A Unified Neural Architecture for Joint Dialog Act Segmentation and Recognition in Spoken Dialog System

Tianyu Zhao
Graduate School of Informatics,
Kyoto University
zhao@sap.ist.i.kyoto-u.ac.jp

Tatsuya Kawahara
Graduate School of Informatics,
Kyoto University
kawahara@i.kyoto-u.ac.jp

Abstract

In spoken dialog systems (SDSs), dialog act (DA) segmentation and recognition provide essential information for response generation. A majority of previous works assumed ground-truth segmentation of DA units, which is not available from automatic speech recognition (ASR) in SDS. We propose a unified architecture based on neural networks, which consists of a sequence tagger for segmentation and a classifier for recognition. The DA recognition model is based on hierarchical neural networks to incorporate the context of preceding sentences. We investigate sharing some layers of the two components so that they can be trained jointly and learn generalized features from both tasks. An evaluation on the Switchboard Dialog Act (SwDA) corpus shows that the jointly-trained models outperform independently-trained models, single-step models, and other reported results in DA segmentation, recognition, and joint tasks.

1 Introduction

A growing interest in interactive conversational agents and robots has motivated research focus on spoken language understanding (SLU). As an essential part of spoken dialog system (SDS), SLU analyzes user input, and provides the dialog system with information to make a response. In conversations, dialog act (DA) represents the communicative function of an utterance (Stolcke et al., 2000). For instance, we can use DA tag *Statement* to describe utterance *"Me, I'm in the legal department."* and use *Yes-No-Question* to describe *"Do you have to have any special training?"*. Recognition of DA benefits the understanding of dialog structure, thus allows SDS to conduct meaningful and smooth conversation, e.g. a *Yes-Answer* or *No-Answer* to a *Yes-No-Question*, and end the conversation after a *Conventional-closing*.[1]

Most of previous works focused on DA recognition given transcriptions that are manually segmented (Stolcke et al., 2000; Ivanovic, 2005; Webb et al., 2005; Sridhar et al., 2009; Li and Wu, 2016; Khanpour et al., 2016; Lee and Dernoncourt, 2016; Shen and Lee, 2016; Joty and Hoque, 2016). Early works applied decision trees, Hidden Markov Model (Stolcke et al., 2000), and n-gram models (Stolcke et al., 2000; Ivanovic, 2005) to classify DA tags. Recently, hierarchical neural networks have been introduced to the task. Such models encode a DA segment into a *sentence encoding* by one network and apply the other network for DA recognition given a sequence of *sentence encoding*. Different combinations of networks such as CNN-ANN, RNN-ANN (Lee and Dernoncourt, 2016), and RNN-RNN (Li and Wu, 2016; Khanpour et al., 2016) are shown to greatly improve the accuracy of DA recognition. Ji et al. (2016) introduced an extra latent variable to a hierarchical RNN model to represent discourse relation. Jointly training the latent variable model on DA recognition and language modeling tasks yields competitive results. Recent works (Kumar et al., 2017; Chen et al., 2017) on DA recognition use a hierarchical encoder to generate a vector representation for each DA segment, then a Conditional Random Field (CRF) tagger is applied to sequence labeling given the sequence of segment representations. Kumar et al. (2017) reported an accuracy of 79.2% on SwDA, while Chen et al. (2017) achieved the current state-of-the-art accuracy of 81.3% by incorporating attentional mechanism and extra inputs (character embeddings, Part-

[1] Examples of DA tags and utterances are selected from the Switchboard Dialog Act (SwDA) corpus.

Proceedings of the SIGDIAL 2018 Conference, pages 201–208,
Melbourne, Australia, 12-14 July 2018.

Words	okay	so	I	guess	it	starts	recording	now
Segmentation	E	I	I	I	I	I	I	E
DA	Backchannel	Statement						

Table 1: DA segmentation and recognition: "I" tag refers to inside of a segment, and "E" is the end of a segment.

of-Speech tags, and named entitiy tags). However, these models with CRF layer assume that complete dialog is given before prediction. Thus the reported performances will not apply to real-time SDS, where DA tags are often predicted incrementally.

As shown in Table 1, an utterance in a conversational turn can consist of several DA units. In the example, we use *"E"* tag to denote the end of a segment and *"I"* for inside. The utterance *"okay so I guess it starts recording now"* are split into two segments, which are a *Backchannel* and a *Statement* respectively. However, automatic speech recognition (ASR) in SDS usually provides no punctuation that gives hints for DA segmentation, thus it is necessary to build a sequence labeler for automatic DA segmentation.

A majority of previous works of DA segmentation formulated DA segmentation and recognition in a single step (Zimmermann et al., 2006; Zimmermann, 2009; Quarteroni et al., 2011; Granell et al., 2009). Segmentation labels are combined with DA labels (e.g. *"E_Statement"* denotes the end of a *Statement* segment), and a sequence labeling model is applied to predict tags for both tasks. This approach has a merit of integration so that recognition helps segmentation and segmentation errors are not propagated to the recognition step. On the other hand, it has a drawback that it can hardly incorporate a history of preceding sentences to predict the DA tag of the current sentence. Another approach is to process the data in a pipeline manner. Manuvinakurike et al. (2016) used a CRF for DA segmentation and a Supported Vector Machine (SVM) for DA recognition given predicted segments. For pipeline methods, downstream task (e.g. DA recognition) is vulnerable to errors from upstream task (e.g. DA segmentation). In this paper we propose a unified architecture based on neural networks for DA segmentation and recognition to solve the aforementioned problems. Our method uses separate models for DA segmentation and recognition but introduces joint learning so that the models can learn from both tasks.

Joint learning (also multi-task learning) allows a model to learn from different tasks in parallel, which benefits the generalization of the model. Collobert and Weston (2008) introduced a unified architecture based on Convolutional Neural Networks (CNNs) to natural language processing tasks such as Part-of-Speech (POS) tagging and chunking, and showed that joint learning of related tasks improves model performance. Inspired by this work, we investigate joint learning of DA segmentation and recognition for better generalized model. We compare the jointly-trained models under the unified architecture with models trained separately and previous works on the Switchboard Dialog Act (SwDA) corpus.

2 Models and Training

The proposed method applies a word sequence tagger for segmentation and a sentence classifier for recognition. Under a unified neural architecture, the sequence tagger and the classifier share parameters to learn features from each other and improve generalization. As shown in Figure 1, the left part corresponds to a word sequence tagger for segmentation using Bidirectional Long Short Term Memory (BiLSTM) (Schuster and Paliwal, 1997) and on the right-hand side is a sentence classifier for DA recognition based on hierarchical LSTMs (Hochreiter and Schmidhuber, 1997).

Components for segmentation and recognition will be explained in Section 2.1 and 2.2. In Section 2.3, three proposed models are introduced. In order to compare the proposed models with conventional approach, we describe a single-step model that uses combined labels in Section 2.4.

2.1 Word Sequence Tagger for DA Segmentation

Regarding DA segmentation as a sequence labeling problem, BiLSTM naturally fits the task since it can exploit information of surrounding words in the prediction of the current word. The sequence tagger predicts a segmentation label y_t

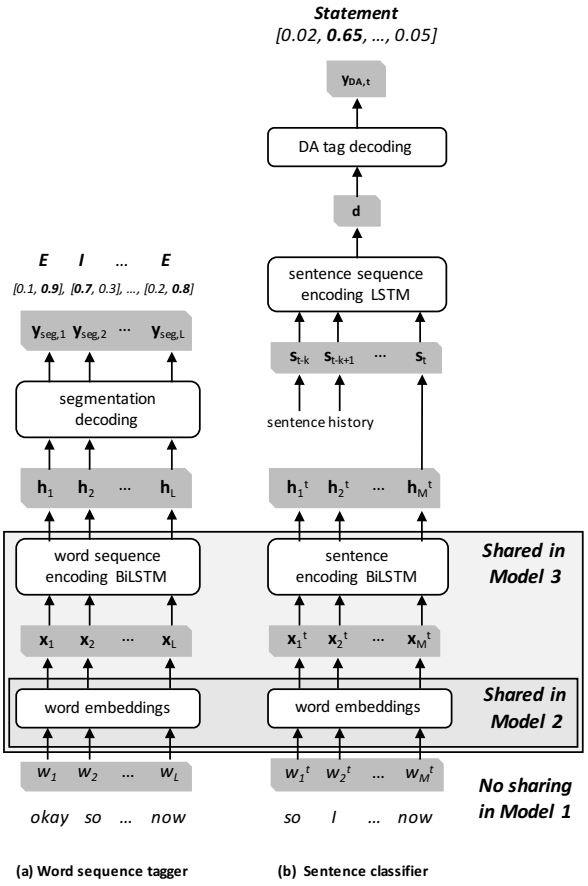

Figure 1: A unified neural architecture consisting of a word sequence tagger for DA segmentation and a sentence classifier for DA recognition.

for each word w_t in the input utterance $w_{1:L}$. A word embedding layer firstly maps the input words $w_{1:L}$ into distributed vector representation of words $\mathbf{x}_{1:L}$. Then we use a BiLSTM to process the sequence and output hidden states $\mathbf{h}_{1:L}$. The last decoding layer computes a probability distribution $\mathbf{y}_{seg,1:L}$ over segmentation labels:

$$\mathbf{x}_{1:L} = \text{word-embedding}(w_{1:L}), \quad (1)$$
$$\mathbf{h}_{1:L} = \text{BiLSTM}(\mathbf{x}_{1:L}), \quad (2)$$
$$\mathbf{y}_{seg,t} = \text{softmax}(W_{seg}\mathbf{h}_t + \mathbf{b}_{seg}), \quad (3)$$

where W_{seg} and $\mathbf{b}_{seg}$ are trainable parameters in the decoding layer.

2.2 Sentence Classifier for DA Recognition

Accurate recognition of DA requires understanding of discourse relations (Ji et al., 2016). Therefore, preceding sentences are needed as context in the recognition of the current sentence. Hierarchical neural networks are able to encode intra-sentence information and capture inter-sentence relations through a two-level hierarchy. The lower level of the network generates a *sentence encoding* $\mathbf{s}_t$ for input sentence $w^t_{1:M}$ via BiLSTM, and the higher-level LSTM network predicts a DA tag of the input sentence given *sentence encoding* $\mathbf{s}_t$ as well as *sentence encoding*s of preceding k sentences $\mathbf{s}_{t-k}, \mathbf{s}_{t-k+1}, \cdots, \mathbf{s}_{t-1}$.

We use a word embedding layer and a BiLSTM layer to obtain hidden states $\mathbf{h}^t_{1:M}$ as same as in the sequence tagger. The last hidden state $\mathbf{h}^t_M$ (sum of the last hidden states on two directions of BiLSTM as shown in Equation 4) is used as *sentence encoding* $\mathbf{s}_t$. In the same way, $\mathbf{s}_{t-k:t-1}$ are calculated and used as a context in the sentence sequence encoding network. We use a vanila LSTM to process sequence $\mathbf{s}_{t-k:t}$, and input the last hidden state $\mathbf{d}$ to a DA tag decoding layer to compute the probability distribution over DA tags.

$$\mathbf{h}^t_M = \overrightarrow{\mathbf{h}}^t_M + \overleftarrow{\mathbf{h}}^t_M, \quad (4)$$
$$\mathbf{s}_t = \mathbf{h}^t_M, \quad (5)$$
$$\mathbf{d} = \text{LSTM}(\mathbf{s}_{t-k:t}), \quad (6)$$
$$\mathbf{y}_{DA,t} = \text{softmax}(W_{DA}\mathbf{d} + \mathbf{b}_{DA}), \quad (7)$$

where W_{DA} and $\mathbf{b}_{DA}$ are trainable parameters in the DA tag decoding layer.

2.3 Proposed Models

Based on the aforementioned word sequence tagger and sentence classifier, we introduce three models. Different from the single-step method in past works, the proposed models work in a cascading manner, i.e. to split the input text $w_{1:L}$ into segments using the word sequence tagger, then feed each segment $w^t_{1:M}$ to the sentence classifier to predict its DA tag. As shown in Figure 1, the segmentation component and the DA recognition component have the same structure in their lower-level parts (a word embedding layer and a BiLSTM-based encoder layer). The difference between the three models is the number of shared layers.

- **Model 1** A straightforward method is to separately build a word sequence tagger and a sentence classifier. The model that has no shared layers is called Model 1.

- **Model 2** Word embedding layers are shared between the sequence tagger and the DA classifier in Model 2. When training the sequence tagger on the segmentation task, gradients

from top end are back-propagated into the shared word embeddings that are also used by the DA classifier, vice versa. Parameters in the shared word embedding layer are updated by losses from both tasks, thus the model learns generalized features on the word level.

- **Model 3** We combine both the word embedding layers and the BiLSTM encoding layers which produce $\mathbf{h}_{1:L}$ and $\mathbf{h}^t_{1:M}$ in Model 3. Since the higher-level layers are shared, this model is expected to learn generalization in high-level features.

2.4 Single-step Model for DA Segmentation and Recognition

Previous single-step approaches to DA segmentation and recognition are based on non-network models such as Conditional Random Field (CRF). For a fair comparison between the proposed neural models and single-step method, we implement an LSTM-based sequence tagger to predict combined labels in a single-step manner. The single-step model resembles the segmentation component in Section 2.1 and the only difference is that a set of combined labels are used in the output layer as shown in Figure 2. Therefore, instead of predicting segment boundaries (label *"E"*) only, it generates DA labels at the end of each segment as well (e.g. *"E_Backchannel"*, *"E_Statement"*, etc.).

2.5 Training

The sequence tagger receives a whole turn (i.e. a sequence of consecutive words uttered by one speaker) and predicts segmentation tags (combined tags in the case of single-step model) for all words in the turn. Cross-entropy loss is computed for each word and back-propagated. As for the DA classifier, we use ground-truth segments that are manually transcribed as inputs to the classifier. The model is trained to minimize cross-entropy loss between the predicted DA tag and the oracle DA tag.

When training the joint models (Model 2 and 3), we can apply different strategies to optimize the segmentation and recognition components. One alternative, for example, is to train the segmentation component for one epoch and the recognition component for the next epoch. However, it results in that segmentation loss is likely to dominate the optimizing direction for an entire epoch and vice versa for another epoch. This may prevent the model from learning from different signals simultaneously. Thus, instead of switching between segmentation loss and recognition loss every epoch, we compute and minimize both segmentation loss and recognition loss for every mini-batch of data.

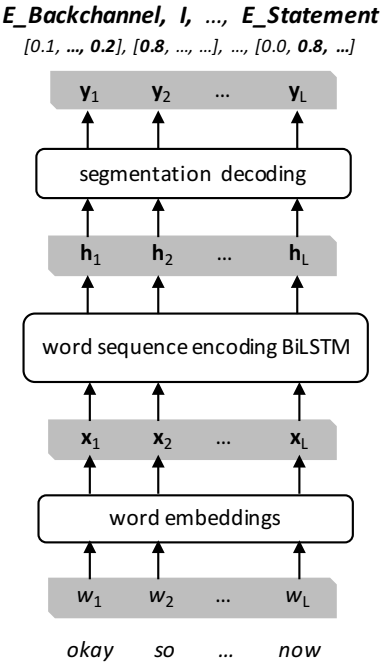

Figure 2: An LSTM-based single-step model that uses combined labels

3 Experimental Evaluations

Three sets of experiments are conducted to evaluate model performance on the DA segmentation task, the DA recognition task, and their joint task respectively. In the segmentation task, we use the word sequence tagger to predict segmentation labels given a sequence of words in a turn. In the recognition task, segments with correct boundaries are given as inputs, and we use the sentence classifier to predict a DA tag for each segment. Lastly in the joint task, instead of using segments with correct boundaries, we split each turn into segments according to the predicted segmentation labels by the sequence tagger. Then the sentence classifier outputs DA tags for the predicted segments.

3.1 Evaluation Metrics

Word-level error rate is used to assess performance on the segmentation task. It compares the predicted boundaries with ground-truth boundaries and counts the number of words that lie in wrongly

Reference	I	I	E_G	I	I	E_S	I	I	E_Q	I	E_S
Prediction	I	I	E_G	I	I	I	E_S	I	E_Q	I	E_R
Word-level Segmentation Error	√	√	√	×	×	×	×	×	×	√	√
Word-level Joint Error	√	√	√	×	×	×	×	×	×	×	×

Table 2: An example of the calculation of metrics for segmentation and joint tasks, where word-level segmentation error rate is 54.5% (6/11), and word-level joint error rate is 72.7% (8/11).

predicted segments. The joint task is evaluated on word level as well. However, it additionally takes DA tags into consideration. An example of the calculation of these metrics is illustrated by Table 2. The DA recognition task is evaluated by accuracy.

3.2 Corpus and Preprocessing

The Switchboard Dialog Act (SwDA) corpus (Jurafsky et al., 1997) is used for evaluation. It contains 1155 human-human telephone conversations and is annotated with 42 clustered DA tags. We follow the train/dev/test set split by Stolcke et al. (2000). Table 3 gives related statistics of the corpus.

dataset	train	dev	test
# of sessions	1003	112	19
# of turns	91k	10k	2k
# of segments	178k	19k	4k
# of words	1565k	164k	35k

Table 3: Corpus statistics of SwDA.

The SwDA corpus is manually transcribed, segmented and labeled with DA tags. In order to conduct meaningful experiments, we removed all the punctuation (i.e. commas, periods, exclamation marks, and question marks) and slash marks ("/") in the transcription because they cover most of segmentation boundaries and we cannot obtain such punctuation labels from ASR results in practice. Moreover, all the annotation comments in brackets are removed. Capitalization of words are also converted into the lower case. Vocabulary is limited to the most frequent 10,000 words (originally 21,177 words after preprocessing) for fast training and inference.

3.3 Experimental Setup

We use the mini-batch gradient descent with momentum to optimize the models with a mini-batch size of 32 for 20 epochs. The learning rate is set as 1 initially and decays in half when the total loss of development dataset does not decrease. Gradients are clipped between [-0.5, 0.5] to avoid exploding. We also experiment with different values of history length k from 1 to 5, which is the number of preceding *sentence encoding*s used in the upper-level LSTM of the DA recognition. For all the implemented models, we choose 200, 100 as the dimension of word embedding and the dimension of LSTM hidden states respectively. Both word sequence encoding BiLSTM and sentence encoding BiLSTM consist of two hidden layers, while the sentence sequence encoding LSTM has only one hidden layer. Dropout (Srivastava et al., 2014) is applied after the word embedding layer and between the BiLSTM layers with a drop probability of 0.5.

3.4 Experimental Results

3.4.1 Segmentation Evaluation

The error rates of the three models are shown in Figure 3. With punctuation and slash marks removed, segmentation error rates are fairly high (from 18.7% to 20.8%). However, the jointly-trained models (Model 2 and 3) always result in lower error rates than Model 1. It indicates that joint training benefits the segmentation model in the unified architecture. Specifically, there is a statistically significant error rate reduction of 1.3% when comparing the best result of Model 2 (18.7%) with that of Model 1 (20.0%), and also a statistically significant reduction of 0.9% when compared with the single-step model's 19.6%.

Quarteroni et al. (2011) reported a segmentation error rate of 1.4% using CRF model in their work. However, they used punctuation and slash marks which we removed, thus it is inappropriate to directly compare the results.

3.4.2 Recognition Evaluation

As shown in Figure 4, Model 1 achieves 77.1% at a history length of 5 and gives a strong baseline. Through joint training, Model 2 and 3 further improved the accuracy to 77.7% and 77.8% at history length of 1 and 2. Since the single model simulta-

Model	Segmentation Error Rate	Recognition Accuracy	Joint Error Rate
Model 1	20.0	77.1	31.8
Model 2	**18.7**	77.7	**30.6**
Model 3	18.9	**77.8**	31.0
single-step model	19.6		33.5
CRF (Quarteroni et al., 2011)	1.4*		29.1*
CNN-ANN (Lee and Dernoncourt, 2016)		73.1	
DRLM (Ji et al., 2016)		77.0	
Hierarchical GRU (Li and Wu, 2016)		79.4**	

* The CRF used punctuation and slash marks for segmentation. For reference, when punctuation and slash marks are reserved in our experiments, Model 2 gets a word-level segmentation error rate of 0.3% and a joint error rate of 20.5%.

** Non-textual features were used in this work.

Table 4: Best results (in %) of models.

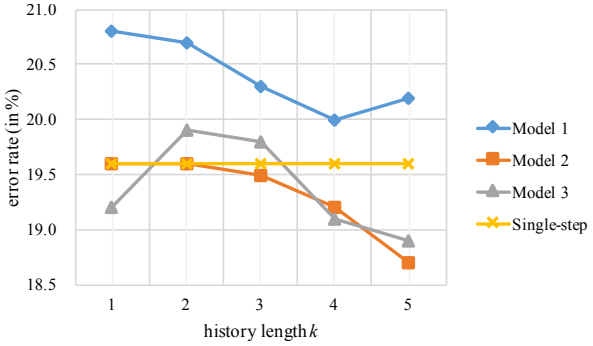

Figure 3: Word-level segmentation error rates

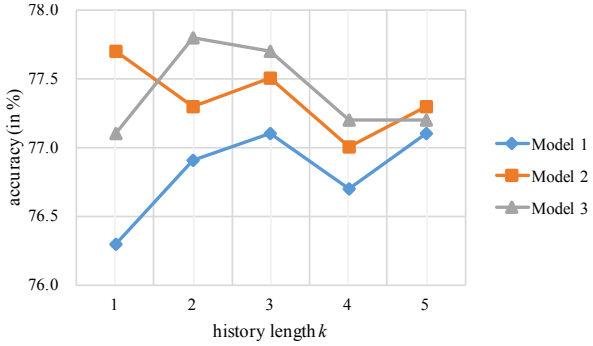

Figure 4: Recognition accuracies

neously predicts segmentation and DA labels, it is unable to predict a DA tag given a sentence with ground-truth boundaries and is excluded from the recognition evaluation.

Lee and Dernoncourt (2016) reported a recognition accuracy of 71.4% using a CNN-ANN model and Ji et al. (2016) reported 77.0% using a jointly-trained latent variable RNN. Li and Wu (2016) reached 79.4% by using extra non-textual features including sentence length, utterance index, sub-utterance index, and turn-taking information.

3.4.3 Joint Evaluation

Figure 5 shows word-level joint error rates of the proposed models. Model 1, 2, and 3 have lowest error rates of 31.8%, 30.6%, and 31.0% respectively. We can see that Model 2 and 3 have better results than Model 1 for all history lengths, which suggests jointly-trained models consistently perform better. It is confirmed from the results that joint learning gives a statistically significant error rate reduction (1.2% reduction from 31.8% of Model 1 to 30.6% of Model 2). The single-step neural network results in 33.5% joint error rate, much higher than the proposed models. A major reason is that the single-step model cannot capture context of preceding sentences, thus degrading recognition accuracy and leading to poor performance in the joint task.

A single-step CRF model by Quarteroni et al. (2011), which uses word and Part-of-Speech (POS) n-grams features, reached a word-level joint error rate of 29.1% while its segmentation error rate reached 1.4% using punctuation and slash marks in transcription. If we also reserve punctuation and slash marks in our experiments, Model 2 is able to get a lowest joint error rate of 20.5% with a segmentation error rate of only 0.3%.

Model 3 shares the higher-level layers than Model 2 but does not develop consistent and significant advantage. We noticed that the segmentation performance and recognition performance of

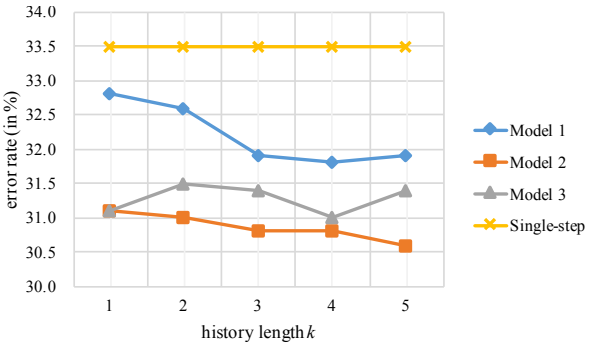

Figure 5: Word-level joint error rates

Model 3 have a reverse trend, i.e. the recognition accuracy decreases when the segmentation error rate reduces. We suspect that since most parameters in the segmentation components are shared (all layers except for the segmentation decoding layer) in Model 3, signals from the DA recognition side can affect the entire segmentation model and lead to problems in optimization.

The best results of the mentioned models in segmentation, recognition, and joint tasks are summarized in Table 4.

4 Conclusion

In this work, we presented a unified neural architecture for joint DA segmentation and recognition for SDS, which consists of a word sequence tagger and a sentence classifier. Since the two components have similar structure, we partially merged them in their word embedding layers (Model 2) and BiLSTM encoding layers (Model 3). Experimental results of segmentation, recognition and the joint tasks on the Switchboard Dialog Act (SwDA) corpus showed that the proposed models gained significant error rate reduction over single-step approaches. Among the three models, Model 2 and 3 improved generalization through joint training and outperformed Model 1, whose segmentation and recognition components are trained independently.

Acknowledgments

This work was supported by JST ERATO Ishiguro Symbiotic Human-Robot Interaction program (Grant Number JPMJER1401), Japan.

References

Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai, and Xiaofei He. 2017. Dialogue act recognition via crf-attentive structured network. *CoRR*, cs.CL.

Ronan Collobert and Jason Weston. 2008. 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*, pages 160–167. ACM.

Ramón Granell, Stephen G. Pulman, and Carlos D. Martínez-Hinarejos. 2009. Simultaneous dialogue act segmentation and labelling using lexical and syntactic features. In *Proceedings of the SIGDIAL 2009 Conference, The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 11-12 September 2009, London, UK*, pages 333–336.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Edward Ivanovic. 2005. Dialogue act tagging for instant messaging chat sessions. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 79–84.

Yangfeng Ji, Gholamreza Haffari, and Jacob Eisenstein. 2016. A latent variable recurrent neural network for discourse relation language models. In *Proceedings of NAACL-HLT*, pages 332–342.

Shafiq R. Joty and Enamul Hoque. 2016. Speech act modeling of written asynchronous conversations with task-specific embeddings and conditional structured models. 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*.

Dan Jurafsky, Elizabeth Shriberg, and Debra Biasca. 1997. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *Institute of Cognitive Science Technical Report*, pages 97–102.

Hamed Khanpour, Nishitha Guntakandla, and Rodney Nielsen. 2016. Dialogue act classification in domain-independent conversations using a deep recurrent neural network. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 2012–2021.

Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, Sachindra Joshi, and Arun Kumar. 2017. Dialogue act sequence labeling using hierarchical encoder with CRF. *CoRR*, cs.CL.

Ji Young Lee and Franck Dernoncourt. 2016. Sequential short-text classification with recurrent and convolutional neural networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 515–520.

Wei Li and Yunfang Wu. 2016. Multi-level gated recurrent neural network for dialog act classification. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 1970–1979.

Ramesh R. Manuvinakurike, Maike Paetzel, Cheng Qu, David Schlangen, and David DeVault. 2016. Toward incremental dialogue act segmentation in fast-paced interactive dialogue systems. In *Proceedings of the SIGDIAL 2016 Conference, The 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 13-15 September 2016, Los Angeles, CA, USA*, pages 252–262.

Silvia Quarteroni, Alexei V. Ivanov, and Giuseppe Riccardi. 2011. Simultaneous dialog act segmentation and classification from human-human spoken conversations. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2011, May 22-27, 2011, Prague Congress Center, Prague, Czech Republic*, pages 5596–5599.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Sheng-syun Shen and Hung-yi Lee. 2016. Neural attention models for sequence classification: Analysis and application to key term extraction and dialogue act detection. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 2716–2720.

Vivek Kumar Rangarajan Sridhar, Srinivas Bangalore, and Shrikanth Narayanan. 2009. Combining lexical, syntactic and prosodic cues for improved online dialog act tagging. *Computer Speech & Language*, 23(4):407–422.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339–373.

Nick Webb, Mark Hepple, and Yorick Wilks. 2005. Dialogue act classification based on intra-utterance features. In *Proceedings of the AAAI Workshop on Spoken Language Understanding*, volume 4, page 5.

Matthias Zimmermann. 2009. Joint segmentation and classification of dialog acts using conditional random fields. In *INTERSPEECH 2009, 10th Annual Conference of the International Speech Communication Association, Brighton, United Kingdom, September 6-10, 2009*, pages 864–867.

Matthias Zimmermann, Andreas Stolcke, and Elizabeth Shriberg. 2006. Joint segmentation and classification of dialog acts in multiparty meetings. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP 2006, Toulouse, France, May 14-19, 2006*, pages 581–584.