Speaker Turn Modeling for Dialogue Act Classification

Zihao He[1,3] **Leili Tavabi**[1,2] **Kristina Lerman**[3] **Mohammad Soleymani**[2]
[1]Department of Computer Science, University of Southern California
[2]Institute for Creative Technologies, University of Southern California
[3]Information Sciences Institute, University of Southern California
zihaoh@usc.edu ltavabi@ict.usc.edu lerman@isi.edu soleymani@ict.usc.edu

Abstract

Dialogue Act (DA) classification is the task of classifying utterances with respect to the function they serve in a dialogue. Existing approaches to DA classification model utterances without incorporating the turn changes among speakers throughout the dialogue, therefore treating it no different than non-interactive written text. In this paper, we propose to integrate the turn changes in conversations among speakers when modeling DAs. Specifically, we learn conversation-invariant speaker turn embeddings to represent the speaker turns in a conversation; the learned speaker turn embeddings are then merged with the utterance embeddings for the downstream task of DA classification. With this simple yet effective mechanism, our model is able to capture the semantics from the dialogue content while accounting for different speaker turns in a conversation. Validation on three benchmark public datasets demonstrates superior performance of our model.[1]

1 Introduction

Dialogue Acts (DAs) are the functions of utterances in the context of a dialogue conveying the speaker's intent (Searle et al., 1969). In natural language understanding, DA classification is of critical importance, as it underlies various tasks such as dialogue generation (Li et al., 2017a) and intention recognition (Higashinaka et al., 2006), thus providing effective means for domains like dialogue systems (Higashinaka et al., 2014), talking avatars (Xie et al., 2014) and therapy (Xiao et al., 2016; Tavabi et al., 2021, 2020).

Recent studies of DA classification have leveraged deep learning techniques, where promising results have been observed. Generally, these methods utilize hierarchical Recurrent Neural Networks (RNNs) to model structural information between utterances, words, and characters (Raheja and Tetreault, 2019; Li et al., 2018; Wan et al., 2018; Chen et al., 2018; Kumar et al., 2018; Bothe et al., 2018). However, most of these approaches treat a spoken dialogue similar to written text, thereby neglecting to explicitly model turn-taking across different speakers. Inherently, computational understanding of a dialogue, which has been generated by multiple parties with different goals and habits in an interactive and uncontrolled environment (Chi et al., 2017), requires modeling turn-taking behavior and temporal dynamics of a conversation. For instance, in a dyadic conversation, given an utterance with dialogue act "Question" from speaker A, if the following utterance is from speaker B, then the corresponding act is likely to be "Answer"; however, if there is no change in speakers, then the following act is less likely to be "Answer." Therefore, modeling turn changes in conversations is essential.

In this regard, we aim to incorporate the speaker turns into encoding an utterance. Specifically, we propose to model speaker turns in conversations and introduce two speaker turn embeddings that are combined with the utterance embeddings. Given a conversation containing a sequence of utterances, we first obtain the utterance embeddings using a large pretrained language model RoBERTa (Liu et al., 2019), and extracting the [CLS] token embeddings from the last layer; meanwhile we use a speaker turn embedding layer to generate speaker turn embeddings given the speaker labels. The speaker turn embeddings are added to the utterance embeddings to obtain speaker turn-aware utterance representations. These representations are fed into an RNN to encode the context of the conversation, where the output hidden states are used for DA classification. We evaluate the proposed method on three benchmark datasets and achieve the state-of-the-art results, among all inductive learning meth-

[1] Our code and data are publicly available at `https://github.com/ZagHe568/speak-turn-emb-dialog-act-clf`.

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 2150–2157
November 7–11, 2021. ©2021 Association for Computational Linguistics

ods, on two of the datasets. We argue that this simple technique provides effective means for obtaining more powerful representations for dialogue.

2 Related Work

Dialogue Act Classification. Chen et al. (2018) propose a CRF-attentive structured network and apply structured attention network to the CRF (Conditional Random Field) layer in order to simultaneously model contextual utterances and the corresponding DAs. Li et al. (2018) introduce a dual-attention hierarchical RNN to capture information about both DAs and topics, where the best results are achieved by a transductive learning model. Raheja and Tetreault (2019) utilize a context-aware self-attention mechanism coupled with a hierarchical RNN. Colombo et al. (2020) leverage the seq2seq model to learn the global tag dependencies instead of the widely used CRF that captures local dependencies; this method, however, requires beam search that introduces more complexity. The aforementioned methods are based on hierarchical RNNs and neglect speaker turns modelled in this paper.

Speaker Role Modeling in Dialogues. Existing work mainly focus on speaker roles for the purpose of encoding dialogue context in conversations, involving distinguishable speaker roles like guide versus tourist. For encoding role-based context information, Chi et al. (2017) and Chen et al. (2017) use individual recurrent modules for each speaker role, modeling the role-dependent goals and speaking styles, and taking the sum of the resulting representations from each speaker. Similarly, Hazarika et al. (2018) obtain history context representations per speaker by modeling separate memory cells using Gated Recurrent Units (GRUs) for each speaker; therefore speaker-based histories undergo identical but separate computations before being combined for the downstream task. Qin et al. (2021) treat an utterance as a vertex and add an edge between utterances of the same speakers to construct cross-utterances connections; such connections are based on specific speaker roles. Different from speaker role-based methods, our method focuses on speaker turns and thus is still useful when speakers are not associated with specific roles. Additionally, previous methods incorporate speaker information by proposing more complex and specialized models, which inevitably introduce a large number of parameters to train, whereas we introduce two global *additive* embedding vectors, requiring negligible modifications to a recurrent model and introducing $O(1)$ space complexity, as can be seen in Section 3.3.

3 Methods

The overall framework of our model is shown in Figure 1. In this section, we will describe our model's components in detail.

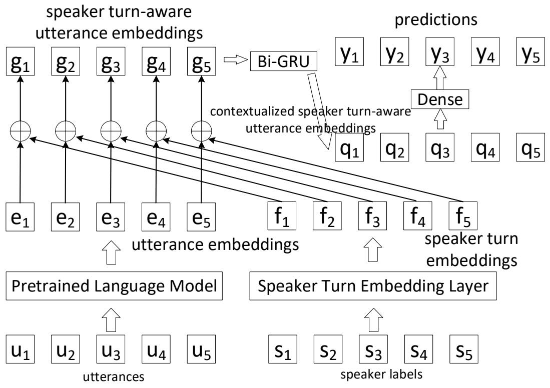

Figure 1: The overall framework of our proposed method. In this toy example, the conversation consists of five utterances.

3.1 Problem Definition

The input corpus $D = \{(C_n, Y_n, S_n)\}_{n=1}^{N}$ consists of N conversations, where $C_n = \langle u_t^n \rangle_{t=1}^{T}$ is a dialogue instance containing a sequence of T utterances, $Y_n = \langle y_t^n \rangle_{t=1}^{T}$ and $S_n = \langle s_t^n \rangle_{t=1}^{T}$ are the corresponding DA labels and speaker labels. The goal is to learn a model from corpus D, such that given an unseen conversation C_p and its corresponding speaker labels S_p, the model is able to predict the DA labels Y_p of utterances in C_p.

3.2 Utterance Modeling

We use the pretrained language model RoBERTa to encode utterances, which enables us to utilize the powerful representations obtained from pretraining on large amounts of data. Given an utterance u, we take the embedding of [CLS] token from the last layer as the utterance embedding, denoted as $e(u)$.

3.3 Speaker Turn Modeling

Different from text written by a single author in a non-interactive environment, dialogues usually involve multiple parties, in minimally-controlled environments where each speaker has their own goals and speaking styles (Chi et al., 2017). Therefore, it is critically important to model how speakers take turns individually and inform the model when there is a speaker turn change. To this end, for a dyadic conversation corpus, we introduce two conversation-invariant speaker turn embeddings, for each interlocutor. These two embeddings are trained across all speakers in the train set and are independent of any given conversation or speaker pair. The two embeddings are learnable parameters during the optimization and have the same size as the utterance embeddings, which are generated by a speaker turn embedding layer with speaker labels as input. Note that in a dyadic conversation, speaker labels $(0/1)$ naturally indicate speaker turn changes. This idea is inspired by the positional encoding in Transformers (Vaswani et al., 2017), where the authors introduce a positional embedding with the same size as the token embedding at each position, and the positional embeddings are shared across different input sequences. For a multi-party conversation corpus, because our goal is to model speaker turns instead of assigning a different embedding to each speaker, we relabel the speakers and flip the speaker label (from 0 to 1 and vice versa) when there is speaker turn change; for example, if the original speaker sequence is $\langle 0, 0, 1, 2, 3, 3, 1\rangle$, after relabeling it becomes $\langle 0, 0, 1, 0, 1, 1, 0\rangle$, which can then be represented by the two introduced speaker turn embeddings. This simplifies turn-change modeling, as the number of speakers in different conversations can be different.

Encoding speaker turns instead of individual speaker styles/characteristics provides the following advantages: 1) in datasets with many different speakers across relatively short dialogue sessions, it is challenging to transfer the learned speaker representations across different sessions; 2) the simplicity of this mechanism makes it more scalable for multi-party dialogue sessions with larger number of speakers.

To obtain the speaker turn-aware utterance embedding $g(u, s)$, given an utterance u and its binary speaker turn label s, the speaker turn embedding $f(s)$ is then added to the utterance embedding $e(u)$, such that $g(u, s) = e(u) + f(s), s \in \{0, 1\}$. The idea of taking the sum is also inspired by Transformers where they add the positional embeddings to token embeddings for sequence representation (Vaswani et al., 2017). We also considered the concatenation of the speaker turn embedding and the utterance embedding, resulting in inferior performance compared to taking the sum.

3.4 Conversational Context Modeling

Context plays an important role in modeling dialogue, which should be taken into account when performing DA classification. Given a sequence of independently encoded speaker turn-aware utterance embeddings $\langle g(u_t, s_t)\rangle_{t=1}^{n}$ in conversation C, we used a Bi-GRU (Cho et al., 2014) to inform each utterance of its context, such that $\langle q(u_t, s_t)\rangle_{t=1}^{n} = \text{GRU}\langle g(u_t, s_t)\rangle_{t=1}^{n}$, where $\langle q(u_t, s_t)\rangle_{t=1}^{n}$ are contextualized speaker turn-aware utterance embeddings from the hidden states of the Bi-GRU model. These embeddings are then fed into a fully connected layer for DA classification, which is optimized using a cross-entropy loss. Different from existing work (Raheja and Tetreault, 2019; Li et al., 2018; Wan et al., 2018; Chen et al., 2018; Kumar et al., 2018; Bothe et al., 2018), we do not use a CRF layer in our method, because our experiments indicate that it brings modest performance gains at the expense of adding more complexity.

4 Experiments and Results

4.1 Datasets

We evaluate the performance of our model on three public datasets: the Switchboard Dialogue Act Corpus (SwDA) (Jurafsky, 1997; Shriberg et al., 1998; Stolcke et al., 2000), the Meeting Recorder Dialogue Act Corpus (MRDA) (Shriberg et al., 2004), and the Dailydialog (DyDA) (Li et al., 2017b). **SwDA**[2] contains dyadic telephone conversations labeled with 43 DA classes; the conversations are assigned to 66 manually-defined topics. **MRDA**[3] consists of multi-party meeting conversations and 5 DA classes. **DyDA**[4] corpus consists of human-written daily dyadic conversations labeled with 4 DA classes; the conversations are assigned to 10 topics. For SwDA and MRDA, we use the train, validation and test splits following (Lee and Dernoncourt, 2016). For DyDA, we use its original splits (Li et al., 2017b). The statistics of the three datasets are summarized in Table 1.

[2] https://github.com/cgpotts/swda

[3] https://github.com/NathanDuran/MRDA-Corpus

[4] http://yanran.li/dailydialog

Dataset	$\|C\|$	$\|P\|$	Train	Val	Test
SwDA	43	2	1003/193K	112/20K	19/4.5K
MRDA	5	multiple	51/75k	11/15.3K	11/15K
DyDA	5	2	11K/87.1K	1K/8K	1K/7.7K

Table 1: The statistics of the three datasets. $|C|$ denotes the number of DA classes; $|P|$ denotes the number of parties; Train/Val/Test denotes the number of conversations/utterances in the corresponding split.

4.2 Experimental Setup

On SwDA and DyDA, which are two dyadic conversation corpora, we use the original speaker labels (due to equivalence to speaker turn change labels); however, since MRDA is a multi-party conversation corpus, we use the binary speaker turn change labels obtained from the sequence of speaker labels as mentioned in Section 3.3. On DyDA, because the maximum length of conversations (number of utterances) is less than 50, we treat each conversation as a data point and pad all conversations to the maximum length. However, conversations in SwDA and MRDA are much lengthier (up to 500 in SwDA and 5,000 in MRDA); to avoid memory overflow when training on a GPU, we slice the conversations into shorter fixed-length chunk sizes of 128 and 350 for SwDA and MRDA respectively, as shown in Figure 2, where each chunk would represent a data point. The slicing operation is only needed for training but not in the validation or test, because in training a computation graph is maintained which consumes significantly more GPU memory. More details about the setup is reported in Appendix A. The results using different chunk sizes are reported in Appendix B.

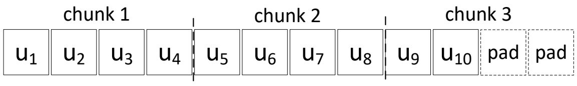

Figure 2: A toy example of slicing a conversation of length 10 into 3 chunks of length of 4.

4.3 Baselines

We consider deep learning based approaches as baselines including DRLM-Cond (Ji et al., 2016), Bi-LSTM-CRF (Kumar et al., 2018), CRF-ASN (Chen et al., 2018), ALDMN (Wan et al., 2018), SelfAtt-CRF (Raheja and Tetreault, 2019), SGNN (Ravi and Kozareva, 2018), DAH-CRF-Manual (Li et al., 2018), and Seq2Seq (Colombo et al., 2020). We report the results of DRLM-Cond and Bi-LSTM-CRF on DyDA implemented by (Li et al., 2018). Our proposed speaker turn modeling is usable in other embedding-based approaches to DA classification, but because none of the recently published work have made the code available, we do not implement the proposed speaker turn modeling on top of the baselines.

For fair comparison with DAH-CRF-Manual$_{conv}$ (Li et al., 2018) where manual conversation-level topic labels are used, we assign all utterances in a conversation the corresponding conversation topic label. To utilize the topic information, following the idea of speaker turn embedding in Section 3.3, we introduce an embedding $h(m)$ for each topic m and add it to the speaker turn-aware utterance embedding, such that $l(u, s, m) = g(u, s) + h(m)$ where $l(u, s, m)$ is the obtained speaker turn and topic-aware utterance embedding.

Note that we do not compare our results to DAH-CRF-LDA$_{conv}$ and DAH-CRF-LDA$_{utt}$ (Li et al., 2018), which are categorized as transductive learning because they utilize the data from training, validation and test sets to perform LDA topic modeling and use the learned topic labels to supervise the training process. In contrast, our method and all baselines are categorized as inductive learning, which do not use supervision from the validation or test set. In addition, we do not compare to Seq2Seq (Colombo et al., 2020) on SwDA where they adopt a different test split from the one used in our method and the baselines.

4.4 Results

The results from our method and the baselines are shown in Table 2. Our method achieves state-of-the-art results on SwDA and DyDA; on MRDA it achives the performance comparable to the state-of-the-art. Notably, on SwDA and MRDA, comparing the proposed model (Ours) to the model without speaker turn embeddings (Ours¬Speaker), we observe significant improvements in performance, signifying the effectiveness of modeling speaker turns in dialogue representation. On DyDA, the performance gains slightly after applying speaker turn modeling; we argue that this is because in conversations in DyDA, there is a consistent speaker turn change after each utterance following the pattern $\langle 0, 1, 0, 1, 0, 1\rangle$; such a pattern is more predictable, and therefore, modeling speaker turns provides limited auxiliary information, from the perspective of information theory.

In addition, on DyDA, the model Ours¬Speaker

Dataset	SwDA	MRDA	DyDA
DRLM-Cond	77.0	88.4	81.1
Bi-LSTM-CRF	79.2	90.9	83.6
CRF-ASN	80.8	91.4	-
ALDMN	81.5	-	-
SelfAtt-CRF	82.9	91.1	-
SGNN	83.1	86.7	-
DAH-CRF-Manual	80.9	-	86.5
Seq2Seq	-	**91.6**	-
Ours¬Speaker	82.4	90.7	86.8
Ours	**83.2**	91.4	86.9
Ours+Topic	82.4	-	**87.5**

Table 2: Results of DA classification on three different methods. "Ours¬Speaker" represents our method without adding speaker turn embeddings; "Ours+Topic" represents the proposed method using speaker turn and topic-aware embeddings for fair comparison to baselines utilizing topic information. State-of-the-art results are highlighted in bold.

outperforms the baselines, although this is not observed on SwDA and MRDA. We hypothesize that the reason may be from the fact that RoBERTa (Liu et al., 2019) is pretrained on a large corpus of written text, which will make it better for processing the human-written conversations in DyDA, in comparison to the transcripts of telephone conversations and meeting records in SwDA and MRDA. As a result, the generated utterance embeddings are of higher quality, leading to the high performance of Ours¬Speaker on DyDA.

In terms of modeling topics, on DyDA, topic information significantly improves the classification performance; in contrast, on SwDA, the performance suffers when utilizing topic information, as can be observed from the comparison of Ours and Ours+Topic. Therefore, leveraging topic labels does not consistently lead to performance improvement; on the other hand, it is consistently improved by encoding speaker turn changes on all three datasets.

5 Conclusion and Future Work

In this paper, we propose a model for encoding speaker turn changes to tackle DA classification. Specifically, we introduce conversation-invariant speaker turn embeddings and add them to utterance embeddings produced by a pretrained language model. Such a simple yet scalable module can be easily added to other models to obtain significantly better results. Experiments on three datasets demonstrate the effectiveness of our method. For future work, we will explore transformer encoders (Vaswani et al., 2017) instead of RNNs for encoding context, since they have shown to be advantageous in performance and training time. Our improved representations can be further utilized in other downstream tasks involving dialogue, including speaker intent classification. Our findings motivate future work on encoding other interactive aspects of dialogue data into existing text representations.

Acknowledgements

Research was sponsored in-part by the Army Research Office and was accomplished under Cooperative Agreement Number W911NF-20-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

Chandrakant Bothe, Cornelius Weber, Sven Magg, and Stefan Wermter. 2018. A context-based approach for dialogue act recognition using simple recurrent neural networks. *arXiv preprint arXiv:1805.06280*.

Po-Chun Chen, Ta-Chung Chi, Shang-Yu Su, and Yun-Nung Chen. 2017. Dynamic time-aware attention to speaker roles and contexts for spoken language understanding. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 554–560. IEEE.

Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai, and Xiaofei He. 2018. Dialogue act recognition via crf-attentive structured network. In *The 41st international acm sigir conference on research & development in information retrieval*, pages 225–234.

Ta-Chung Chi, Po-Chun Chen, Shang-Yu Su, and Yun-Nung Chen. 2017. Speaker role contextual modeling for language understanding and dialogue policy learning. *arXiv preprint arXiv:1710.00164*.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Pierre Colombo, Emile Chapuis, Matteo Manica, Emmanuel Vignon, Giovanna Varni, and Chloe Clavel. 2020. Guiding attention in sequence-to-sequence models for dialogue act prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7594–7601.

Devamanyu Hazarika, Soujanya Poria, Amir Zadeh, Erik Cambria, Louis-Philippe Morency, and Roger Zimmermann. 2018. Conversational memory network for emotion recognition in dyadic dialogue videos. In *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2018, page 2122. NIH Public Access.

Ryuichiro Higashinaka, Kenji Imamura, Toyomi Meguro, Chiaki Miyazaki, Nozomi Kobayashi, Hiroaki Sugiyama, Toru Hirano, Toshiro Makino, and Yoshihiro Matsuo. 2014. Towards an open-domain conversational system fully based on natural language processing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 928–939.

Ryuichiro Higashinaka, Katsuhito Sudoh, and Mikio Nakano. 2006. Incorporating discourse features into confidence scoring of intention recognition results in spoken dialogue systems. *Speech Communication*, 48(3-4):417–436.

Yangfeng Ji, Gholamreza Haffari, and Jacob Eisenstein. 2016. A latent variable recurrent neural network for discourse relation language models. *arXiv preprint arXiv:1603.01913*.

Dan Jurafsky. 1997. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *Institute of Cognitive Science Technical Report*.

Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, and Sachindra Joshi. 2018. Dialogue act sequence labeling using hierarchical encoder with crf. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ji Young Lee and Franck Dernoncourt. 2016. Sequential short-text classification with recurrent and convolutional neural networks. *arXiv preprint arXiv:1603.03827*.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017a. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*.

Ruizhe Li, Chenghua Lin, Matthew Collinson, Xiao Li, and Guanyi Chen. 2018. A dual-attention hierarchical recurrent neural network for dialogue act classification. *arXiv preprint arXiv:1810.09154*.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017b. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Libo Qin, Zhouyang Li, Wanxiang Che, Minheng Ni, and Ting Liu. 2021. Co-gat: A co-interactive graph attention network for joint dialog act recognition and sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13709–13717.

Vipul Raheja and Joel Tetreault. 2019. Dialogue act classification with context-aware self-attention. *arXiv preprint arXiv:1904.02594*.

Sujith Ravi and Zornitsa Kozareva. 2018. Self-governing neural networks for on-device short text classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 887–893.

John R Searle, PG Searle, S Willis, John Rogers Searle, et al. 1969. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press.

Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang, and Hannah Carvey. 2004. The icsi meeting recorder dialog act (mrda) corpus. Technical report, INTERNATIONAL COMPUTER SCIENCE INST BERKELEY CA.

Elizabeth Shriberg, Andreas Stolcke, Daniel Jurafsky, Noah Coccaro, Marie Meteer, Rebecca Bates, Paul Taylor, Klaus Ries, Rachel Martin, and Carol Van Ess-Dykema. 1998. Can prosody aid the automatic classification of dialog acts in conversational speech? *Language and speech*, 41(3-4):443–492.

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.

Leili Tavabi, Kalin Stefanov, Larry Zhang, Brian Borsari, Joshua D Woolley, Stefan Scherer, and Mohammad Soleymani. 2020. Multimodal automatic coding of client behavior in motivational interviewing. In *Proceedings of the 2020 International Conference on Multimodal Interaction*, pages 406–413.

Leili Tavabi, Trang Tran, Kalin Stefanov, Brian Borsari, Joshua Woolley, Stefan Scherer, and Mohammad Soleymani. 2021. Analysis of behavior classification in motivational interviewing. In *Proceedings of the Seventh Workshop on Computational Linguistics and Clinical Psychology: Improving Access*, pages 110–115.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yao Wan, Wenqiang Yan, Jianwei Gao, Zhou Zhao, Jian Wu, and S Yu Philip. 2018. Improved dynamic memory network for dialogue act classification with adversarial training. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 841–850. IEEE.

Bo Xiao, Dogan Can, James Gibson, Zac E Imel, David C Atkins, Panayiotis G Georgiou, and Shrikanth S Narayanan. 2016. Behavioral coding of therapist language in addiction counseling using recurrent neural networks. In *Interspeech*, pages 908–912.

Lei Xie, Naicai Sun, and Bo Fan. 2014. A statistical parametric approach to video-realistic text-driven talking avatar. *Multimedia tools and applications*, 73(1):377–396.

A Experimental Setup

The maximum feasible chunk sizes without a CUDA memory overflow, on our machines with 11GB of RAM, are 300(>128) and 700(>350) on SwDA and MRDA respectively, which indicates that using the entire un-sliced conversation is not necessary and will lead to performance deterioration due to the gradient vanishing and gradient explosion problems in RNN.

We implement our model using PyTorch and train our model using Adam optimizer on 2 GTX 1080Ti GPUs. On SwDA and MRDA, we use a batch size of 2; and on DyDA, the batch size is 10. All batch sizes are the maximum before a memory overflow happens. On all three datasets, we use a learning rate of $1e-4$ and train the model for a maxium of 50 epochs and report the test accuracy in the epoch where the best validation accuracy is achieved. The running time for an epoch are ~20min, ~5min, and ~45min on SwDA, MRDA and DyDA respectively.

B Effect of Chunk Sizes

Keeping other hyperparameters unchanged, we show the results of using different chunk sizes on SwDA and MRDA in Table 3 and Table 4 respectively. On both datasets, with the chunk size increasing from a small value, the performance increases, where more context information is available to the RNN to leverage. However, after a certain value, the performance deteriorates as the chunk size further increases, in which case the gradient vanishing and gradient explosion happens in RNN and it forgets the long-term dependencies. Therefore, we argue that in order to achieve better performance in DA classification, taking the holistic conversation as input leads to inferior performance compared to slicing a long conversation into shorter chunks.

chunk_size	32	64	85	128
acc	82.9	82.7	82.8	83.2
chunk_size	160	196	256	300
acc	82.7	83.0	82.9	82.3

Table 3: The accuracies using different chunk sizes on SwDA.

chunk_size	85	175	350	700
acc	91.3	91.1	91.4	91.3

Table 4: The accuracies using different chunk sizes on MRDA.