Computer Speech & Language 63 (2020) 101073

Contents lists available at ScienceDirect

Computer Speech & Language

journal homepage: www.elsevier.com/locate/csl

Learning Multi-Level Information for Dialogue Response Selection by Highway Recurrent Transformer

Ting-Rui Chiang, Chao-Wei Huang, Shang-Yu Su, Yun-Nung Chen Ph.D.*

National Taiwan University, No. 1, Sec. 4, Roosevelt Rd., Taipei 10617, Taiwan

ARTICLE INFO

Article History:
Received 26 July 2019
Revised 12 January 2020
Accepted 21 January 2020
Available online 3 February 2020

Keywords:
Response selection
Transformer
Attention mechanism
Dialogue
DSTC

ABSTRACT

With increasing research interests in dialogue modeling, there is an emerging branch that formulates this task as next sentence selection, where given the partial dialogue context, the goal is to determine the most probable next sentence. To model natural language information, recurrent models have been applied to sequence modeling and shown promising results in various NLP tasks (Sutskever et al., 2014). Recently, the Transformer (Vaswani et al., 2017) has advanced modeling semantics for natural language sentences via attention, achieving improvement for sequence modeling. However, the Transformer focuses on modeling the intra-sentence attention but ignores inter-sentence information. In terms of dialogue modeling, the cross-sentence information is salient to understand dialogue content, so that the response selection can be better determined. Therefore, this paper proposes a novel attention mechanism based on multi-head attention, called *highway attention*, in order to allow the model to pass information through multiple sentences, and then builds a recurrent model based on the Transformer and the proposed highway attention. We call this model *Highway Recurrent Transformer*. This model focuses on not only intra-sentence dependency, but also inter-sentence dependency in the structure of dialogues. Experiments on the response selection task of the seventh Dialog System Technology Challenge (DSTC7) demonstrate that the proposed Highway Recurrent Transformer is capable of modeling both *utterance-level* and *dialogue-level* information for achieving better performance than the original Transformer in the single positive response scenario.

1. Introduction

With an increasing focus on dialogue modeling, response selection and generation have been widely studied in the NLP community. In order to further push the current capability of machine learning models, a benchmark dataset was proposed in the seventh Dialog System Technology Challenge (DSTC7) (D'Haro et al., 2020), where the task is to select the most probable response given a partial conversation. In this task, generalization should be examined; hence, two datasets, Ubuntu IRC dialogs (Chulaka Gunasekara and Lasecki, 2019; Kummerfeld et al., 2018) and course advising corpus (Chulaka Gunasekara and Lasecki, 2019), are utilized for the experiments. These datasets have very different properties: (1) the Ubuntu dataset includes dialogues from an Ubuntu IRC channel that aim to solve technical problems, and (2) the advising dataset comprises conversations between a student and an advisor, where the advisor helps the student with course taking. Compared with the advising dataset, utterances in a dialogue from the Ubuntu data are more coherent, so selecting the next sentence may require understanding of previous

*Corresponding author.
E-mail addresses: r07922052@csie.ntu.edu.tw (T.-R. Chiang), r07922069@csie.ntu.edu.tw (C.-W. Huang), f05921117@csie.ntu.edu.tw (S.-Y. Su), y.v.chen@ieee.org (Y.-N. Chen).

https://doi.org/10.1016/j.csl.2020.101073
0885-2308/

dialogue turns. On the contrary, the topic in a dialogue from advising data may change frequently, but the behavior for information access is more goal-oriented. How much information in the dialogue context should be considered for sentence selection in these two datasets may be different, so effective utilization of the information is challenging and salient. In sum, the challenge covers a wide range of scenarios in real-world applications and serves as a set of benchmark experiments for evaluating dialogue response selection models.

Conversation differs from written articles, because opinion, topic and meaning of some terms may change as the dialogue proceeds. Meanwhile, a conversation comprises utterances spoken by the participants, where each utterance is short and has a clear boundary. Especially, one utterance is often a response to the previous utterance; therefore, we could expect high dependency between every two consecutive utterances in a conversation. Given the above assumption, modeling dependency over utterances may be helpful for understanding conversations; however, the methods for modeling such dependency have not been widely explored. For example, the recently proposed Transformer (Vaswani et al., 2017) achieved improved performance of sequence modeling, but it only considered the information within a single sentence. In order to further enable the cross-sentence attention in the Transformer, this paper proposes the *Highway Recurrent Transformer* to explicitly model not only intra-utterance, but also inter-utterance dependency over the dialogue structure. The intra-utterance dependency is modeled with the Transformer encoder block proposed in Vaswani et al. (2017), while the inter-utterance dependency is modeled by using the proposed highway attention recurrently. Specifically, the highway attention is a modified version of multi-head attention, designed to have the ability to utilize the information from context while preserving the meaning of the current utterance. Experiments show that the Highway Recurrent Transformer model is effective on the response selection task; furthermore, the proposed model can also generalize to other retrieval tasks.

2. Related work

A lot of prior work has used the large-scale Ubuntu Dialog Corpus (Lowe et al., 2015). One category of approaches encodes the given partial conversation and candidate sentences into vectors separately, and then selects the answer by matching the vectors. In this category, LSTM and CNN were applied to encode dialogues and response candidates (Hochreiter and Schmidhuber, 1997; LeCun et al., 1998; Kadlec et al., 2015; Lowe et al., 2015). Moreover, Zhou et al. (2016) used GRU and CNN to form a hierarchical structure for obtaining word-level and conversation-level representations. Another category of approaches focuses on explicitly matching the conversation and the candidates instead of encoding them into vectors. In this category, Wu et al., 2017 matched the GRU-encoded words in the utterances and candidates with an attention mechanism (Bahdanau et al., 2014). Zhang et al. (2018) modeled not only the relation between utterances and responses, but also the relation between utterances and the last utterance in the given partial conversation. Zhou et al., 2018 formed a 3D similarity matrix by stacking the matching matrices between words in the utterances and each candidate, and then 3D convolution is used to calculate the score for each candidate. Among the two categories, only a few works considered the relation between utterances (Zhang et al., 2018; Zhou et al., 2018), but none applied attention for this task. Specifically, Zhang et al. (2018) only modeled the utterance relation to the last one, while Zhou et al., 2018 modeled the relation between utterances implicitly with a convolution operation. In order to leverage inter-utterance relations, our paper focuses on applying attention mechanisms to multiple utterances in a given dialogue, such that our model is capable of modeling both inter-utterance and intra-utterance dependency for better dialogue understanding.

3. Highway recurrent transformer

In the response selection challenge, a partial conversation and a set of response candidates are provided, and the model needs to select one response from the candidates set. The partial conversation consists of l utterances: $U : \{u_1, u_2, \ldots, u_l\}$, an utterance is a sequence of words, and the ith utterance is denoted as $u_i : \{w^U_{i,0}, w^U_{i,1}, w^U_{i,2}, \ldots, w^U_{i,n_i}\}$. Each speaker participating in the conversation is given a special token, say <speaker1>, <speaker2>, and the special token is prepended to the utterances from that speaker. A candidate set consisting of k candidates is denoted as $X : \{x_1, x_2, \ldots, x_k\}$, and each candidate is a sequence of words $x_j : \{w^X_{j,1}, w^X_{j,2}, \ldots, w^X_{j,m_j}\}$. For some datasets, some knowledge-grounded features of a word w are also available, denoted as $F(w)$. Among the candidates, none or some may be the correct responses, and the labels indicating if the candidates are correct answers are denoted as $Y : \{y_1, y_2, \ldots, y_k\}$.

To better model the information in dialogues, we propose the *Highway Recurrent Transformer* to model both intra-utterance and inter-utterance dependency. The model is composed of two main components: highway attention and Transformer (Vaswani et al., 2017). The whole model architecture is illustrated in Fig. 1, where the proposed highway attention is recurrently applied to model cross-utterance information for producing better dialogue embeddings, and then the response can be determined based on the learned embeddings.

3.1. Word feature augmentation

To better understand dialogue content, not only words but also knowledge entities mentioned in the utterances should be considered. Hence, words in the utterances of a conversation and candidates are first converted into their word embeddings, and

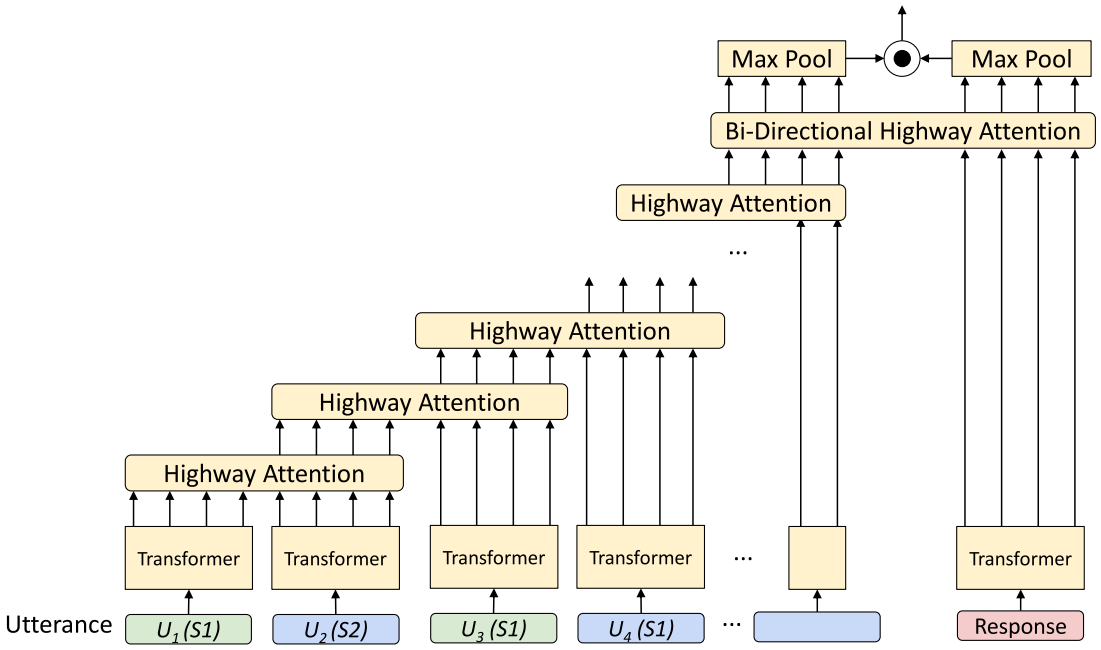

Fig. 1. The illustration of the proposed Highway Recurrent Transformer. In the bottom left part, the blue and green rounded rectangles represent word vectors in the partial conversation U spoken by two different speakers ($S1$ and $S2$).

the embeddings are augmented with some extra knowledge-grounded features, if such features are available. We denote the sequences of words with extra features in the context and candidates as

$$\tilde{U} : \{\tilde{u}_1, \tilde{u}_2, \ldots, \tilde{u}_l\}, \qquad \tilde{X} : \{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_k\},$$

respectively, and each word embedding in the sequences is concatenated with features

$$\tilde{w} = [w; F(w)], \tag{1}$$

where $F(w)$ is a vector of knowledge grounded features depending on the dataset. Specifically, there are some knowledge entities in the dialogues such as "*course name*" in the student-teacher discussions.

3.2. Transformer encoder block

The Transformer encoder block (Fig. 2) proposed in Vaswani et al. (2017) consists of a multi-head attention layer and a position-wise feed-forward network, where a residual connection and layer normalization are used to connect the two components. Details are specified as follows.

3.2.1. Multi-head attention layer

Multi-head attention (Vaswani et al., 2017) (Fig. 2) consists of multiple heads of attention, where each head performs a linear transformation followed by an attention operation. With different sets of trainable parameters, each attention head potentially models different relationships between two sequences. Specifically, the inputs of the multi-head attention layer are three sequences of vectors: *query* $Q \in \mathbb{R}^{l_1 \times d_f}$, *key* $K \in \mathbb{R}^{l_2 \times d_f}$, *value* $V \in \mathbb{R}^{l_2 \times d_f}$, where l_1, l_2 are the length of the first and second sequence respectively. Then for the hth head, three weight matrices $W^{Qh}, W^{Kh}, W^{Vh} \in \mathbb{R}^{d_f \times d_p}$ are used to project the three inputs to a lower dimension d_p, and then an attention function is performed

$$A^h = \text{Attention}(Q^h, K^h, V^h), \tag{2}$$

where $Q^h = QW^{Qh}, K^h = KW^{Kh}, V^h = VW^{Vh}$. The attention function generates a vector for each vector in the query sequence Q. Let the outputs of the attention function be $A^h \in \mathbb{R}^{l_1 \times d_p}$, which is weighted sum of value V based on similarity matrices S^h. For $a = 1, 2, \ldots, l_1$, the ath output is calculated as below:

$$S^h = Q^h (K^h)^T, \tag{3}$$

$$A^h_a = \sum_{p=1}^{l_2} \frac{\exp(s^h_{a,p})}{\sum_{t=1}^{l_2} \exp(s^h_{a,t})} V^h_p, \tag{4}$$

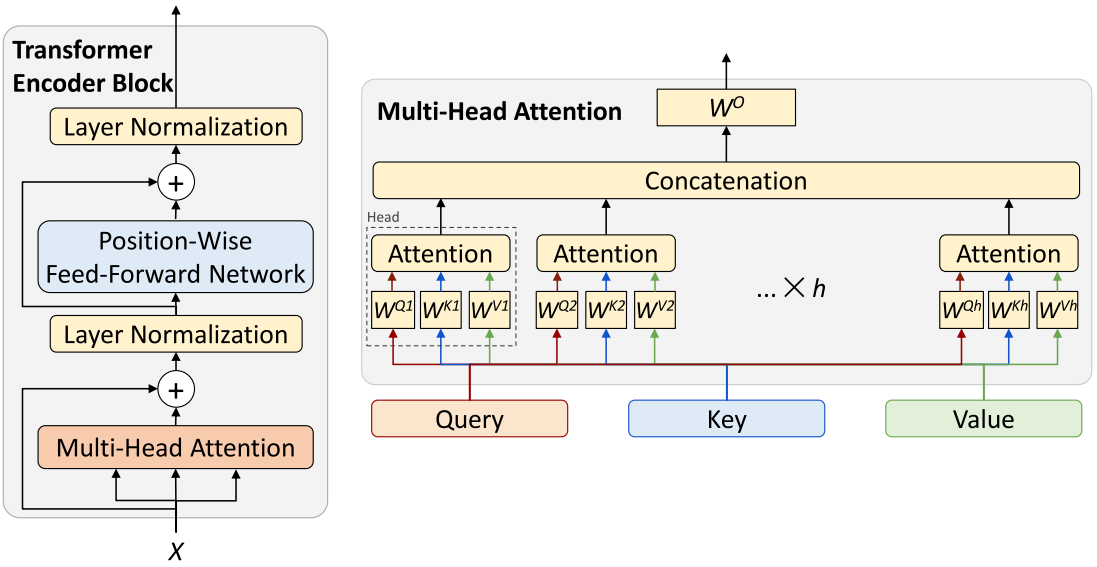

Fig. 2. The Transformer encoding block in the proposed Highway Recurrent Transformer is illustrated in the left part. The input of multi-head attention module includes key, query, and value sequences detailed in the right part; the bottom branches imply that X is fed as the three parameters at the same time.

where s are the similarity scores in the similarity matrix S^h. Then the output of multi-head attention is the linear transformed concatenation of outputs from the attention heads:

$$\text{MultiHead}(Q,K,V) = \text{Concat}(A^1, A^2, \ldots, A^H)W^O \tag{5}$$

where H is the number of heads, and $W^O \in W^{H \cdot d_p \times d_f}$ is a trained weight matrix.

3.2.2. Position-wise feed-Forward network

The position-wise feed-forward network (FFN) transforms each vector in a sequence identically as follows:

$$\text{FFN}(X) = \max(0, XW_1 + b_1)W_2 + b_2 \tag{6}$$

3.2.3. Residual connection and layer normalization

The above two components are connected with a residual connection and layer normalization Ba et al. (2016):

$$\begin{aligned} &\text{ResiNorm}(f,X) = \text{LayerNorm}(X + f(X)), \\ &\text{TransformerBlock}(X) = \text{ResiNorm}(\text{FFN}, \text{ResiNorm}(\text{MultiHead}, (X,X,X))). \end{aligned} \tag{7}$$

Note that here we use the same sequence for the query, key, and value arguments of the mult-head attention, implementing self-attention.

3.3. Highway attention

Motivated by highway networks (Srivastava et al., 2015), we propose a modified version of multi-head attention – highway attention (Fig. 3), in which attention also acts as a highway preserving information from the lower layer. The highway is achieved by performing attention on the query vector itself in addition to the key/value sequence. Specifically, given query $Q \in \mathbb{R}^{l_1 \times d_f}$, key $K \in \mathbb{R}^{l_2 \times d_f}$, and value $V \in \mathbb{R}^{l_2 \times d_f}$, in addition to the linear transformation defined in (2), we also transform the query sequence Q with W^{Kh} and W^{Vh} into additional features:

$$\begin{cases} Q^h = QW^{Qh}, \\ K^h = (K + b_{co})W^{Kh}, \\ V^h = VW^{Vh}, \\ Q^{Kh} = (Q + b_{self})W^{Kh}, \\ Q^{Vh} = QW^{Vh}. \end{cases} \tag{8}$$

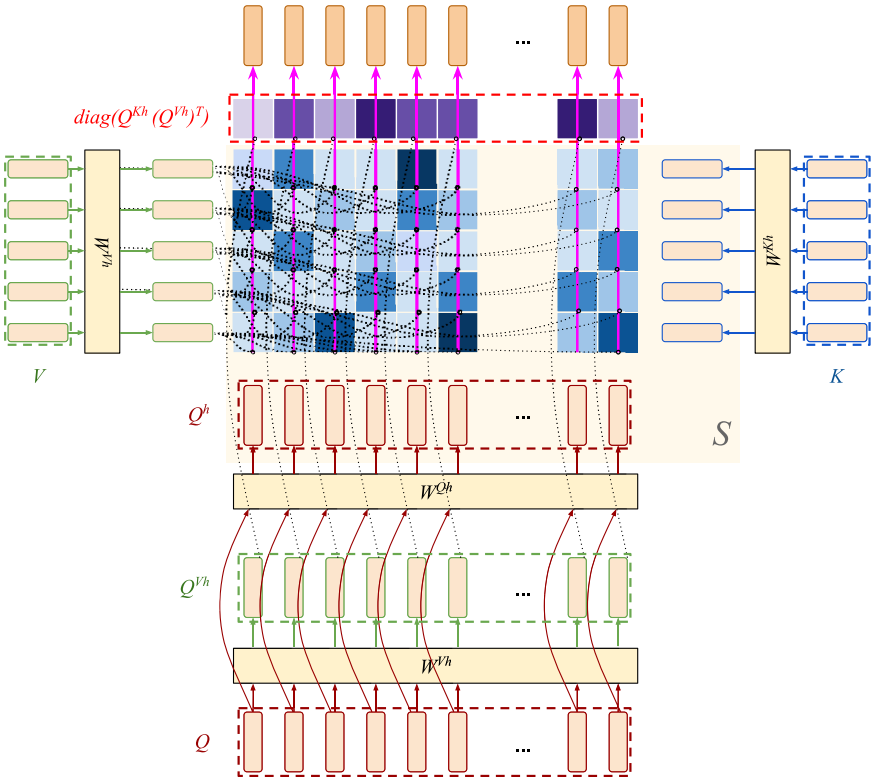

Fig. 3. Highway Attention. The attention values S (blue blocks) are the inner product of Q^h and K^h (rounded rectangles with red border and blue border respectively). The dotted curve from one vector (rounded rectangle) to the attention value (blue block) denotes that the vector is weighted by the value. So the output vectors (top round rectangles with red borders) are the summation of the vectors V, Q^{Vh} (rounded rectangles with green border) weighted by the attention values. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The bias vectors, b_{co} and b_{self}, are added to each vector in the sequences K and Q, respectively. They are added for two reasons:

1. They encode the information of the key's source, which indicates whether the key is from the key sequence K or from the query sequence Q. This information may be crucial, because the calculation of the similarity matrix S is position-independent.
2. The tendency of co-attention or self-attention could be modeled; in other words, it may learn some prior knowledge which is independent of current data samples.

Therefore, the similarity matrices $S \in \mathbb{R}^{l_1 \times l_2}$ and $S_{self} \in \mathbb{R}^{l_1}$ are calculated:

$$\begin{aligned} S^h &= Q^h (K^h)^T, \\ S^h_{self} &= \mathrm{diag}(Q^h (Q^{Kh})^T). \end{aligned} \tag{9}$$

Note that here we only take the diagonal of the self-attention similarity matrix S^h_{self} to measure the tendency to focus on itself for each vector. Hence, (4) can be expanded as

$$A^h_a = \sum_{p=1}^{l_2} \frac{\exp(s^h_{a,p})}{\sum_{t=1}^{l_2} \exp(s^h_{a,t}) + \exp(s^h_{self,a})} V^h_p + \frac{\exp(s^h_{self,a})}{\sum_{t=1}^{l_2} \exp(s^h_{a,t}) + \exp(s^h_{self,a})} Q^{Vh}_a. \tag{10}$$

Namely, A^h_a is the weighted sum of Q^{Vh}_a and vectors in sequence V^h. When S^h_{self} is larger, A^h will retain more information of Q^h. If we treat the attention mechanism as a transformation that transforms vector sequence Q into vector sequence A, then the design here provides a highway to preserve lower layer information. Finally, as in (5), we define the Highway Attention as a linear transformation of the concatenation of the heads:

$$\mathrm{HighwayAttn}(Q, K, V) = \mathrm{Concat}(A^1, \cdots, \mathrm{A^H})\mathrm{W^O}. \tag{11}$$

3.3.1. Recurrence

Each utterance $\tilde{u}_i$ is encoded with a transformer block that shares the parameter of the ones in (7) with the positional encoding:

$$v^U_i = \mathrm{TransformerBlock}(\tilde{u}_i + \mathrm{PE}), \tag{12}$$

where PE represents positional encoding.

Because the current utterances may refer to the instances mentioned in the previous utterance, the highway attention is applied recurrently to route the information from the previous utterances a^U_{i-1} to the current one v^U_i and further infer the attended output a^U_i:

$$a^U_i = \begin{cases} v^U_1 & \text{if } i=1, \\ \text{HighwayAttn}(v^U_i, a^U_{i-1}) & \text{if } i>1. \end{cases} \tag{13}$$

From another perspective, if we view the outputs of the highway attention for a utterance as the "memory" that represents the dialogue state till now, the Highway Attention is hence very similar to the update gate of GRU (Cho et al., 2014). (10) decides how much information is read into the memory, similar to the update gate of GRU.

3.4. Candidate selection

Similarly, each candidate $\tilde{x}_j$ is encoded with a transformer block with the positional encoding:

$$v^X_j = \text{TransformerBlock}(\tilde{x}_j + \text{PE}). \tag{14}$$

The bi-directional highway attention is applied to model the relation between conversations and candidates in two directions.

$$\tilde{v}_j X = \text{HighwayAttn}(v^X_j, a^U_l) \tag{15}$$

$$\tilde{a}^U = \text{HighwayAttn}(a^U_l, v^X_j) \tag{16}$$

where a^U_l represents the most recent output of the highway attention from utterances U. Note that the parameters of the two highway attention blocks are shared.

In another similar model, *Highway Recurrent Transformer-all*, all utterances are considered when applying bidirectional highway attention, so (15) and (16) are replaced with

$$\tilde{v}_j X = \text{HighwayAttn}(v^X_j, [a^U_1; a^U_2; \ldots; a^U_l]), \tag{17}$$

$$\tilde{a}^U = \text{HighwayAttn}([a^U_1; a^U_2; \ldots; a^U_l], v^X_j). \tag{18}$$

Note that this model has the same number of parameters as *Highway Recurrent Transformer-Last* has.

For the submitted system, the attention mechanism is used to condense two sequences into two vectors by a weighted sum over feature sequences:

$$\alpha_k = w^T \tilde{v}_{j,k} X, \quad r^X_j = \sum_k \alpha_k \tilde{v}_{j,k} X, \tag{19}$$

$$\alpha_k = w^T \tilde{a}_k U, \quad r^U = \sum_k \alpha_k \tilde{a}_k U, \tag{20}$$

where w is a trainable weight vector. The score of a candidate x_j is calculated as $s_j = r^X_j \cdot r^U$. However, we afterwards found that max pooling over dimensions is more effective:

$$r^X_j = \max(\tilde{v}_j X), r^U = \max(\tilde{a}^U), \tag{21}$$

therefore the max-pooling method is used in the subsequent experiments in this paper.

We trained the submitted system with the *ranking loss*, which gives an additional penalty if the lowest score among the positive samples' scores is not greater than the highest one among the negative ones' by a margin γ:

$$\text{LSE}(\{s_1, s_2, \ldots, s_k\}) = \log \sum_{i=1}^{k} \exp(s_i), \tag{22}$$

$$L_{rank}(U, X, Y) = \max\{0, \text{LSE}(\{s_j | y_j = 0\}) + \text{LSE}(\{-s_j | y_j = 1\}) + \gamma\}, \tag{23}$$

where S is the set of scores of candidates and LSE(.) is a smooth approximation of the maximum function. Afterwards, we found that ranking loss does not outperform binary cross entropy, so in the experiments in this paper, we utilize the binary cross entropy function as the objective:

$$L(U, X, Y) = \sum_{j=1}^{k} y_j \log\sigma(s_j) + (1 - y_j) \log\Big(1 - \sigma(s_j)\Big) \tag{24}$$

3.5. Ensemble

The results of the hierarchical LSTM and Highway Recurrent Transformer-Last are ensembled by summing the scores of all candidates before applying the sigmoid function.

4. Experiments

To evaluate the proposed Highway Recurrent Transformer, the following experiments are conducted.

4.1. Dataset

DSTC7-Track1 contains two goal-oriented dialogue datasets – (1) Ubuntu data and (2) Advising data. There are five subtasks in this challenge, where this paper focuses on subtask 1, 3 and 4, because the same model architecture can be applied to these subtasks. Here we briefly describe the settings for each subtask:

- Subtask 1: There are 100 response candidates for each dialogue, and only one is correct.
- Subtask 3: There are 100 response candidates for each dialogue. The number of correct responses is between 1 and 5. Multiple correct responses are generated using paraphrases.
- Subtask 4: There are 100 response candidates for each dialogue, and an additional candidate *no answer* should also be considered. The number of correct responses is either 0 or 1.

4.2. Settings

We train and evaluate our model on the Ubuntu and Advising datasets provided by DSTC7 (D'Haro et al., 2020) track 1. Both the datasets are tokenized with Spacy[1] and pre-trained word embeddings from FastText (Bojanowski et al., 2017) are used. For the Advising dataset, in the preprocessing phase, course numbers are normalized to a uniform format (e.g. CS1234n), and we define the domain specific binary features with the provided suggested courses list and the prior taken courses list:

$$F(w) = [F^{prior}(w), F^{suggested}(w)] \tag{25}$$

where $F^{prior}(w)$ and $F^{suggested}(w)$ are 1 if and only if w is a course number and the course is in the prior taken courses list or the suggested courses list respectively. The position-wise feed forward function in the Highway Recurrent Transformer model uses 512 hidden units, and each utterance is encoded by 2 Transformer encoder blocks. The model is trained by sampling negative candidates so the total number of candidates is 10 for each sample. The whole model is optimized with Adam (Kingma and Ba, 2014) with learning rate 0.0001.

4.3. Baseline models

We compare our model with following baseline models:

- Dual LSTM (Lowe et al., 2015): uses two LSTMs to encode the conversation and candidates into two vectors, and selects the candidate based on inner product.
- Hierarchical LSTM: is based on the encoder in HRED (Serban et al., 2016) for encoding the conversations, where one LSTM is used to encode an utterance or a candidate into one vector as its utterance representation, and then the second LSTM encodes the utterance-level representations into a conversation-level representation. Finally, the candidate is selected based on the inner product of its representation and the conversation-level representation.
- Transformer: The utterances are concatenated as a single sequence and then both the sequence and the candidate are encoded by layers of transformer encoder blocks. Then bi-directional highway attention and max pooling are applied and two vectors that represent the conversation and the candidate are obtained. Their inner product is used to for selection.
- Transformer-Last: same as the transformer described above, but only the last utterance is fed into the model.`

4.4. Evaluation metrics

Considering that this work is formulated as a ranking problem, rank-based metrics are performed for evaluation. Let R^* be the rank of the correct candidate (candidate whose label $y = 1$) predicted by the model. Following the official challenge, we compute two widely used metrics below:

- Recall at 10 (Recall@10): Percentage of examples where $R^* \leq 10$.

[1] https://spacy.io/

Table 1
Performance of the different models on the validation/test sets for Ubuntu data.

Ubuntu		Recall@10	MRR	Average
Subtask1	Dual LSTM	62.6/58.7	36.23/35.37	49.39/47.03
	Hierarchical LSTM	65.3/57.5	37.83/34.54	51.56/46.02
	Transformer	64.8/60.6	40.55/36.06	52.68/48.33
	Transformer-Last	62.8/49.1	35.40/28.08	49.10/38.59
	Highway Recurrent Transformer-All	74.24/66.2	46.01/40.40	60.12/53.30
	Highway Recurrent Transformer-Last	74.06/67.0	45.18/40.19	59.62/53.60
	Ensemble	**89.62/67.9**	**66.37/43.52**	**77.99 / 55.71**
Subtask4	Dual LSTM	61.9/69.1	34.72/38.94	48.31/54.02
	Hierarchical LSTM	55.5/57.9	32.73/34.44	44.14/46.17
	Transformer	70.4/75.3	40.77/46.29	55.60/60.79
	Transformer-Last	60.6/62.4	34.35/36.84	47.47/49.62
	Highway Recurrent Transformer-All	71.1/**75.7**	**41.37/46.51**	**56.23/61.11**
	Highway Recurrent Transformer-Last	68.8/73.3	38.56/41.72	53.70/57.51
	Ensemble	**71.2**/72.9	40.89/43.37	56.04/58.13

- Mean Reciprocal Rank (MRR): Average of $\frac{1}{R^n}$.

They measure the quality of a ranking model by checking the position of the correct response in the ranked list.

4.5. Ubuntu results

The performance comparison is shown in Table 1. It is obvious that on the Ubuntu data, the proposed Highway Recurrent Transformer outperforms the Transformer baseline, and the Highway Recurrent Transformer-Last also outperforms the Transformer-Last. Especially for subtask 1, both Highway Recurrent Transformer and Highway Recurrent Transformer-Last significantly outperform the Transformer without recurrent highway attention. For subtask 4, the Highway Recurrent Transformer-Last also obtains comparable performance with the Transformer, but still worse than the Highway Recurrent Transformer-all. Note that the Highway Recurrent Transformer models the relation between conversations and the candidates in the lower layer, and the relation is modeled by the bi-directional highway attention layer at almost the last layer. In other words, our model encodes the partial conversation in vectors independent of the candidates, and the computation cost for scoring one candidate is only the bi-directional highway attention and the inner product. It can be a great advantage over other approaches that model the relation between conversations and candidates in the lower layer. The Highway Recurrent Transformer-Last further reduces the computation cost required to score candidates by considering only the output of the last utterance. Therefore, the performance loss incurred by the Highway Recurrent Transformer-Last can be seen as a trade-off between accuracy and efficiency.

Table 2
Performance of the different models on the validation/test sets for Advising data.

Advising		Recall@10	MRR	Average
Subtask 1	Dual LSTM	62.8/39.8	31.36/15.71	47.14/27.76
	Hierarchical LSTM	64.4/47.8	32.32/23.84	48.42/35.82
	Transformer	71.6/52.4	40.19/25.30	55.90/38.85
	Transformer-Last	70.4/52.6	39.23/25.12	54.81/38.86
	Highway Recurrent Transformer-All	71.6/48.4	39.30/21.43	55.45/34.91
	Highway Recurrent Transformer-Last	69.8/51.6	42.32/25.36	56.07/38.48
	Ensemble	**77.0/54.8**	**39.98/27.89**	**58.49/41.34**
Subtask 3	Dual LSTM	65.0/47.4	43.43/22.41	54.25/34.91
	Hierarchical LSTM	73.4/52.2	52.41/25.73	62.91/38.96
	Transformer	79.2/60.6	55.00/29.89	67.10/45.25
	Transformer-Last	**81.0/61.2**	56.42/31.69	68.71/**46.45**
	Highway Recurrent Transformer-All	80.8/57.2	53.54/27.36	67.17/42.28
	Highway Recurrent Transformer-Last	70.8/54.6	45.52/29.08	58.16/41.84
	Ensemble & Fine Tune	81.0/60.4	**57.15/31.71**	**69.08**/46.06
Subtask 4	Dual LSTM	64.0/43.8	28.19/18.20	46.09/31.00
	Hierarchical LSTM	57.0/40.6	31.25/17.28	44.08/28.94
	Transformer	69.6/52.2	32.62/22.16	51.11/37.18
	Transformer-Last	**68.7/56.8**	**33.92**/24.42	**51.31/40.61**
	Highway Recurrent Transformer-All	60.6/40.4	29.30/16.28	44.95/28.34
	Highway Recurrent Transformer-Last	57.8/39.2	23.54/25.18	40.67/32.19
	Ensemble	64.2/46.4	33.01/**27.09**	48.61/36.75

Table 3
Ablation test for word feature augmentation in the subtask1 of the advising dataset. The numbers are performance on the validation/test set.

Setting	Recall@10	MRR	Average
w/ augmentation	69.8/51.6	42.32/25.36	56.07/38.48
w/o augmentation	47.5/40.6	22.53/17.12	35.01/28.85

4.6. Advising results

In the advising data, the advantage of the recurrent structure is not as significant as in the Ubuntu data. From Table 2, we can also see that for subtask 1 and 3 on advising data, Transformer-all and Transformer-Last obtain similar performance, implying that utterances prior to the last one have little useful information to predict the next one. That may indicate why the recurrent highway attention is not as useful on the advising dataset. Nevertheless, it is surprising that the ensemble of Highway Recurrent Transformer-Last and the hierarchical LSTM leads to significant performance boost compared with either one of the two single models. The results demonstrate that the proposed Highway Recurrent Transformer-Last model may have some complementary advantages the hierarchical LSTM does not have. In sum, the proposed model achieves the improvement for the subtask 1 and comparable performance for the subtasks 3 and 4 with the Transformer model for advising dialogues.

4.7. Ablation study for word feature augmentation

We conduct a ablation study for word feature augmentation in the subtask 1 of the advising dataset. The results in Table 3 shows that if the domain specific features defined in Eq. (25) is not used, then the performance for all metrics can drop significantly. It shows the effectiveness of word feature augmentation.

5. Conclusions

This paper proposes the Highway Recurrent Transformer that effectively models information from multiple levels, including utterance-level and conversation-level, via a highway attention mechanism. The experiments of DSTC7 empirically demonstrate the superior capability of estimating the relation between the dialogue and the response and further selecting the proper response given the dialogue context. Compared with the state-of-the-art transformer models, our proposed model achieves improved performance. The proposed Highway Recurrent Transformer can be investigated for other tasks in the future.

Acknowledgments

This work was financially supported from the Young Scholar Fellowship Program by Ministry of Science and Technology (MOST) in Taiwan, under Grants 108-2636-E-002-003 and 108-2634-F-002-019.

References

Zhou, X., Li, L., Dong, D., Liu, Y., Chen, Y., Zhao, W.X., Yu, D., Wu, H., 2018. Multi-turn response selection for chatbots with deep attention matching network. In: Proceedings of the Fifty-sixth Annual Meeting of the Association for Computational Linguistics, 1. ACL, pp. 1118–1127.
Ba, J. L., Kiros, J. R., Hinton, G. E., 2016. Layer normalization. arXiv preprint arXiv:1607.06450.
Bahdanau, D., Cho, K., Bengio, Y., 2014. Neural machine translation by jointly learning to align and translate. In: Proceedings of 3rd International Conference on Learning Representations.
Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2017. Enriching word vectors with subword information. Transaction of the Association for Computational Linguistics 5, 135–146.
D'Haro, L.F., Yoshino, K., Hori, C., Marks, T.K., Polymenakos, L., Kummerfeld, J.K., Galley, M., Gao, X., 2020. Overview of the seventh Dialog System Technology Challenge: DSTC7. Computer Speech & Language 62, In press.
Gunasekara, C., Kummerfeld, J.K., Polymenakos, .L., Lasecki, W., 2019. Dstc7 task 1: noetic end-to-end response selection. In: Proceedings of the Seventh Edition of the Dialog System Technology Challenges at AAAI 2019.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.
Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. pp. 1724−1734.
Kadlec, R., Schmid, M., Kleindienst, J., 2015. Improved deep learning baselines for Ubuntu corpus dialogs. arXiv preprint arXiv:1510.03753.
Kingma, D. P., Ba, J., 2014. Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980.
Kummerfeld, J. K., Gouravajhala, S. R., Peper, J., Athreya, V., Gunasekara, C., Ganhotra, J., Patel, S. S., Polymenakos, L., Lasecki, W. S., 2018. Analyzing assumptions in conversation disentanglement research through the lens of a new dataset and model. arXiv preprint arXiv:1810.11118.
LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proceedings of the Institute of Electrical and Electronics Engineers 86 (11), 2278–2324.
Lowe, R., Pow, N., Serban, I., Pineau, J., 2015. The Ubuntu dialogue corpus: a large dataset for research in unstructured multi-turn dialogue systems. in Proceedings of the SIGDIAL 2015 Conference, pages 285−294.

Srivastava, R. K., Greff, K., Schmidhuber, J., 2015. Highway networks. arXiv preprint arXiv:1505.00387.
Serban, I.V., Sordoni, A., Bengio, Y., Courville, A.C., Pineau, J., 2016. Building end-to-end dialogue systems using generative hierarchical neural network models.. In: Proceedings of AAAI Conference on Artificial Intelligence, 16. Association for the Advancement of Artificial Intelligence, pp. 3776–3784.
Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. In: Proceedings of Advances in Neural Information Processing Systems, pp. 3104–3112.
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in Neural Information Processing Systems, pp. 5998–6008.
Wu, Y., Wu, W., Xing, C., Zhou, M., Li, Z., 2017. Sequential matching network: a new architecture for multi-turn response selection in retrieval-based chatbots. In: Proceedings of the Fifty-fifth Annual Meeting of the Association for Computational Linguistics, 1. ACL, pp. 496–505.
Zhang, Z., Li, J., Zhu, P., Zhao, H., Liu, G., 2018. Modeling multi-turn conversation with deep utterance aggregation. In: Proceedings of the Twenty-seventh International Conference on Computational Linguistics, pp. 3740–3752.
Zhou, X., Dong, D., Wu, H., Zhao, S., Yu, D., Tian, H., Liu, X., Yan, R., 2016. Multi-view response selection for human-computer conversation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 372–381.

Ting-Rui Chiang is a current graduate student in Department of Computer Science and Information Engineering at National Taiwan University, Taipei, Taiwan. He received B.S. in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan in 2018. His research has been focused on dialogue system, question answering, and machine learning.

Chao-Wei Huang is a current graduate student in Department of Computer Science and Information Engineering at National Taiwan University, Taipei, Taiwan. He received B.S. in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan in 2018. His research has been focused on dialogue systems, spoken language understanding, and speech technologies.

Shang-Yu Su is a current Ph.D. student in Department of Computer Science and Information Engineering at National Taiwan University, Taipei, Taiwan. He holds the bachelor degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan. His research has been focused on deep learning and dialogue systems.

Yun-Nung Chen received Ph.D. in Computer Science from at Carnegie Mellon University, PA. She has been an assistant professor in the Department of Computer Science and Information Engineering at National Taiwan University, Taipei, Taiwan. Her research interests include spoken dialogue understanding, speech summarization, information extraction, and machine learning. Dr. Chen received Google Faculty Research Award and NVIDIA Scientific Research Award and currently serves a member in Speech and Language Technical Committee in IEEE.