A JOINT MULTI-TASK LEARNING FRAMEWORK FOR SPOKEN LANGUAGE UNDERSTANDING

Changliang Li[1], Cunliang Kong[1,2], Yan Zhao[1,2]

[1]Institute of Automation, Chinese Academy of Sciences
[2]Beijing Language and Culture University

ABSTRACT

Spoken language understanding (SLU), which mainly involves intent prediction and slot filling, is a core component of a spoken dialogue system. Usually, intent determination and slot filling are carried out independently. Recently, joint learning of intent determination and slot filling has been proved effective in SLU. In this paper, we propose a novel joint multi-task learning framework for SLU, which predicts user intent and slot label via shared LSTM architecture, as well as next word's part of speech (POS) via neural language model. The proposed model exploits the correlation among different tasks and makes full advantage of all supervised signals. We conduct experiments on popular benchmark ATIS dataset, which consists of rich dialogues collected from real world. The experiment results show that our model achieves state-of-the-art in terms of several popular metrics.

Index Terms— Spoken language understanding, language model, jointly learning, part of speech

1. INTRODUCTION

As a crucial part of spoken dialogue system, SLU is used to understand semantic meanings of user utterances. To achieve this goal, there are two basic tasks in SLU, which can be explained as extracting semantic constituents in user utterances and identifying intents of users. The former is called slot filling, and the latter is called intent determination.

Slot filling can be considered as a sequence labeling problem, and intent determination can be considered as a classification problem. Usually, intent determination and slot filling are carried out separately. However, separate modeling of these two tasks is constrained to take full advantage of all supervised signals. Since these two tasks usually appear simultaneously in SLU systems, and both are based on understanding of the semantic meanings in user utterances, they may share the same information. Therefore, we believe that it is a good method to perform intent determination and slot filling jointly.

In this paper, we propose a joint multi-task learning framework for SLU that performs intent determination, slot filling and POS prediction. We combine SLU module with a variation of language model to predict the next word's POS tag. The intent class and slot label of current word are employed during POS prediction process. We use LSTM as basic recurrent unit to represent semantic information in user utterances. And the semantic information is shared in all three tasks. Via joint training, correlation among different tasks is exploited, and different tasks can promote each other. Furthermore, with this mechanism, additional linguistic information can be introduced into the model and brings improvement. To verify the proposed model, we conducted several experiments on ATIS benchmark. The experiment results show that our model outperforms former models in terms of several popular metrics.

2. RELATED WORK

Since SLU has a long research history, there are many prior work trying to fulfill this task.

Traditional machine learning approaches usually deal with these two problems separately. For slot filling, hidden Markov models (HMMs) and conditional random field (CRF) are widely used [2-5]. [2] formulated several statistical models for SLU in the literature as extensions to HMMs as segment models. [3] presented a new Markovian sequence model, which allows observations to be represented as arbitrary overlapping feature and defines the conditional probability of state sequences given observation sequences. [4] evaluated discriminative algorithms based on Support Vector Machine sequence classifier and Conditional Random Fields (CRF) carried on two SLU tasks. For intent determination, maximum entropy and support vector machine with linear kernel (LinearSVM) are applied in many models. [6] presented a series of experiments on speech utterance classification, and compared the performance of n-gram classifiers with that of Naive Bayes and maximum entropy classifiers. [7] proposed a global optimization process based on an optimal channel communication model that allows a combination of possibly heterogeneous binary SVM classifiers.

Due to the need of feature engineering, traditional approaches are demanding and time-consuming. The application of deep learning approaches in SLU solved this problem, and has made lots of achievements [8-15]. [8]

compared a DBN-initialized neural network to three widely used traditional text classification algorithms, and the DBN-based model gives a call-routing classification accuracy that is equal to the best of other models. [9] introduced a model called knowledge-guided structural attention networks (K-SAN) which is a generalization of RNN to additionally incorporate non-flat network topologies guided by prior knowledge. [12] proposed to use RNN for slot filling task, and deep convex network (DCN) was used for intent determination in [15].

Recently, many neural network-based models are proposed to jointly perform intent determination and slot filling, since both slot tags and intents are semantic representations of user utterances and may share the same knowledge [16-22]. [17] proposed a joint model for intent determination and slot filling using triangular CRF based on convolutional neural networks (CNN). [18] used recursive neural network to perform intent determination and slot filling jointly. [19] proposed an end-to-end deep recurrent neural network with limited contextual dialogue memory by jointly training natural language understanding (NLU) and system action prediction (SAP). [1] implemented a RNN based SLU model added with a language model that jointly performs intent determination, slot filling and language modeling, and received effective results.

3. METHOD

As is shown in figure 1, the proposed model consists of five layers: embedding layer, LSTM layer, NLU module that consists of slot filling layer and intent determination layer, and POS prediction layer. In this section, we will introduce these layers in details.

3.1 Embedding Layer

Embedding layer is employed to map input words to vector space as word embeddings. Given a sequence of input words $\mathbf{w} = (w_0, \dots\dots, w_{T+1})$, where w_0 and w_{T+1} are the beginning-of-sentence (<BOS>) and end-of-sentence (<EOS>) tokens, we use this layer to represent the sequence of words as a sequence of vectors $\mathbf{v} = (v_0, \dots\dots, v_{T+1})$.

3.2 LSTM Layer

At each time step, we use the LSTM layer to encode the information of all history words, intents and slot labels seen previously. These information is concatenated as one vector as following equation:

$$x_t = [v_t, y^i_{t-1}, y^s_{t-1}] \tag{1}$$

where y^i_{t-1} and y^s_{t-1} are true intent and slot labels respectively at time step $t-1$, v_t is embedded word vector at time step t.

With current input vector x_t, the LSTM unit can be further expanded as,

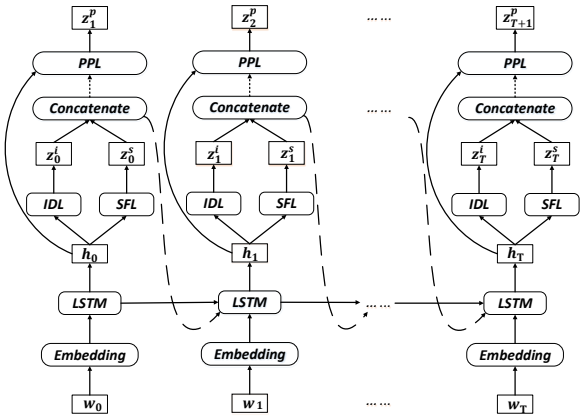

Fig. 1. Overview of our model. SFL means slot filling layer, IDL means Intent determination layer, PPL means POS prediction layer. The dotted line means local context information. The dashed line means recurrent context information.

$$\begin{cases} f_t = \text{sigmoid}(W_f h_{t-1} + U_f x_t + b_f) \\ i_t = \text{sigmoid}(W_i h_{t-1} + U_i x_t + b_i) \\ g_t = \tanh(W_c h_{t-1} + U_c x_t + b_c) \\ c_t = f_t \odot c_{t-1} + i_t \odot g_t \\ o_t = \text{sigmoid}(W_o h_{t-1} + U_o x_t + b_o) \\ h_t = o_t \odot \tanh(c_t) \end{cases} \tag{2}$$

where W, U, b are trainable parameters; $\odot$ is an element-wise product; h_t is current hidden state, h_{t-1} is the previous hidden state.

At time step t, we obtain the hidden state h_t as following equation:

$$h_t = \text{LSTM}(h_{t-1}, x_t) \tag{3}$$

3.3 NLU Module

The NLU module is comprised of slot filling layer and intent determination layer. We now describe these two layers detailly.

3.3.1 Slot Filling Layer

Since we have the encoded information h_t, we use it as input to generate a slot label for current word.

We use a multilayer feedforward neural network for mapping h_t to M-dimensions, where M is the number of all the slot labels. Then we use a softmax function to get the probability distribution over all M-dimensions, and choose the maximum dimension as the predicted slot label.

$$\begin{cases} d^s_t = \text{softmax}(f^s(h_t)) \\ z^s_t = e^s(\text{argmax}(d^s_t)) \end{cases} \tag{4}$$

where f^s is a multilayer feedforward neural network, d^s_t is a probability distribution of the predicted slot label, e^s means embedding lookup process in slot labels' vector space, z^s_t means predicted slot label at time step t.

3.3.2 Intent Determination Layer

The intent determination layer aims to generate an intent class for each word. Since our intent determination is on sentence level, we choose the intent for last word in the input sequence as the final intent class. With the similar architecture as slot filling layer, intent labels can be predicted as:

$$\begin{cases} d_t^i = \text{softmax}(f^i(h_t)) \\ z_t^i = e^i(\text{argmax}(d_t^i)) \end{cases} \quad (5)$$

where f^i is a multilayer feedforward neural network, d_t^i is a probability distribution of the predicted intent, e^i means embedding lookup process in intents' vector space, z_t^i means predicted intent at time step t.

3.4 POS Prediction Layer

This layer is a variation of RNN language model. Traditional RNN language model, which uses history information to predict the next word, has a defectiveness that it's hard to predict the next word accurately all the time due to the large vocabulary. To address this problem, we propose to predict the next word's POS tag. Let $\boldsymbol{y^p} = (y_1^p, \dots\dots, y_{T+1}^p)$ be the sequence of true POS tags. Using the chain rule, the probability of $\boldsymbol{y^p}$ can be illustrated as:

$$P(\boldsymbol{y^p}|\mathbf{w}) = \prod_{t=0}^{T} P(y_{t+1}^p | w_{\leq t}, y_{\leq t}^i, y_{\leq t}^s) \quad (6)$$

where $w_{\leq t}$ are previous words before time step t, $y_{\leq t}^i$ are previous intents before time step t, $y_{\leq t}^s$ are previous slot labels before time step t.

Similar with intent determination layer and slot filling layer, POS tag of next word can be predicted as:

$$\begin{cases} d_{t+1}^p = \text{softmax}(f^p([h_t, y_t^i, y_t^s])) \\ z_{t+1}^p = e^p(\text{argmax}(d_{t+1}^p)) \end{cases} \quad (7)$$

where f^p is a multilayer feedforward neural network, y_t^i is the true intent, y_t^s is the true slot label, d_{t+1}^p is a probability distribution of the predicted POS tag, e^i means embedding lookup process in POS tags' vector space, z_{t+1}^p means predicted next word's POS tag at time step $t+1$.

3.5 Joint Training

To measure discrimination between the true labels and predicted labels, we use cross-entropy as loss function to train the model. For slot filling, intent determination and POS prediction, the loss functions are as follows:

$$\begin{cases} \mathcal{L}^s = -P(\boldsymbol{y^s}) \log P(\boldsymbol{d^s}) \\ \mathcal{L}^i = -P(\boldsymbol{y^i}) \log P(\boldsymbol{d^i}) \\ \mathcal{L}^p = -P(\boldsymbol{y^p}) \log P(\boldsymbol{d^p}) \end{cases} \quad (8)$$

where $P(\boldsymbol{y^s})$, $P(\boldsymbol{y^i})$, $P(\boldsymbol{y^p})$ are probability distributions of sequences of true slot labels, intents and POS tags respectively; $P(\boldsymbol{d^s})$, $P(\boldsymbol{d^i})$, $P(\boldsymbol{d^p})$ are probability distributions of sequences of predicted slot labels, intents and POS tags respectively.

The entire model can be jointly trained with the total loss function in below:

$$\mathcal{L} = \sum_{\mathcal{D}} [\mathcal{L}^s + \mathcal{L}^i + \mathcal{L}^p] - \lambda R(\theta) \quad (9)$$

where θ is the set of parameters of three feedforward neural networks, $R(\theta)$ is an $L2$ regularization term used on parameter set θ, $\mathcal{D}$ means the total training dataset.

Via joint learning with the unit loss functions, our model can efficiently exploit the correlation of the three tasks and make them promote each other.

3.6 Inference

There are some differences in training and inference stage. During training, true intents and slot labels can be used in LSTM layer and POS prediction layer. But during inference, we can only use predicted intents and slot labels in these two layers. However, the predicted intent classes during first few time steps are of lower confidence because of limited information. This may cause defective effects for LSTM layer and POS prediction layer.

To mitigate this problem, we set an intent contribution weight α that can be tuned dynamically. Additionally, we set a threshold k. At first k time steps, we don't use the intent class information and set α as 0. Then, we gradually increase α to 1 to raise the intent contribution in the model. We use the following formula to calculate α.

$$\alpha = \begin{cases} 0, & t \leq k \\ \frac{t-k}{T-k}, & otherwise \end{cases} \quad (10)$$

where t is current time step, T is the length of input sequence.

In this way, during inference, Eq. (3) in LSTM layer is modified as:

$$h_t = \text{LSTM}\ (h_{t-1}, [v_t, \alpha z_{t-1}^i, z_{t-1}^s]) \quad (11)$$

where z_{t-1}^i and z_{t-1}^s are the predicted intents and slot labels at time step $t-1$ respectively.

And Eq. (7) in POS layer is modified as:

$$\begin{cases} d_{t+1}^p = \text{softmax}(f^p([h_t, \alpha z_t^i, z_t^s])) \\ z_{t+1}^p = e^p(\text{argmax}(d_{t+1}^p)) \end{cases} \quad (12)$$

where z_t^i and z_t^s are the predicted intents and slot labels at time step t respectively.

4. EXPERIMENT

4.1 Data

ATIS (Airline Travel Information Systems) dataset is broadly used in SLU research. All dialogues in the dataset are derived from real world. It provides human being with real conversational abilities and linguistic expressions in real

Table 1. Intent detection error, slot filling f1 score, and language modeling perplexity on ATIS Test set. Line 1-2 are cited from [18]. Line 3-15 are cited from [1].

	Model	Intent Error	F1 Score	LM PPL
1	RecNN	4.60	93.22	-
2	RecNN+Viterbi	4.60	93.96	-
3	Independent training RNN intent model	2.13	-	-
4	Independent training RNN slot filling model	-	**94.91**	-
5	Independent training RNN language model	-	-	11.55
6	Basic joint training model	2.02	94.15	11.33
7	Joint model with *local* intent context	1.90	94.22	11.27
8	Joint model with *recurrent* intent context	1.90	94.16	10.21
9	Joint model with *local & recurrent* intent context	1.79	94.18	10.22
10	Joint model with *local* slot label context	1.79	94.14	11.14
11	Joint model with *recurrent* slot label context	1.79	94.64	11.19
12	Joint model with *local & recurrent* slot label context	1.68	94.52	11.17
13	Joint model with *local* intent + slot label context	1.90	94.13	11.22
14	Joint model with *recurrent* intent + slot label context	1.57	94.47	10.19
15	Joint model with *local & recurrent* intent + slot label context	1.68	94.45	10.28
16	SLU-LM-POS model with *recurrent* intent + slot label context	1.68	94.57	**2.89**
17	SLU-LM-POS model with *local & recurrent* intent + slot label context	**1.46**	94.81	2.92

application scenarios. This helps researchers to build an effective intelligent dialogue agent.

We follow the ATIS corpus used in [1]. The training set contains 4978 utterances from the ATIS-2 and ATIS-3 corpora, and the test set contains 893 utterances from the ATIS-3 NOV93 and DEC94 data sets. There are in total 127 distinct slot labels and 18 different intent types.

4.2 Results and Analysis

For comparison purpose, we used the same training configurations as work [1]. And we compare our model and other models using three metrics: intent error rate, slot filling f1 score, and language model perplexity.

Table 1 gives the performance of our model and reports results of other models cited from [1] and [18]. Our model is referred as SLU-LM-POS. According to whether using local information, we report two results as shown in line 16 and line 17. Recurrent information is used in both experiments.

Line 1 gives performance of joint intent and slot filling model using RNN [18]. Line 2 gives performance of the same model added with Viterbi sequence optimization for slot filling [18]. Line 3 to line 5 give the performance of the independent training model results on intent detection, slot filling, and language modeling. Line 6 to line 15 give the performances of SLU-LM model and its variations [1].

Comparing with all the models, our model achieves significant improvement. For SLU, our model outperforms previous best result 0.11% in term of intent error (compared with line 14), 0.17% in term of F1 score (compared with line 11). Although the F1 score is slightly lower than the independent slot filling model shown in line 4, our model still obtains the best result among all the other joint models. Since the joint model can provide great convenience of completing several tasks at the same time, we think this very tiny decline is inconsiderable.

Besides, the language model perplexity of our proposed model dropped from 10.19 to 2.89, a 71.6% relative error reduction, which is significantly improved in statistic meaning. This is easy to understand that in our model, we use POS prediction layer to predict the next word's POS tag. In this way, the vocabulary size is sharply reduced, so that the accuracy of prediction is significantly increased.

Additionally, the best results of SLU-LM model appeared on line 11 and 14. Both only use recurrent context but without local context. [1] speculates that the most useful information for the next word prediction can be well captured in the RNN state, and thus adding explicit dependencies on local intent class and slot label is not very helpful.

We found another interesting phenomenon that we can receive better results when we use both recurrent and local context to predict the next word's POS tag. According to our experiments, we believe that the reason of the improvement is the introduction of POS information in our model. By using the POS prediction layer, additional linguistic information can be merged into our model during back propagation. Hence the performance of our model for SLU can be improved significantly.

5. CONCLUSION

In this paper, we proposed a novel joint multi-task learning framework for SLU, which can be used to predict user intent and slot label. By sharing LSTM architecture and predicting next word's POS tag, our proposed model can make full use of the correlation among different tasks, and make them promote each other. On ATIS dataset, our proposed model gives robust performance and achieves state-of-the-art on multi tasks. One promising future work is to explore how different representation methods affect performance of model. Moreover, we intend to build a spoken dialogue system based on current work.

6. REFERENCES

[1] Bing Liu, and I. Lane. "Joint Online Spoken Language Understanding and Language Modeling With Recurrent Neural Networks." *Meeting of the Special Interest Group on Discourse and Dialogue* 2016:22-30.

[2] Ye-Yi Wang, L. Deng, and A. Acero. "Spoken language understanding." *Signal Processing Magazine IEEE* 22.5 2005:16-31.

[3] Andrew McCallum, D. Freitag, and F. C. N. Pereira. "Maximum entropy markov models for information extraction and segmentation." *Proc of Icml* 2000:591--598.

[4] Christian Raymond, and G. Riccardi. "Generative and discriminative algorithms for spoken language understanding. " *INTERSPEECH* 2007:1605--1608.

[5] John D. Lafferty, A. Mccallum, and F. C. N. Pereira. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data." *Eighteenth International Conference on Machine Learning* Morgan Kaufmann Publishers Inc. 2001:282-289.

[6] C. Chelba, M. Mahajan, and A. Acero. "Speech utterance classification." *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings* IEEE, 2003:I-280-I-283 vol.1.

[7] P. Haffner, G. Tur, and J. H. Wright. "Optimizing SVMs for complex call classification." IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings IEEE, 2003:I-632-I-635 vol.1.

[8] Ruhi Sarikaya, G. E. Hinton, and B. Ramabhadran. "Deep belief nets for natural language call-routing." *IEEE International Conference on Acoustics, Speech and Signal Processing* IEEE, 2011:5680-5683.

[9] Yun-Nung Chen, et al. "Knowledge as a Teacher: Knowledge-Guided Structural Attention Networks." *arXiv preprint arXiv: 1609.03286* 2016.

[10] Yun-Nung Chen, et al. "Syntax or semantics? knowledge-guided joint semantic frame parsing." *Spoken Language Technology Workshop*IEEE, 2016:348-355.

[11] Kai-Sheng Yao, et al. "Spoken Language Understanding using Long Short-Term Memory Neural Networks." *IEEE – Institute of Electrical & Electronics Engineers* 2014:189 - 194.

[12] Grégoire Mesnil, et al. "Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding." *IEEE/ACM Transactions on Audio Speech & Language Processing* 23.3 2015:530-539.

[13] Kai-Sheng Yao, et al. "Recurrent conditional random field for language understanding." *IEEE International Conference on Acoustics, Speech and Signal Processing IEEE*, 2014:4077-4081.

[14] Pu-Yang Xu, and R. Sarikaya. "Contextual domain classification in spoken language understanding systems using recurrent neural network." *IEEE International Conference on Acoustics, Speech and Signal Processing* IEEE, 2014:136-140.

[15] Gokhan Tur, et al. "Towards deeper understanding: Deep convex networks for semantic utterance classification." *IEEE International Conference on Acoustics, Speech and Signal Processing* IEEE, 2012:5045-5048.

[16] A. Bhargava, et al. "Easy contextual intent prediction and slot detection." *IEEE International Conference on Acoustics, Speech and Signal Processing* IEEE, 2013:8337-8341.

[17] Pu-Yang Xu, and R. Sarikaya. "Convolutional neural network based triangular CRF for joint intent detection and slot filling." *Automatic Speech Recognition and Understanding* IEEE, 2014:78-83.

[18] Daniel Guo, et al. "Joint semantic utterance classification and slot filling with recursive neural networks." *Spoken Language Technology Workshop* IEEE, 2015:554-559.

[19] Xuesong Yang, et al. "End-to-end joint learning of natural language understanding and dialogue manager." *IEEE International Conference on Acoustics, Speech and Signal Processing* IEEE, 2017:5690-5694.

[20] Kingma, Diederik P, and J. Ba. "Adam: A Method for Stochastic Optimization." *Computer Science* 2014.

[21] Yang-Yang Shi, et al. "Contextual spoken language understanding using recurrent neural networks." *IEEE International Conference on Acoustics, Speech and Signal Processing* IEEE, 2015:5271-5275.

[22] Yun-Nung Chen, et al. "End-to-End Memory Networks with Knowledge Carryover for Multi-Turn Spoken Language Understanding." *The Meeting of the International Speech Communication Association* 2016.