Dialogue Intent Classification with Long Short-Term Memory Networks

Lian Meng, Minlie Huang

State Key Laboratory of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Dept. of Computer Science and Technology, Tsinghua University, Beijing 100084, PR China
mengl15@foxmail.com, aihuang@tsinghua.edu.cn

Abstract. Dialogue intent analysis plays an important role for dialogue systems. In this paper,we present a deep hierarchical LSTM model to classify the intent of a dialogue utterance. The model is able to recognize and classify user's dialogue intent in an efficient way. Moreover, we introduce a memory module to the hierarchical LSTM model, so that our model can utilize more context information to perform classification. We evaluate the two proposed models on a real-world conversational dataset from a Chinese famous e-commerce service. The experimental results show that our proposed model outperforms the baselines.

1 Introduction

Dialogue intent analysis is an important task that dialogue systems need to perform in order to understand the user's utterance in the dialogue. The intention of a speaker delivered in dialogue is called a dialogue act(DA) [1]. In real-world applications, understanding user's utterances is crucial for downstream processes such as dialogue management, knowledge base search, and language generation.

In open-domain conversations, context information (one or a few previous utterances) is particularly important to language understanding [2]. The real spoken dialogue scenario always have multiple turns, and the number of back and forth between both sides increases as the complexity of the scenarios grows. The accurate understanding of next dialogue sentence often requires reasoning from its previous conversational history, to which we refer as context. Failing to consider the contextual information may result in incorrect interpretation of the user's intent.

In order to perform dialogue intent analysis,various classification models have been proposed to deal with natural language understanding tasks. However, models that use original lexical features without any modifications always encounter the problem of data sparseness, and constructing sufficient training data to overcome this problem is labor-intensive, time-consuming, and expensive.

In recent studies, the method of deep learning is widely used in Natural Language Processing(NLP) tasks. Inspired by the performance of recent studies utilizing deep learning in NLP, various RNN structures have been proposed. RNN is now the most popular method in text or sentence classification [3] which is also a typical NLP task. While neural network based techniques have been extensively applied to most of the dialogue problems in recent years,they have not been fully explored for contextual understanding

In this paper, we propose neural networks for classifying the intent of online service conversations. We present a hierarchical long short-term memory(HLSTM) network for dialogue intent classification, where a word-level LSTM is used to model a utterance and a sentence-level LSTM to model the contextual dependence between sentences. Further, we propose a memory module in this network to enhance the capability of context modeling. Results show these attempts improve the basic LSTM model.

2 Related Work

2.1 Dialogue act classification

Previous work in dialogue act classification mainly focused on domain-specific classification for goal-oriented dialogue systems [4] and researchers in linguistics, computational linguistics, and natural language processing had conducted these previous research. Those work has showed that the dialogue act recognition performance was dependent on the classification systems and the methods used.

Previous work on dialogue act recognition has mainly focused on supervised learning method. Almost all standard approaches to classification have been applied in DA classification, from Support Vector Machines (SVM) and Hidden Markov Models (HMM) [5] to Decision Trees (DT) [6], Bayesian Networks (BN) [7] and rule-based approaches [8].

The above studies do not consider context information from the whole session level. The main disadvantage of previous methods is their heavy dependency on the size of the training dataset for recognizing multiple contexts within the same user utterance and correctly identifying the user's intention in ambiguous expressions.Recently, approaches based on deep learning methods were used to build contextual information model in dialogue. Since a dialogue session is naturally a sequence-to-sequence process at the utterance level, recurrent neural network (RNN) is proposed to model the process [9] and deep RNN was used to classify dialogue acts [10].

2.2 Memory Network

The Memory network architecture, introduced by [11], consists of two main components:supporting memories and final answer prediction. It is trained end-to-end, and hence requires significantly less supervision during training, making it more generally applicable in realistic settings. Supporting memories are in turn comprised of a set of input and output memory representations with memory cells.

Memory networks can extend the state representation of RNN with an external memory, which can represent more information and offer a more flexible way for context modeling [12]. [13] showed a neural network with an explicit memory and a recurrent attention mechanism, in their language modeling tasks, it slightly outperforms tuned RNNs and LSTMs of comparable complexity. [14] introduced a dynamic memory network (DMN) to do NLP applications including sequence modeling, classification and question answering.

Classical neural network memory models such as associative memory networks aim to provide content-addressable memory,given a key vector to output a value vector and references therein.

3 Proposed Model

3.1 Overview

Recurrent Neural Networks(RNN) [15] are increasingly used to do classify task. For sequence modeling task such as intent classification, capturing long distance information is a key issue. Figure 1 illustrates a typical structure of an RNN, where x_t is the input at time step t and h_t is the hidden state. As can be seen, information from previous layers h_{t-1}, is contributed to the succeeding layer's computations that generate h_t.

$$h_t = f(W_x x_t + W_h h_{t-1} + b_n) \tag{1}$$

Theoretically, RNN is able to capture dependence of arbitrary length, it tends to suffer from the gradient vanishing and exploding problems which limit the length of reachable context. In addition, an additive function of the previous hidden layer and the current input is too simple to describe the complex interactions within a sequence.

We care about remembering some information that is crucial for the final result and it is important to have some information omitted during the operation of the network, as not everything affects positively the network performance. Considering the aforementioned problems with RNNs, we use Long Short Term Memory (LSTM), which is a variation of RNNs that is tuned to preserve long-distance dependencies as their default specificity. It adopted a gating mechanism. Another reason for using LSTM is that it uses a forget gate layer to distill trivial weights, which belong to unimportant words from the cell state. There are many variants of LSTM unit, here we adopt one widely used architecture where inputs are d dimensional vectors, i_t is the input gate, f_t is the forget gate, o_t is the output gate, c_t is the memory cell, h_t is the hidden state, t denotes time step and $\odot$ represents element-wise multiplication.

$$i_t = \sigma(W^{(i)} X_t + U^{(i)} h_{t-1} + b^{(i)}) \tag{2}$$

$$f_t = \sigma(W^{(f)} X_t + U^{(f)} h_{t-1} + b^{(f)}) \tag{3}$$

$$o_t = \sigma(W^{(o)} X_t + U^{(o)} h_{t-1} + b^{(o)}) \tag{4}$$

$$u_t = \tanh(W^{(u)} X_t + U^{(u)} h_{t-1} + b^{(u)}) \tag{5}$$

$$c_t = i_t \odot u_t + f_t \odot c_{t-1} \tag{6}$$

$$h_t = o_t \odot \tanh(c_t) \tag{7}$$

In LSTMs, the gates in each cell that decide dynamically which signals are allowed to pass through the whole chain. LSTMs are able to view information over multiple time scales due to the fact that gating variables are assigned different values for each vector element. Deep LSTM structure had been used to classifying dialogue acts [10]

3.2 Hierarchical LSTM

The basic LSTM model is used to encode the information from the input word sequence into a fixed-length vector representation. The dialogue is a hierarchical sequence of data: each sentence is a sequence of words, and each session is a list of sentences. To model the whole context, compared to the basic LSTM model, we introduce the power of context into a standard LSTM model and propose the Hierarchical LSTM(HLSTM) model. The initial of this model is to represent the diolague session more completely. Given a dialogue (n sentences) $d = [s_1, s_2, ..., s_n]$. We first use a LSTM (LSTM$_1$) to model all the sentences in each session independently. The hidden states of sentence s_i obtained at this step are used to generate a sentence vector v_i using another LSTM (LSTM$_2$) for each sentence s_i in the dialogue. These sentence vectors can be used as features for dialogue act analysis in next step.

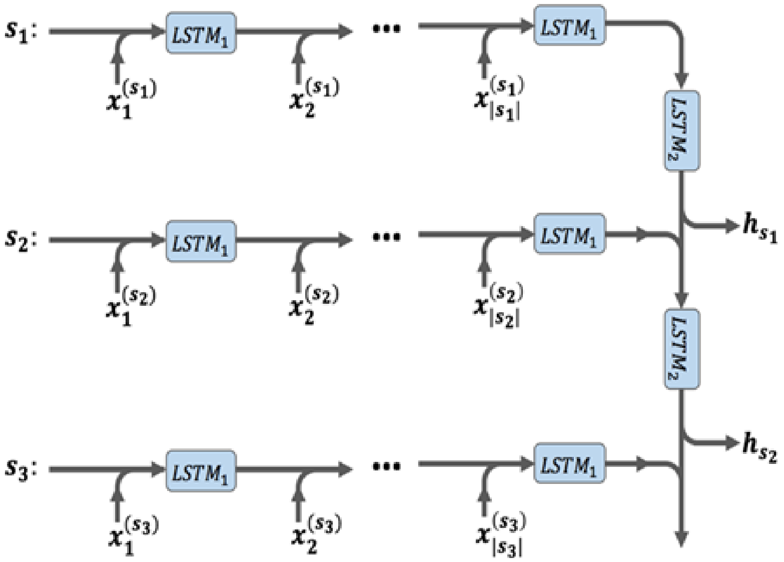

Fig. 1. The Hierarchical LSTM Model

The hierarchical LSTM model connect the relationship between sentences and context information more closely, so it can combinate the dialogue session context to make the sentence intent classification more effectively. The output sentence vector of LSTM$_2$ is two dimension Matrix, we do dropout operation at a certain ratio. After dropout, we reshape the Matrix based sentence number rows and there is a softmax layer over output vectors.

$$P_\theta(y_j|h_{s_i}) = softmax(h_{s_i}w + b) \tag{8}$$

$$Y_{pred} = argmaxP_\theta \tag{9}$$

The final prediction is the label with the highest probability P_θ.

3.3 Memory Augmented Hierarchial LSTM

To further enhance the modeling of complex dialogues context information, we add a memory component to the HLSTM model. This component is placed on the output of LSTM$_2$, which will memorize and provide useful context information when calculate the sentence vector. The saved vectors in memory will be updated after each read.

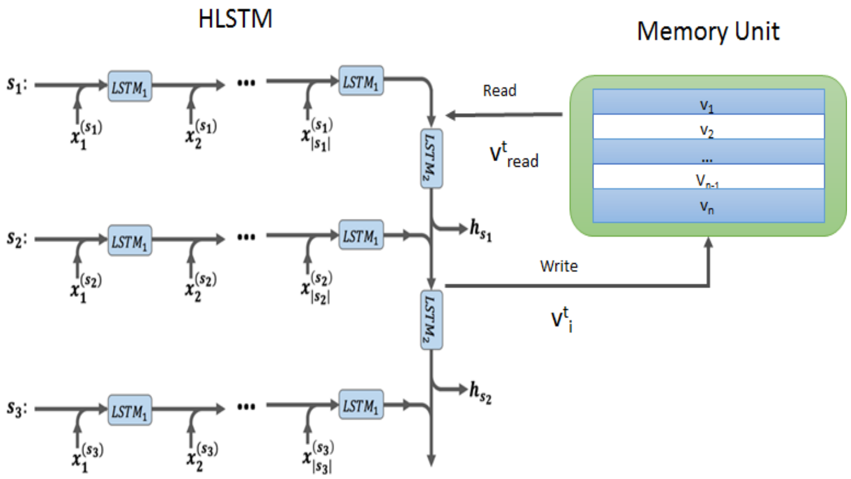

Fig. 2. The memory-augmented HLSTM.

We use $v = [v_1, v_2...v_n]$ to represent the vectors we set in the memory, the read and write procedure of the memory can be formulated:

$$a_i = softmax(h_{s_{t-1}} v_i^{t-1}) \tag{10}$$

where the softmax is normalized over all memory units.

$$v_{read}^t = \sum_{i=1}^{N} a_i v_i^{t-1} \tag{11}$$

We use $h_{s_{t-1}}$ to represent the prior hidden vector given by LSTM$_2$, s_t is sentence vector given by LSTM$_1$, the c_t and h_{s_t} can be updated by:

$$c_t, h_{s_t} = LSTM(c_{t-1}, h_{s_{t-1}}, s_t, v_{read}^t) \tag{12}$$

The write process of memory is formulated as:

$$v_i^t = tanh(W_v v_i^{t-1} + W_{s_t} s_t + b) \tag{13}$$

The weight of the memory units will be updated after the calculation of LSTM_2's hidden vector.

3.4 Model Training

We formulate the DA classification as a mutiple classification task. The training criterion is a cross-entropy loss [16] for a session example, which is annotated by true lables predefined.To train our network, we use mini-batch stochastic gradient descent(SGD) with adaptive learning rate computed by Adadelta, which shows better performance and convergence property. We update model parameters after every mini batch, check validation accuracy and save model after every 10 batches. After each optimization epoch, we monitor the performance of the model. When the performance stops increasing for several iterations, we terminate the training and select the best-performing model.

4 Experiments

The domain of the dialogue dataset we use focus on the scenario of buy cellphone online. According to our daily experience of shopping online, we usually ask some questions about the cellphone we about to buy. For example, the property of cellphone is our most concern aspect. We need to know clearly about its price, performance, express delivery and other aspects we care about. To accurately identify the intent of every sentence, the related team in this e-commerce company had done some significant and effective work. They build a shopping online ontology system based on several different interactive scenario, including purchase, commodity, after sale and so on. Under each ontology, they also classified more detailed intent as the second level. According to the actual transaction scenario, every ontology has two or three level subordinate intent. The dialogue intent label mainly focus on the third level.

4.1 Data and Setup

We perform experiments on the real dialogue dataset provided by one e-commerce company. The dataset includes about 1504 real online dialogue annotated sessions from the cellphone domain, contains 24760 sentences and 108 labels. The average length of each session is 16. The dataset is randomly split into training set(80%), validation set(10%) and test set(10%).

Table 1. The size of used dataset

Sessions	Sentences	Labels	Avg number of sentences per session
1504	24760	108	16.4

We used this dataset to tune all hyperparameters of model. The sessions in the training set were preprocessed, so the LSTM parameters can be trained though a reasonable number of epochs. Each time we tuned one parameter value and measured the accuracy on the test set, if the accuracy on the development set did not change for 10 epochs, we stop training.

Our implementation of HLSTM is based on open source library Theano. We use word2vec vectors [17] which were trained on 400 billion words from Weibo corpus as word embedding. The vectors' dimensionality is 100 and those words not in the vectors are set randomly. We update model parameters after every mini-batch, we run 50 epochs in total, and the model with highest test accuracy is treated as the optimal model.

4.2 Results

We evaluated the performance of DA classification on the basic LSTM single sentence modle, HLSTM model and HLSTM+Mem model. Results are shown in Tabel2. The average accuracy of the baseline LSTM model on this dataset is 74.5% , while the average accuracy of the HLSTM model is 76.3%. The HLSTM model has an improvement of 1.8% over the basic single sentence model.

Table 2. Accuracy of the three models.

Model	**Accuracy(%)**
Basic LSTM	74.5
HLSTM	76.3
HLSTM+Mem	76.7

Because the dialogue act of each sentence is labeled by predefined rules, which conducted by the specific programs. In the real data, there is a certain proportion of sentences can not match the existing labelling ontology system, these sentences were labeled by 'N/A'. To eliminate the impact of these sentences on contextual information of each session. We sort the labels based on their statistics and chose the top 20 label to do the classification task. Results are shown in Tabel3. As we can see, the overall performance of 20 classification task was higher than before.

Table 3. 20 classification accuracy .

Model(20 label)	**Accuracy(%)**
Basic LSTM	79.7
HLSTM	81.6
HLSTM+Mem	83.9

Compare the above two experimental results, we can find out the HLSTM model achieved better results in comparison to basic LSTM. After add the memory unit, performance had been further improved, the number of intent label which used to classify and the proportion of sentences under the same label are important factors which determine the model's performance.

4.3 Error Analysis

Based on the experimental results and analysis of the existing data, we summarized some characteristics and difficulties in the data.

The data contains different kinds of emojis, URL addresses, photograph links and other non-literal symbols. All these symbols have its own unique meaning, they also represent a dialogue intent of user utterance. So identifying and translating these symbols is very important for us to get contextual information.

Since the data based on true conversation, the task of labelling spoken, conversational data is clearly complex. Some categories in the ontology system are difficult for humans and machines to separate. The existing labelling mechanism is not enough to deal with all possible situations.

5 Conclusion

In this study, we proposed deep hierarchical LSTM models for classifying dialogue intents in an e-commerce domain. The two models include an HLSTM and an memory-augmented HLSTM. Experiment results show that our proposed models efficiently utilize dialogue context information for intent classification. The adoption of the memory component can further improve the model's performance.

In the future, we would like to further improve our model and apply to other classification problems in the dialogue system.

Acknowledgments

References

1. Austin, J. L. Gu, and Yueguo. *How to do things with words*. Clarendon Press,, 2012.
2. Chunxi Liu, Puyang Xu, and Ruhi Sarikaya. Deep contextual language understanding in spoken dialogue systems. 2015.
3. Lei Shen and Junlin Zhang. Empirical evaluation of rnn architectures on sentence classification task. 2016.
4. T. Bub and J. Schwinn. Verbmobil: the evolution of a complex large speech-to-speech translation system. 4:2371–2374 vol.4, Oct 1996.
5. Dinoj Surendran and Gina Anne Levow. Dialog act tagging with support vector machines and hidden markov models. In *In Proceedings of Interspeech/ICSLP*, pages 1–28, 2006.
6. S. A. Ali, N. Sulaiman, A. Mustapha, and N. Mustapha. Improving accuracy of intention-based response classification using decision tree. *Information Technology Journal*, 8(6), 2009.
7. Simon Keizer. Dialogue act modelling using bayesian networks. 2001.
8. Yasuhisa Niimi, Tomoki Oku, Takuya Nishimoto, and Masahiro Araki. A rule based approach to extraction of topics and dialog acts in a spoken dialog system. In *Eurospeech 2001 Scandinavia, European Conference on Speech Communication and Technology, INTERSPEECH Event, Aalborg, Denmark, September*, pages 2185–2188, 2001.

9. Matthew Henderson, Blaise Thomson, and Steve Young. Word-based dialog state tracking with recurrent neural networks. In *Meeting of the Special Interest Group on Discourse and Dialogue*, pages 292–299, 2014.
10. Hamed Khanpour, Nishitha Guntakandla, and Rodney Nielsen. Dialogue act classification in domain-independent conversations using a deep recurrent neural network. In *COLING*, 2016.
11. Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *Eprint Arxiv*, 2014.
12. Baolin Peng, Kaisheng Yao, Li Jing, and Kam Fai Wong. Recurrent neural networks with external memory for spoken language understanding. In *Ccf Conference on Natural Language Processing and Chinese Computing*, pages 25–35, 2015.
13. Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. *Computer Science*, 2015.
14. Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. Ask me anything: dynamic memory networks for natural language processing. *Computer Science*, pages 1378–1387, 2015.
15. Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *INTERSPEECH 2010, Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September*, pages 1045–1048, 2010.
16. Lih Yuan Deng. The cross-entropy method: A unified approach to combinatorial optimization, monte-carlo simulation, and machine learning. *Technometrics*, 48(1):147–148, 2006.
17. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *Computer Science*, 2013.