MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

# Context-Sensitive and Role-Dependent Spoken Language Understanding using Bidirectional and Attention LSTMs

Hori, C.; Hori, T.; Watanabe, S.; Hershey, J.R.

TR2016-074 September 2016

**Abstract**

To understand speaker intentions accurately in a dialog, it is important to consider the context of the surrounding sequence of dialog turns. Furthermore, each speaker may play a different role in the conversation, such as agent versus client, and thus features related to these roles may be important to the context. In previous work, we proposed context-sensitive spoken language understanding (SLU) using role-dependent long short-term memory (LSTM) recurrent neural networks (RNNs), and showed improved performance at predicting concept tags representing the intentions of agent and client in a human-human hotel reservation task. In the present study, we use bidirectional and attention-based LSTMs to train a roledependent context-sensitive model to jointly represent both the local word-level context within each utterance, and the left and right context within the dialog. The different roles of client and agent are modeled by switching between role-dependent layers. We evaluated label accuracies in the hotel reservation task using a variety of models, including logistic regression, RNNs, LSTMs, and the proposed bidirectional and attentionbased LSTMs. The bidirectional and attention-based LSTMs yield significantly better performance in this task.

*Interspeech*

# Context-Sensitive and Role-Dependent Spoken Language Understanding using Bidirectional and Attention LSTMs

*Chiori Hori, Takaaki Hori, Shinji Watanabe and John R. Hershey*

Mitsubishi Electric Research Laboratories
Cambridge MA, USA
{chori, thori, watanabe, hershey}@merl.com

## Abstract

To understand speaker intentions accurately in a dialog, it is important to consider the context of the surrounding sequence of dialog turns. Furthermore, each speaker may play a different role in the conversation, such as agent versus client, and thus features related to these roles may be important to the context. In previous work, we proposed context-sensitive spoken language understanding (SLU) using role-dependent long short-term memory (LSTM) recurrent neural networks (RNNs), and showed improved performance at predicting concept tags representing the intentions of agent and client in a human-human hotel reservation task. In the present study, we use bidirectional and attention-based LSTMs to train a role-dependent context-sensitive model to jointly represent both the local word-level context within each utterance, and the left and right context within the dialog. The different roles of client and agent are modeled by switching between role-dependent layers. We evaluated label accuracies in the hotel reservation task using a variety of models, including logistic regression, RNNs, LSTMs, and the proposed bidirectional and attention-based LSTMs. The bidirectional and attention-based LSTMs yield significantly better performance in this task.

**Index Terms**: spoken language understanding, context sensitive understanding, role-dependent model, Bidirectional LSTMs, Attention LSTMs

## 1. Introduction

Spoken dialog systems for Human-machine interfaces have been widely used for many applications such as smart phones and car navigation systems nowadays. Spoken language understanding (SLU) technologies in dialog systems have been intensively investigated to estimate the intention of user utterances obtained from an automatic speech recognition (ASR) system [1, 2]. Conventional intention estimation approaches for SLU are either based on phrase matching, or traditional classification methods such as boosting, support vector machines (SVM), and logistic regression (LR), using bag of word (BoW) features as inputs.

Recurrent neural networks (RNNs) have been more actively applied to utterance classification to consider history of a word sequence in each utterance [3, 4]. Furthermore, long short-term memory (LSTM) RNNs were applied to spoken language understanding [5]. However, these models were only used for word sequence context within an utterance without considering the broader context of the sequence of utterances. One might expect that the speaker intentions of each utterance can be more accurately inferred, especially in dialogs, if the context of the utterance within the dialog is also taken into account. This hypothesis appears to be borne out in previous work: context sensitive understanding using phrase matching, weighted finite state transducer-based dialog management (WFSTDM) was previously proposed [6]. More recently, conventional RNNs considering contextual information were applied to domain and intention classification [7], intention classification, and goal estimation [8] and system response generation [9]. LSTMs are a form of RNN designed to improve learning of long-range context, and have been shown to be effective for problems with complicated dependency like translation [10].

In the previous study, we applied LSTMs to capture long-term characteristics over an entire dialog [11]. Each word is input sequentially into a LSTM and concept tags are output at the end of each utterance. To propagate contextual information through a dialog, the activation vector of the LSTM for an utterance serves as input to the LSTM for the next utterance. The LSTMs were trained from a human-to-human dialog corpus annotated with concept tags which represent client and agent intentions for hotel reservation. Furthermore, each utterance has different expressions in terms of context of role-dependent features such as for task-oriented roles like agents and clients. In order to precisely model the role dependent expressions, we introduce two parallel LSTM layers representing client and agent expressions. We confirmed the LSTMs outperformed the other models such as LR, RNNs by considering the context and the proposed role-dependent context-sensitive LSTMs was better than LSTMs w/o role-dependent layers.

In this study, we use bidirectional and attention LSTMs to train role-dependent context-sensitive models for SLU. The bidirectional model uses forward and backward features of dialog context. Moreover, the attention model introduces an attention mechanism [12][13] to some specific portions in the context and infers the most probable concept tag by taking the utterance-level context into account. These extended models can be better than models using only forward contextual features if dialogs have relatively clear goals in a task-oriented scenario.

## 2. Context-sensitive SLU using LSTMs

The model we use for context-sensitive spoken language understanding is a recurrent neural network depicted in Fig. 1. The network has an input layer that takes each input word, a projection layer that reduces the word vector in a low-dimensional space, a hidden layer with recurrent connections that keeps context information, and an output layer that estimates posterior probabilities of output labels. In the hidden layer, we use a set of LSTM cells instead of regular network units. In theory, an LSTM cell can remember a value for an arbitrary length of time due to a system of gating. The LSTM cell contains input, forget, and output gates which determine when the input is significant enough to remember, when it should continue to remember or forget the value, and when it should contribute to the output value. An example of an LSTM cell is depicted in Fig. 1.

Suppose, given a sequence of $M$ utterances, $u_1, \ldots, u_\tau, \ldots, u_M$, each utterance consists of word sequence $w_{\tau,1}, \ldots, w_{\tau,t}, \ldots, w_{\tau,T_\tau}$ and its concept tag (or dialog act) $a_\tau$. The input vector $x_{\tau,t}$ is prepared as

$$x_{\tau,t} = \text{OneHot}(w_{\tau,t}), \tag{1}$$

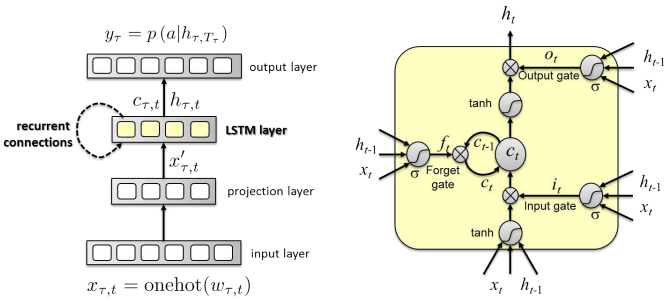


Figure 1: RNN (left) and LSTM cell (right).

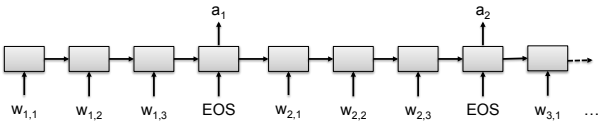


Figure 2: Propagation through time in context-sensitive SLU

where word $w_{\tau,t}$ in vocabulary $\mathcal{V}$ is converted by 1-of-N coding using function OneHot$(w)$, i.e. $x_{\tau,t} \in \{0,1\}^{|\mathcal{V}|}$.

The input vector is projected to the $D$ dimensional vector

$$x'_{\tau,t} = W_{pr} x_{\tau,t} + b_{pr} \tag{2}$$

and fed to the recurrent hidden layer, where $W_{pr}$ and $b_{pr}$ are the projection matrix and the bias vector.

At the hidden layer, activation vector $h_{\tau,t}$ is computed using the LSTM cells according to the way of [14][15], i.e.

$$i_{\tau,t} = \sigma(W_{xi} x'_{\tau,t} + W_{hi} h_{\tau,t-1} + b_i) \tag{3}$$
$$f_{\tau,t} = \sigma(W_{xf} x'_{\tau,t} + W_{hf} h_{\tau,t-1} + b_f) \tag{4}$$
$$c_{\tau,t} = f_{\tau,t} c_{\tau,t-1} + i_{\tau,t} \tanh(W_{xc} x'_{\tau,t} + W_{hc} h_{\tau,t-1} + b_c) \tag{5}$$
$$o_{\tau,t} = \sigma(W_{xo} x'_{\tau,t} + W_{ho} h_{\tau,t-1} + b_o) \tag{6}$$
$$h_{\tau,t} = o_{\tau,t} \tanh(c_{\tau,t}), \tag{7}$$

where $\sigma()$ is the element-wise sigmoid function, and $i_{\tau,t}$, $f_{\tau,t}$, $o_{\tau,t}$ and $c_{\tau,t}$ are the input gate, forget gate, output gate, and cell activation vectors for the $t$-th input word in the $\tau$-th utterance, respectively. The weight matrices $W_{zz}$ and the bias vectors $b_z$ are identified by the subscript $z \in \{x, h, i, f, o, c\}$. For example, $W_{hi}$ is the hidden-input gate matrix and $W_{xo}$ is the input-output gate matrix.

The output vector is computed at the end of each utterance as

$$y_\tau = \text{softmax}(W_{HO} h_{\tau,T_\tau} + b_O), \tag{8}$$

where $W_{HO}$ and $b_O$ are the transformation matrix and the bias vector to classify the input vector into different categories according to the hidden vector. softmax() is an element-wise softmax function that converts the classification result into label probabilities, i.e. $y_\tau \in [0,1]^{|\mathcal{L}|}$ for label set $\mathcal{L}$.

$$\hat{a}_\tau = \underset{a \in \mathcal{L}}{\text{argmax}}\, y_\tau[a], \tag{9}$$

where $y_\tau[a]$ indicates the component of $y_\tau$ for label $a$, which corresponds to label probability $P(a|h_{\tau,T_\tau})$.

To inherit the context information from the previous utterances, the hidden and cell activation vectors at the beginning of each utterance are equivalent to those at the final position $T_{\tau-1}$ in the previous utterance, i.e.,

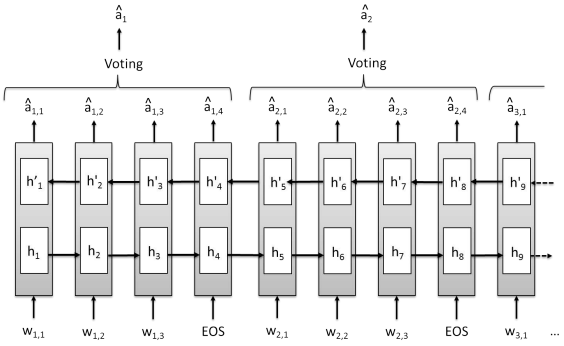


Figure 3: Prediction process using Forward and backward propagation through time in context-sensitive SLU

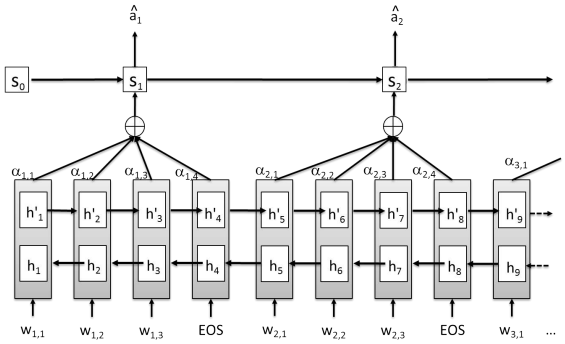


Figure 4: Prediction process using attention-based random access propagation through time in context-sensitive SLU

$$h_{\tau,0} = h_{\tau-1,T_{\tau-1}}, \quad c_{\tau,0} = c_{\tau-1,T_{\tau-1}}, \tag{10}$$

where $\tau > 1$ and $h_{1,0} = c_{1,0} = 0$.

Figure 2 illustrates a propagation process of our context-sensitive SLU. Words are sequentially input to the LSTM and an output label corresponding to the utterance concept at the end of the utterance, where symbol "EOS" stands for a sentence end. This model is similar to the LSTM used in [5] for SLU, but it considers the entire context from the beginning of the dialog, while the model in [5] considers each utterance independently. Accordingly, the label probabilities can be inferred using not only the sentence-level intentions but also the dialog-level context.

Next, we extend the LSTM-based SLU models to bidirectional networks. Figure 3 shows Prediction process using Forward and backward propagation. The backward propagation can be performed in the same way as the forward propagation through Eqs. (3)-(7) but in reverse order from the end of each utterance or the dialog.

Let $h'_{\tau,t}$ be the backward hidden activation vector. We can consider the both context by concatenating the forward and backward activation vectors as

$$\tilde{h}_{\tau,t} = \begin{bmatrix} h_{\tau,t} \\ h'_{\tau,t} \end{bmatrix}, \tag{11}$$

and infer the output label as

$$y_{\tau,t} = \text{softmax}(W_{HO} \tilde{h}_{\tau,t} + b_O). \tag{12}$$

Unlike the case of uni-directional networks, bidirectional models can use the both contextual information at any position of

each utterance. Accordingly, it is desirable that the concept tag for each utterance is decided based on the decision at every position in the utterance. To do this, we apply a majority voting scheme [16], i.e., the output label is chosen by

$$\hat{a}_\tau = \underset{a \in \mathcal{L}}{\text{argmax}} \sum_{t=1}^{T_\tau} \delta(\hat{a}_{\tau,t}, a), \tag{13}$$

using individual decisions

$$\hat{a}_{\tau,t} = \underset{a \in \mathcal{L}}{\text{argmax}}\, y_{\tau,t}[a], \tag{14}$$

where $\delta(\cdot)$ denotes Kronecker's delta.

Additionally, we further extend the bidirectional model with an attention mechanism [12][13]. With this extension, the model can pay attention to any portion of the input sequence by random access and use only important contextual information to predict the output label. This mechanism is realized by using *attention weights* to hidden activation vectors throughout the input sequence. Accordingly, limited but important phrases are emphasized with these weights.

Let $\alpha_{\tau,t}$ be an attention weight for the $t$-th word in the $\tau$-th utterance. A summary vector of the $\tau$-th utterance is obtained as a weighted sum of hidden activation vectors, i.e.

$$g_\tau = \sum_{t=1}^{T_\tau} \alpha_{\tau,t} \tilde{h}_{\tau,t}, \tag{15}$$

where $\tilde{h}_{\tau,t}$ is the bidirectional hidden activation vector given by Eq. (11). In this work, we limit the area of attention within each utterance to reduce the computation for summing up the hidden activations in Eq. (15).

As in [12], we prepare a decoder network that generates an output label sequence with summary vector $g_\tau$. The decoder network is also an LSTM-based RNN, where the decoder state can be obtained as

$$s_\tau = \text{lstm}(s_{\tau-1}, y_{\tau-1}, g_\tau), \tag{16}$$

where lstm() represents a function of LSTM layer in the decoder network, which can be computed with the same way as the process of Eqs. (3)-(7), where $h_{\tau,t}$ and $x'_{\tau,t}$ are replaced with $s_\tau$ and $[y_{\tau-1}^\intercal, g_\tau^\intercal]^\intercal$, respectively. Then, the output of the network is computed as

$$y_\tau = \text{softmax}(W_{HO} s_\tau + b_O) \tag{17}$$

and the output label is decided as in Eq. (9). Figure 4 illustrates the SLU process using the attention-based model.

On the other hand, the attention weights can be obtained as the same manner in [12]

$$\alpha_{\tau,t} = \frac{\exp(e_{\tau,t})}{\sum_{k=1}^{T_\tau} \exp(e_{\tau,k})} \tag{18}$$

and

$$e_{\tau,t} = w^\intercal \tanh(U s_{\tau-1} + V \tilde{h}_{\tau,t} + b), \tag{19}$$

where $w$ and $b$ are vectors, $U$ and $V$ are matrices, and $e_{\tau,t}$ is a scalar.

## 3. Role-dependent LSTM layers

In this study, the LSTMs were trained from a human-to-human dialog corpus annotated with concept tags which represent client and agent intentions for hotel reservation. The expressions are characterized by each role of agent and client. In order to precisely model the role dependent expressions, two parallel

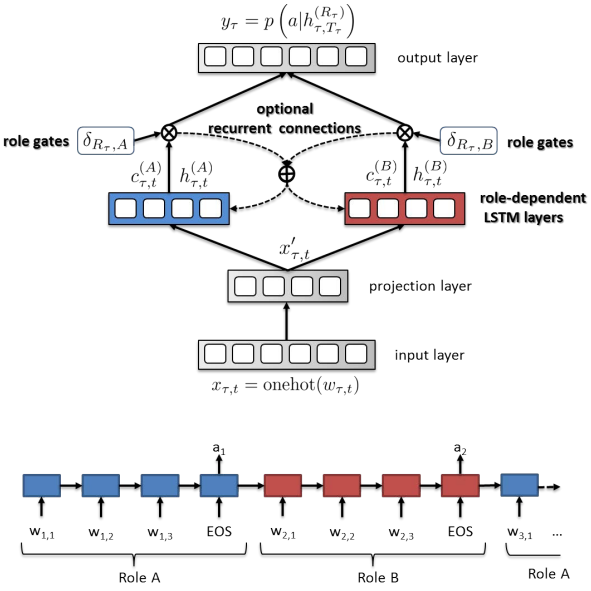


Figure 5: Upper: LSTM with role-dependent layers. Lower: Propagation through time in context-sensitive role-dependent SLU. The blue boxes correspond to client and the red boxes correspond to agent. Role gates control which role is active.

LSTM layers representing client and agent expressions are incorporated in the model as shown in Fig. 5.

The two LSTM layers have different parameters depending on the speaker roles. The input vector is thus processed differently by the left layer for the client's utterances, and by the right layer for the agent's utterances. The active role for a given utterance is controlled by a role variable $\mathcal{R}$, which is used to gate the output of each LSTM layer. The gated output then passes both to the recurrent LSTM inputs and to the output layer. The recurrent LSTM inputs thus receive the output from the role-dependent layer active at the previous frame, allowing for transitions between roles. Error signals in the training phase are also back-propagated through the corresponding layers. Here, we assume that the role of each speaker does not change during a dialog and it is known which speaker uttered which utterance. However, the model structure leaves open the possibility of dynamically inferring the roles. Accordingly, we can compute the activation at the output layer as

$$y_\tau = \text{softmax}\left(\delta_{\mathcal{R},R_\tau}(W_{HO} h^{(\mathcal{R})}_{\tau,T_\tau} + b_O)\right), \tag{20}$$

where $h^{(\mathcal{R})}_{\tau,T_\tau}$ is the hidden activation vector given by the LSTM layer of role $\mathcal{R}$, and $\delta_{\mathcal{R},R_\tau}$ is Kronecker's delta, i.e. if $R_\tau$ the role of the $\tau$-th utterance equals role $\mathcal{R}$, it takes 1, otherwise takes 0. Furthermore, at the beginning of each utterance, the hidden and cell activation vectors of the role-dependent layer are given as

$$h^{(R_\tau)}_{\tau,0} = h^{(R_{\tau-1})}_{\tau-1,T_{\tau-1}}, \qquad c^{(R_\tau)}_{\tau,0} = c^{(R_{\tau-1})}_{\tau-1,T_{\tau-1}}\ . \tag{21}$$

Figure 5 shows the temporal process of the role-dependent SLU. For each utterance in a given role, only the LSTM layer for that role is active, and the hidden activation and the cell memory are propagated over dialog turns. In the figure, the blue boxes correspond to client utterance states and the red boxes correspond to agent utterance states. With this architecture, the both layers can be trained considering a long context of each

Table 1: Label Accuracies. The numbers listed as w/ and w/o show the results of role-dependent and role-independent, respectively.

| | Dialog Act (DA) | | | | Dialog Act + Slot type (DA+ST) | | | |
|---|---|---|---|---|---|---|---|---|
| | Dev. set | | Test set | | Dev. set | | Test set | |
| Role-dependent layers | w/ | w/o | w/ | w/o | w/ | w/o | w/ | w/o |
| LR | – | 69.8 | – | 70.8 | – | 61.4 | – | 61.6 |
| LR + word2vec | – | 71.1 | – | 72.4 | – | 62.1 | – | 62.3 |
| Utterance-based LSTM | – | 73.3 | – | 69.8 | – | 59.5 | – | 56.2 |
| LSTM | **84.6** | 80.2 | **84.0** | 78.5 | **69.4** | 64.7 | **70.3** | 64.5 |
| Online-BLSTM | **83.7** | 83.3 | **86.9** | 82.8 | **72.6** | 68.8 | **72.6** | 69.1 |
| BLSTM | **85.8** | 83.5 | **86.8** | 84.2 | **72.6** | 72.6 | **72.0** | 72.0 |
| Online-Attention | **85.5** | 82.7 | **86.4** | 82.5 | **68.4** | 65.8 | **69.4** | 65.3 |
| Attention | **86.0** | 84.0 | **84.7** | 84.7 | **67.2** | 65.4 | **66.8** | 63.8 |

dialog, and the model can predict role-dependent concept labels more accurately.

This role-dependent extension can also be applied to the bidirectional LSTM networks and attention-based models described in the previous section. In the case of bidirectional LSTMs, output for each $t$-th word is given by

$$y_{\tau,t} = \text{softmax}\left(\delta_{\mathcal{R},R_\tau}(W_{HO}\tilde{h}^{(\mathcal{R})}_{\tau,t} + b_O)\right). \quad (22)$$

In the case of attention models, a summary vector is obtained based on role-dependent hidden activations, i.e

$$g^{(\mathcal{R})}_{\tau} = \sum_{t=1}^{T_\tau} \alpha^{(\mathcal{R})}_{\tau,t}\tilde{h}^{(\mathcal{R})}_{\tau,t}, \quad (23)$$

where $\alpha^{(\mathcal{R})}_{\tau,t}$ is determined by Eqs. (18) and (19) using the role-dependent activation $\tilde{h}^{(\mathcal{R})}_{\tau,t}$.

# 4. Experiments

## 4.1. Dialog Data

We trained models using a human-to-human dialog data annotated with concept tags representing client and agent intentions for hotel reservation [11]. In the experiments, Japanese utterances were used. 131 dialogs were split into 97 dialogs (5213 utterances) for training, 17 dialogs (1006 utterances) for development sets and 17 dialogs (1134 utterances) for test sets. The vocabulary size of the training data is 1550. The concept tags are based on Interchange Format (IF) which is an Interlingua for speech translation systems. The original tags indicate a combination of dialog acts, slot types and slot values. To model dialog discourses, two different layers of tags are used. One is a combination of dialog acts and slot types such as "request-information+room". The total number of the tags is 419 consisting of 186 client and 233 agent tags. The other one is dialog acts layer only such as "request-information" in which consists of 65 tags including 29 client and 36 agent tags.

## 4.2. Classifiers

We compared label accuracies using Logistic Regression (LR), LSTMs, BLSTMs, and attention models with and without the role-dependent LSTM layers. The conditions of LR and LR using the word2vec were described in [11]. The real-time dialog systems cannot use the future information and thus the parameters of Online-BLSTM and Online-Attention models were trained using backward activation vectors propagated within each utterance. All the models including speaker role based LSTM layers were trained on the platform of Chainer [17].

All LSTM-based models had one projection layer of 100 units. The LSTM model had one LSTM layer with 100 cells. The BLSTM model had forward and backward LSTM layers, each of which had 50 cells. The attention model had BLSTM layers of 50 cells in the encoder network and a LSTM layer with 50 cells in the decoder network. $U$ and $V$ in Eq. (19) were $100 \times 50$ and $100 \times 100$ matrices, respectively, and $w$ and $b$ were 100 dimensional vectors. In the case of role-dependent models, each role-dependent LSTM layer had the same number of cells as that of its role-independent model. These model sizes were basically selected using the development set. We used the cross-entropy criterion to train all the models, where we applied stochastic gradient descent (SGD) for the LSTM and BLSTM models and AdaDelta [18] for the attention models.

## 4.3. Evaluation Results

The experimental results are shown in Table 1. We confirmed the BLSTMs and attention models outperformed the other models such as LR, RNNs, LSTMs due to the forward and backward features of context of dialogs. In addition, the proposed role-dependent context-sensitive models were better than the models w/o role-dependent layers. The online models of BLSTMs and attention were comparable with the original models even using only the backward features within each utterance. The attention models were not always better than the BLSTMs and worse especially for the precise tags of DA+ST. The data size of this study might be insufficient to train attention models.

# 5. Conclusion

We proposed an efficient context-sensitive SLU approaches using role-based LSTM layers. In order to capture long-term characteristics over an entire dialog, we implemented bidirectional and attention LSTMs representing intention using consequent word sequences of each concept tag and concept tag sequence of each dialog. We evaluated the performance of importing contextual information of an entire dialog for SLU and the effectiveness of the speaker role based LSTM layers. The proposed role-dependent context-sensitive models were better than the models w/o role-dependent layers. The BLSTMs outperform LSTMs and improves the SLU baseline by 2.9% and 2.3% (absolute) for the layer of DA and DA+ST, respectively. Future work will test the proposed models trained using much large scale data.

# 6. Acknowledgements

The authors thank Prof. Alex Waibel of Karlsruhe Institute of Technology and Carnegie Mellon University to let us test our proposed method using the hotel reservation dialogs with Interchange Format.

# 7. References


[1] D. Jurafsky and J. H. Martin, *Speech & Language Processing*. Pearson Education, 2000.

[2] R. De Mori, "Spoken language understanding: a survey." in *ASRU2007*, 2007, pp. 365–376.

[3] K. Yao, G. Zweig, M.-Y. Hwang, Y. Shi, and D. Yu, "Recurrent neural networks for language understanding." in *Interspeech2013*, 2013, pp. 2524–2528.

[4] K. Yao, B. Peng, G. Zweig, D. Yu, X. Li, and F. Gao, "Recurrent conditional random field for language understanding," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2014)*. IEEE, 2014, pp. 4077–4081.

[5] K. Yao, B. Peng, Y. Zhang, D. Yu, G. Zweig, and Y. Shi, "Spoken language understanding using long short-term memory neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (SLT2014)*, 2014.

[6] C. Hori, K. Ohtake, T. Misu, H. Kashioka, and S. Nakamura, "Statistical dialog management applied to WFST-based dialog systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2009)*, 2009.

[7] Y. Shi, K. Yao, H. Chen, Y.-C. Pan, M.-Y. Hwang, and B. Peng, "Contextual spoken language understanding using recurrent neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2015)*.

[8] Y. Luan, S. Watanabe, and B. Harsham, "Efficient learning for spoken language understanding tasks with word embedding based pre-training," in *Proc. Interspeech2015*, 2015.

[9] R. Lowe, N. Pow, I. Serban, and J. Pineau, "The ubuntu dialogue corpus A large dataset for research in unstructured multi-turn dialogue systems," in *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, 2015.

[10] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[11] C. Hori, T. Hori, S. Watanabe, and J. R. Hershey, "Context sensitive spoken language understanding using role dependent lstm layers," in *Machine Learning for SLU Interaction NIPS 2015 Workshop*.

[12] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014. [Online]. Available: http://arxiv.org/abs/1409.0473

[13] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 577–585. [Online]. Available: http://papers.nips.cc/paper/5847-attention-based-models-for-speech-recognition.pdf

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, May 2013, pp. 6645–6649.

[16] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, *Human Behavior Understanding: Second International Workshop, HBU 2011, Amsterdam, The Netherlands, November 16, 2011. Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ch. Sequential Deep Learning for Human Action Recognition, pp. 29–39. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-25446-8_4

[17] P. Networks, "Chainer," in *"http://chainer.org/"*.

[18] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: http://arxiv.org/abs/1212.5701