUP_SEQ2SEQ [14]	21.9	39.7	62.4	80.3	85.3	86.5
SEQ4+[14]	26.2 (+4.3)	42.1 (+2.4)	67.1 (+4.7)	80.4 (+0.1)	85.1 (−0.2)	87.2 (+0.8)
SEQ4- [14]	30.1 (**+8.2**)	42.1 (+2.4)	70.4 (+8.0)	81.2 (+0.9)	84.1 (+1.2)	86.5 (+0.0)

Table 4: Evaluation result on JOBS w.r.t. the size of labeled training data. The numbers in brackets represent the improvements relative to the performance of supervised baseline.

Method	**JOBS**				
	50	**100**	**250**	**400**	**500**
SUP-SEQ2SEQ	51.4	64.3	77.1	82.1	88.6
SELF-TRAIN	51.4 (+0.0)	59.9 (−4.4)	77.9 (+0.8)	82.1 (+0.0)	-
AUTO-ENCODER	55.0 (+3.6)	65.7 (+1.4)	78.6 (+1.5)	81.4 (−0.7)	-
CVT	52.5 (+1.1)	65.6 (+1.3)	77.7 (+0.6)	80.3 (−1.8)	-
SETNet	70.7 (**+19.3**)	72.1 (**+7.8**)	80.8 (**+3.7**)	87.9 (**+5.8**)	89.3 (**+0.7**)
- SETNet.TG	58.6 (+7.2)	67.9 (+3.6)	79.3 (+2.2)	83.6 (+1.5)	88.2 (−0.4)
- SETNet.ELL	66.4 (+15.0)	70.0 (+5.7)	80.8 (+3.7)	85.7 (+3.6)	89.3 (+0.7)

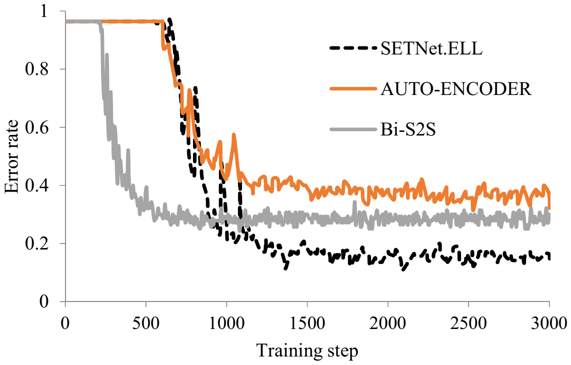

Figure 4: Error rate about miss and mismatch key information with 50 labeled example on Jobs.

Study on Target Generation As shown in Fig 5, in early training steps, the teacher decoder in SETNet.TG makes more accurate predictions than the student decoder at the most. Though the difference is small, we can find that the student has learned from the teacher and improved the performance in comparison with SUP-SEQ2SEQ. When the model tends to converge, the teacher and student performance better alternately.

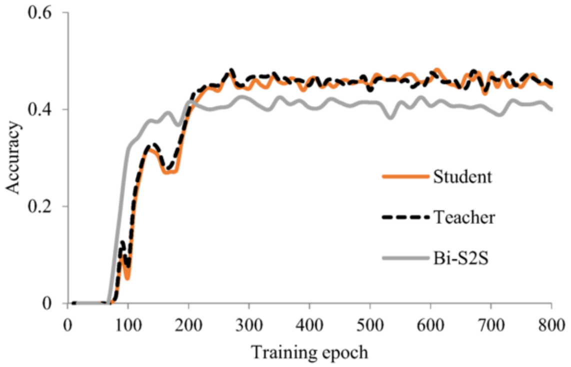

Figure 5: Performance of SUP-SEQ2SEQ, teacher and student decoder on SETNet.TG with 50 labeled example on Jobs.

Comparison of Two Strategies It's clear to see that both strategies have served their purposes and improved performance to varying degrees. SETNet.ELL outperforms baseline in all 13 settings where labeled size is smaller than size of the whole dataset, while SETNet.TG outperforms in 12 of 13 settings. In most settings, SETNet.ELL contributes more improvements than SETNet.TG. The biggest gap between them is 7.8% in the setting of 50 labeled examples on JOBS. This demonstrates that our method contributes more to the encoder representation learning than the decoder network. In addition, on GEO and JOBS, the improvement of SETNet.ELL declines considerably as the size of labeled data increases, while SETNet.TG remains relatively stable. For example, the biggest gaps between settings on GEO are 9.5% for SETNet.ELL and 1.2% for SETNet.TG. While on ATIS, the difference is not obvious.

5.5 Effects of hyper-parameters

Table 5: Performance of SETNet with 50 labeled examples on JOBS w.r.t. the size of unlabeled data

Size	0	50	100	250	400	500
Acc	51.4	57.1	57.9	65.7	68.6	70.7

Table 6: Performance of SETNet.TG with 50 labeled examples on JOBS w.r.t. the value of exponential moving average decay in the beginning

EMA decay	-	0.9	0.99	0.999
Acc	51.4	58.6	55.8	50.3

We discuss how unlabeled size influences model performance.

Effect of unlabeled size It's easy to find that model performance improves with increasing size of unlabeled data, as is shown in Tab 5. This demonstrates once again that our method can learn from unlabeled utterances and the more, the better.

Effect of exponential moving average decay Tab 6 shows the impact of exponential moving average decay we set in the beginning. It indicates that we should loosen the constraint between the averaging model and the original model before meaningful prediction is obtained. When we use 0.999 as the value of decay, the teacher decoder remembers the old and inaccurate student weights.

6 Conclusions

In this work, we study on semi-supervised semantic parsing and propose two strategies to leverage unlabeled natural language utterances, one for entity lexicon earning and the other for target generation. Experiments show that with unlabeled utterances, it performs better than in fully supervised manner and achieves more impressive improvement than previous semi-supervised methods.

ACKNOWLEDGEMENTS

This work was supported in part by the National Key R&D Program of China 2018YFB1800502, in part by the National Natural Science Foundation of China under Grants 61671079 and 61771068, in part by the Beijing Municipal Natural Science Foundation under Grant 4182041, in part by the Ministry of Education and China Mobile Joint Fund MCM20180101, and in part by State Grid Corporation of China Project (SGTJDK00DWJS1900242) Research and application of key technologies of knowledge management and cognitive reasoning in operation inspection field for ubiquitous electric internet of things.

REFERENCES

[1] Priyanka Agrawal, Ayushi Dalmia, Parag Jain, Abhishek Bansal, Ashish R. Mittal, and Karthik Sankaranarayanan, 'Unified semantic parsing with weak supervision', in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 4801–4810, (2019).

[2] Ruisheng Cao, Su Zhu, Chen Liu, Jieyu Li, and Kai Yu, 'Semantic parsing with dual learning', in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 51–64, (2019).

[3] Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu, 'Semi-supervised learning for neural machine translation', 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*, (2016).

[4] Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc V. Le, 'Semi-supervised sequence modeling with cross-view training', in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 1914–1925, (2018).

[5] Ronan Collobert and Jason Weston, 'A unified architecture for natural language processing: deep neural networks with multitask learning', in *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, pp. 160–167, (2008).

[6] Andrew M. Dai and Quoc V. Le, 'Semi-supervised sequence learning', in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 3079–3087, (2015).

[7] Li Dong and Mirella Lapata, '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*, (2016).

[8] Li Dong and Mirella Lapata, '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*, pp. 731–742, (2018).

[9] Xing Fan, Emilio Monti, Lambert Mathias, and Markus Dreyer, 'Transfer learning for neural semantic parsing', in *Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017*, pp. 48–56, (2017).

[10] Yu Gong, Xusheng Luo, Yu Zhu, Wenwu Ou, Zhao Li, Muhua Zhu, Kenny Q. Zhu, Lu Duan, and Xi Chen, 'Deep cascade multi-task learning for slot filling in online shopping assistant', in *AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 6465–6472, (2019).

[11] Sepp Hochreiter and Jürgen Schmidhuber, 'Long short-term memory', *Neural Computation*, **9**(8), 1735–1780, (1997).

[12] Robin Jia and Percy Liang, 'Data recombination for neural semantic parsing', 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*, (2016).

[13] Rohit J. Kate and Raymond J. Mooney, 'Semi-supervised learning for semantic parsing using support vector machines', in *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, April 22-27, 2007, Rochester, New York, USA*, eds., Candace L. Sidner, Tanja Schultz, Matthew Stone, and ChengXiang Zhai, pp. 81–84. The Association for Computational Linguistics, (2007).

[14] Tomás Kociský, Gábor Melis, Edward Grefenstette, Chris Dyer, Wang Ling, Phil Blunsom, and Karl Moritz Hermann, 'Semantic parsing with semi-supervised sequential autoencoders', in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 1078–1087, (2016).

[15] Ioannis Konstas, Srinivasan Iyer, Mark Yatskar, Yejin Choi, and Luke Zettlemoyer, 'Neural AMR: sequence-to-sequence models for parsing and generation', 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*, pp. 146–157, (2017).

[16] Tom Kwiatkowski, Luke S. Zettlemoyer, Sharon Goldwater, and Mark Steedman, 'Inducing probabilistic CCG grammars from logical form with higher-order unification', in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP 2010, 9-11 October 2010, MIT Stata Center, Massachusetts, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1223–1233, (2010).

[17] Samuli Laine and Timo Aila, 'Temporal ensembling for semi-supervised learning', in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, (2017).

[18] Percy Liang, Michael I. Jordan, and Dan Klein, 'Learning dependency-based compositional semantics', *Computational Linguistics*, **39**(2), 389–446, (2013).

[19] Thang Luong, Hieu Pham, and Christopher D. Manning, 'Effective approaches to attention-based neural machine translation', in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pp. 1412–1421, (2015).

[20] Maxim Rabinovich, Mitchell Stern, and Dan Klein, 'Abstract syntax networks for code generation and semantic parsing', 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*, pp. 1139–1149, (2017).

[21] Rico Sennrich, Barry Haddow, and Alexandra Birch, 'Improving neural machine translation models with monolingual data', 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*, (2016).

[22] Ivan Skorokhodov, Anton Rykachevskiy, Dmitry Emelyanenko, Sergey Slotin, and Anton Ponkratov, 'Semi-supervised neural machine translation with language models', in *Proceedings of the Workshop on Technologies for MT of Low Resource Languages, LoResMT@AMTA 2018, Boston, MA, USA, March 21, 2018*, pp. 37–44, (2018).

[23] Raymond Hendy Susanto and Wei Lu, 'Neural architectures for multilingual semantic parsing', in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pp. 38–44, (2017).

[24] Antti Tarvainen and Harri Valpola, 'Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results', in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*, (2017).

[25] Hai Ye and Lu Wang, 'Semi-supervised learning for neural keyphrase generation', in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 4142–4153, (2018).

[26] Pengcheng Yin, Chunting Zhou, Junxian He, and Graham Neubig, 'Structvae: Tree-structured latent variable models for semi-supervised semantic parsing', in *ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 754–765, (2018).

[27] Luke S. Zettlemoyer and Michael Collins, 'Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars', in *UAI '05, Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence, Edinburgh, Scotland, July 26-29, 2005*, pp. 658–666, (2005).

[28] Luke S. Zettlemoyer and Michael Collins, 'Online learning of relaxed CCG grammars for parsing to logical form', in *EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic*, pp. 678–687, (2007).

[29] Jiajun Zhang and Chengqing Zong, 'Exploiting source-side monolingual data in neural machine translation', in *EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 1535–1545, (2016).

[30] Kai Zhao and Liang Huang, 'Type-driven incremental semantic parsing with polymorphism', in *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pp. 1416–1421, (2015).