# Modeling Long-Range Context for Concurrent Dialogue Acts Recognition

Yue Yu* [1] Siyao Peng* [2] Grace Hui Yang [1]

[1]Department of Computer Science [2]Department of Linguistics
Georgetown University

## Introduction

- In dialogues, the existence of multiple dialogue acts (DAs) in one utterance requires DA models to handle long utterances and complex DA context. We call this task: **Concurrent Dialogue Acts (CDA) recognition.**
- We propose a **Convolutional Recurrent Neural Network (CRNN)** model for sequence prediction, which imposes fewer restrictions on the structure of DAs and captures textual features from a wider context.

## Task Definition

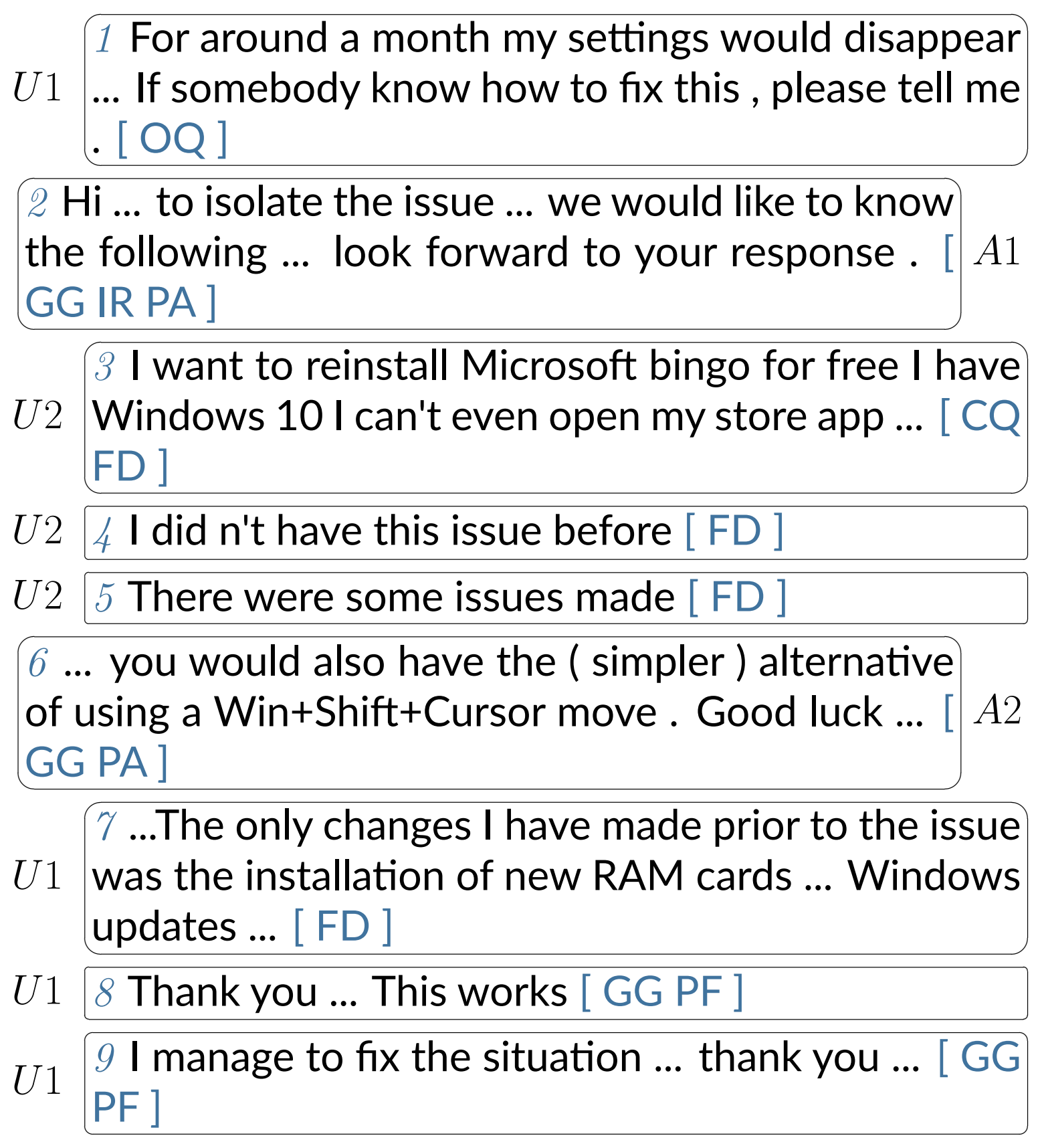
The task is defined as a CDA recognition problem where for each utterance $u_t$ (the $t$-th utterance) in a dialogue, we predict a subset of DA labels $y_t$ that describes the functionality of the utterance from a candidate set of DA labels $\mathcal{L} = \{l_1, l_2, ..., l_c\}$. For a dialog with $s$ utterances, the inputs to the algorithm is $\mathcal{U} = \{u_1, u_2, ..., u_s\}$, and the output is $\mathcal{Y} = \{y_1, y_2, ..., y_s\}$, where $y_t$ is the annotated DA label set for $u_t$, in which $y_t = \{y_t^1, y_t^2, ..., y_t^c\}$. Here, $y_t^j = \{1, 0\}$ denotes whether the $t$-th utterance of the dialog is labeled with DA label $l_j$ or not. When $\sum_{j=1}^{c} y_t^j > 1$, we say CDAs are recognized. Given a dialogue $\mathcal{U}$, the goal is to predict the DA sequence $\mathcal{Y}$ from the text.

## Dataset & Examples

We use the MSDialog-Intent dataset [4] where each of the 10,020 utterances is annotated with a subset of 12 DAs. The abundance of information in a single utterance (avg. 72 tokens/utterance) breeds CDA (avg. 1.83 DAs/utterance). We observe a strong correlation between the number of DAs and utterance length which necessitates a CDA model.

| *DA* | *Taxonomy* | *%* | *DA* | *Taxonomy* | *%* |
|---|---|---|---|---|---|
| GG | Greetings/Gratitude | 40.0 | FQ | Follow-up Question | 8.6 |
| PA | Potential Answer | 39.7 | NF | Negative Feedback | 7.6 |
| FD | Further Details | 24.8 | CQ | Clarifying Question | 7.5 |
| OQ | Original Question | 23.3 | RQ | Repeat Question | 6.1 |
| PF | Positive Feedback | 10.7 | JK | Junk | 2.6 |
| IR | Information Request | 10.7 | O | Others | 1.5 |

Table 1: Taxonomy of 12 DAs in a tech forum.



$U1$ $1$ For around a month my settings would disappear ... If somebody know how to fix this , please tell me . [ OQ ]

$2$ Hi ... to isolate the issue ... we would like to know the following ... look forward to your response . [ GG IR PA ] $A1$

$U2$ $3$ I want to reinstall Microsoft bingo for free I have Windows 10 I can't even open my store app ... [ CQ FD ]

$U2$ $4$ I did n't have this issue before [ FD ]

$U2$ $5$ There were some issues made [ FD ]

$6$ ... you would also have the ( simpler ) alternative of using a Win+Shift+Cursor move . Good luck ... [ GG PA ] $A2$

$U1$ $7$ ...The only changes I have made prior to the issue was the installation of new RAM cards ... Windows updates ... [ FD ]

$U1$ $8$ Thank you ... This works [ GG PF ]

$U1$ $9$ I manage to fix the situation ... thank you ... [ GG PF ]

Figure 1: A dialogue between 2 users and 2 agents.

## Approach

### Convolutional Layer

Our model is based on a CNN module [2]. The module works by 'sliding' through the embedding matrix of an utterance with various filter sizes to capture semantic features in differently ordered n-grams. The convolution operation is applied to every possible window of words in an utterance of length $n$ and generates a feature map $\mathbf{k}$.

$$\mathbf{k} = [k_1, k_2, ..., k_{n-d+1}] \tag{1}$$

### Dynamic $k$-Max Pooling

A max-over-time pooling operation [2] is applied over the feature map and takes the maximum value as the feature corresponding to this particular filter. The idea is to capture the most important features of an utterance. We use Dynamic $k$-Max Pooling [3] to pool the most powerful features from $p$ sub-sequences of an utterance with $m$ words which accommodates variable utterance length.

$$p(\mathbf{k}) = \left[\max\left\{\mathbf{k}_{1:\lfloor\frac{m}{p}\rfloor}\right\}, \ldots, \max\left\{\mathbf{k}_{\lfloor m-\frac{m}{p}+1\rfloor:m}\right\}\right] \tag{2}$$

### Recurrent Layer

Bidirectional RNNs, i.e. LSTM [6] and GRU [1], are applied to gather features from a wider context for recognizing the DAs in the target utterance, $u_t$.

### Highway Connection

We add highway connections [5] between the convolutional layer and the last fully connected layer. The information about the target utterance, $u_t$, can flow across the recurrent layer without attenuation. Thus, the fully connected layer learns directly from both recurrent and convolutional layers.

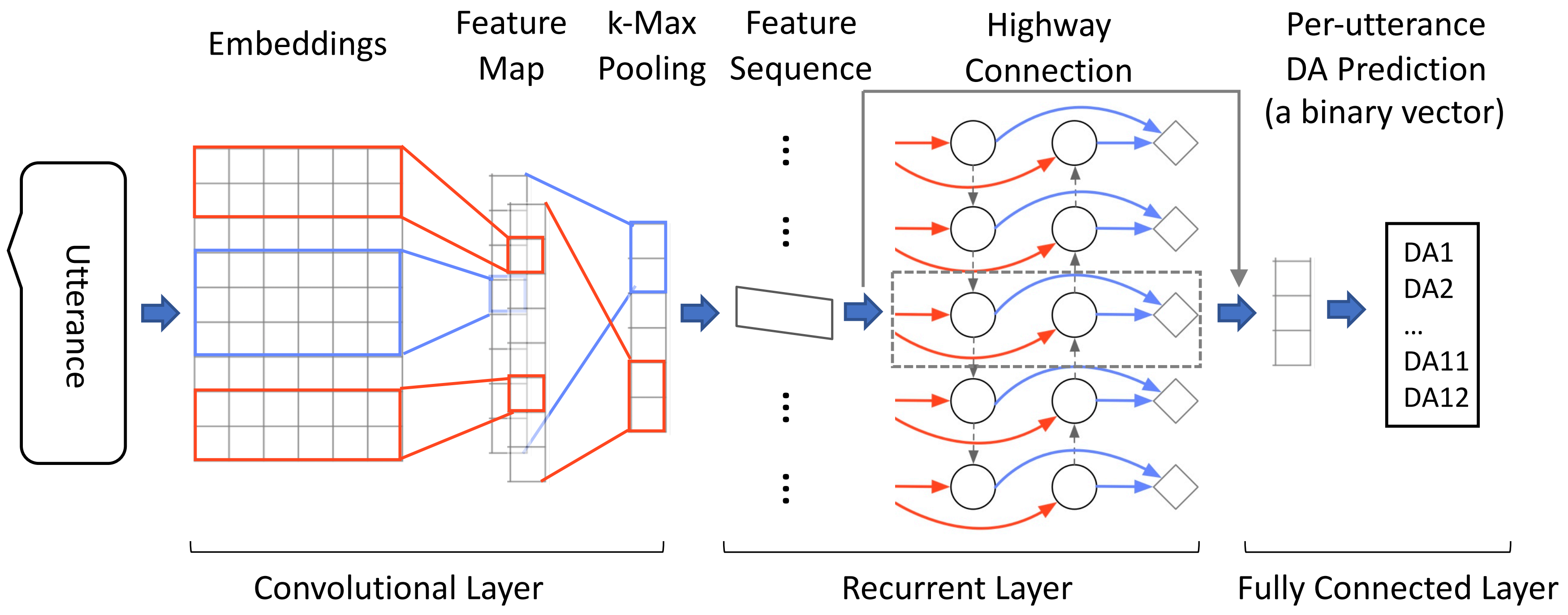


Figure 2: The proposed CRNN model architecture.

## Experiments

Three experiments with incremental improvements are evaluated against a CNN baseline [2] and the state-of-the-art approach for CDA recognition [4].

- **CNN-Kim[2]:** One of the first attempts to apply CNN to text classification.
- **CNN-CR[4]:** The SOTA approach for CDA recognition on the MSDialog-Intent dataset [4] that incorporates context with *windowsize=3*.
- **CRNN** ($v_1$)**:** Our base model using BCE loss and sigmoid activation function.
- **CRNN** ($v_2$)**:** CRNN ($v_1$) + highway connections.
- **CRNN** ($v_3$)**:** CRNN ($v_1$) + highway connections + dynamic $k$-max pooling.

## Results

Our CRNN models ($v_3$ especially) outperform both baselines in terms of:

1. Highest accuracy, recall and $F_1$ with LSTM; and precision with GRU (Table 2).
2. Higher accuracy for all reference DA sizes and especially for 1-3 DAs (Table 3).
3. Closer average number of predicted DAs for each utterance (Table 3).
4. Higher mean accuracy for dialogues with more than 6 utterances (Figure 3).

| Models | **Accuracy** | Precision | Recall | $F_1$ score |
|---|---|---|---|---|
| CNN-Kim[2] | 0.5785 | 0.6371 | 0.6745 | 0.6553 |
| CNN-CR[4] | 0.6354 | 0.7108 | 0.6952 | 0.7029 |
| CRNN ($v_1$) w/ LSTM | 0.6668* | 0.7238 | 0.7297 | 0.7267 |
| CRNN ($v_1$) w/ GRU | 0.6543* | 0.7056 | 0.7065 | 0.7061 |
| CRNN ($v_2$) w/ LSTM | 0.6731**** | 0.7315 | 0.7315 | 0.7315 |
| CRNN ($v_2$) w/ GRU | 0.6734** | 0.7280 | 0.7334 | 0.7307 |
| CRNN ($v_3$) w/ LSTM | **0.6822****** | 0.7254 | **0.7422** | **0.7337** |
| CRNN ($v_3$) w/ GRU | 0.6733*** | **0.7358** | 0.7215 | 0.7286 |

Table 2: Performance of CNN-Kim, CNN-CR, & CRNN.

| # of ref DAs | % | Mean accuracy | | Avg. num. of pred DAs | |
|---|---|---|---|---|---|
| | | CRNN ($v_3$) | CNN-CR | CRNN ($v_3$) | CNN-CR |
| 1 | 36.9 | **0.7704**** | 0.7126 | 1.44 | 1.44 |
| 2 | 42.8 | **0.6641***** | 0.6232 | **2.02**** | 1.89 |
| 3 | 16.7 | **0.5596*** | 0.5177 | **2.56***** | 2.37 |
| ≥4 | 3.6 | **0.5618** | 0.5339 | **2.68** | 2.74 |

Table 3: Mean accuracy and average number of predicted DAs grouped by the number of reference DAs. The percentage indicates the frequency of each DA group in the test set.

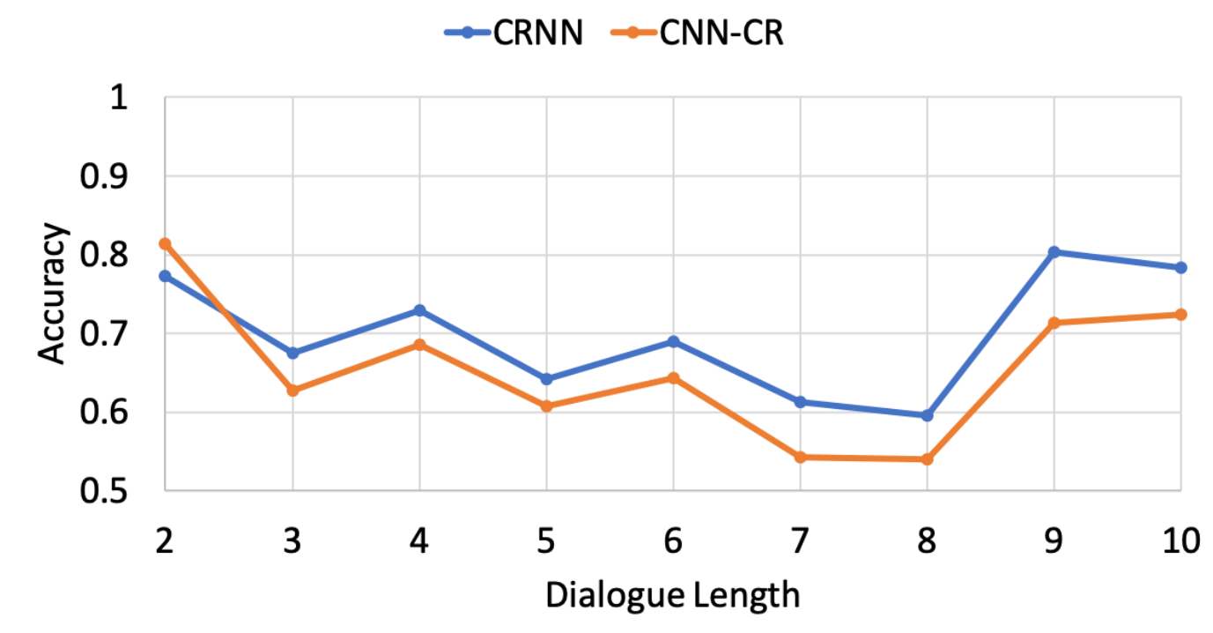


Figure 3: CRNN ($v_3$) vs. CNN-CR over dialogues of different length.

## Conclusion

- Our CRNN models achieve the new SOTA for CDA recognition on a tech forum dataset, where the dialogues are packed with complex DA structures and information-rich utterances.
- Our best model significantly outperforms CNN-CR[4] on accuracy by $4.68\%$; $1.46\%$ on Precision, $4.70\%$ on Recall, and $3.08\%$ on $F_1$.
- All of our proposed adaptations, i.e. highway connections and dynamic $k$-max pooling, contribute to the model.

## Acknowledgements

This research is supported by U.S. National Science Foundation IIS-145374. Any opinions, findings, conclusions, or recommendations expressed in this paper are of the authors, and do not necessarily reflect those of the sponsor.

Paper Poster Slides

* Both authors contributed equally to this research.

* for p ≤ 0.05, ** for p ≤ 0.01, *** for p ≤ 0.001 and **** for p ≤ 0.0001

[1] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).
[2] Y. Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *EMNLP'14*. 1746–1751.
[3] J. Liu, W.-C. Chang, Y. Wu, and Y. Yang. 2017. Deep Learning for Extreme Multi-label Text Classification. In *SIGIR '17*. New York, NY, USA, 115–124.
[4] C. Qu, L. Yang, B. Croft, Y. Zhang, J. R. Trippas, and M. Qiu. 2019. User Intent Prediction in Information-seeking Conversations. *CHIIR '19* (2019).
[5] P. Sanders and D. Schultes. 2005. Highway hierarchies hasten exact shortest path queries. In *European Symposium on Algorithms*. Springer, 568–579.
[6] W. Zaremba and I. Sutskever. 2014. Learning to execute. (2014).