Multi-Turn Response Selection in Retrieval Based Chatbots with Hierarchical Residual Matching Network

Zhuo Zhang*, Danyang Zheng, Ping Gong

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

*p@bupt.edu.cn

Abstract: Response selection in retrieval-based chatbot aims to find the most relevant response in a candidate repository given the conversation context. A key technique to this task lies in how to measure the matching degree between conversation context and response at rich semantic information. In this paper, we propose a hierarchical residual matching network (HRMN) to fully extract and make use of the rich semantic information in the conversation history and response for the multi-turn response selection task. We empirically verify HMRN on two benchmark data sets and compare against advanced approaches. Evaluation results demonstrate that HMRN outperforms strong baselines and has a distinct improvement in response selection.

Keywords: Multi-turn response selection, hierarchical semantic information, residual matching.

1. Introduction

The rapid development of artificial intelligence makes a fundamental breakthrough in the construction of chatbots[1]. Existing chatbots are either generation-based or retrieval-based. The generation-based chatbot could generate highly coherent new responses given the conversation context[2-4]. On the contrary, retrieval-based chatbot tries to find the most relevant response in a candidate responses repository with some conversation context given[5, 6]. In this paper, we focus on the problem of response selection for retrieval-based chatbots since it has the advantage of informative responses.

Recent studies have shown that capturing and making full use of the richer semantic information from the conversation history is essential to facilitate selecting the next utterance[6-9]. A deep neural network is one of the ways to discover richer and more useful semantic information. But it has two shortcomings: (1) The deeper the level, the more abstract the semantic information captured where matching on high abstract context vectors loses relationships among utterances[6]. (2) It is a recognized fact that deep networks are difficult to train. This is boiled down to not only disappear/explode gradient but also difficulty pertaining to feature propagation.

To cope with these issues, this paper proposes a hierarchical residual matching network (HMRN) for multi-turn response selection. The HMRN distills hierarchical semantic information in the multiple
semantic encoder layer and calculates the matching score based on rich semantic information. To alleviate the inherent issue of a deep network, we devise two different residual techniques to ameliorate gradient flow and exhaustively leverage semantic information across all hierarchies. Examining on extensive experiments, HRMN achieves a significant improvement compared to other baselines.

2. Model

Firstly, we formulate the response selection problem. Suppose that we have a dataset \( D = \{C, R, Y\} \). Specifically, \( C = \{U_1, U_2, \ldots, U_n\} \) represents the conversation context with \( \{U_k\}_{k=1}^n \) as the utterances and \( R \) as a response candidate. \( Y \in \{0, 1\} \) is a binary label. The goal of the selection task is to learn the matching models from the conversation data set \( D \), which reflects the matching degree between \( C \) and \( R \). To this end, we devise a well-designed hierarchical residual matching network which consists of multiple semantic encoder layer, semantic matching layer, residual layer, and aggregation prediction layers.

Multiple semantic encoder layer: our strong intuition is that the dialogue context contains manifold hierarchical semantic information, which is beneficial to the choice of response. Therefore, we design multiple encoders to exhaustively distill multifarious semantic information from the dialogue context. As shown in Fig. 1, multiple semantic encoder layer stacks Embedding sub-layer, GRU-based sub-layer, CNN-based sub-layer, and Transformer-based sub-layer and different sub-layer help model extract diverse hierarchical semantic information. More concretely, embedding sub-layer, jointing pre-training word embedding, and character embedding, represents the original information. The GRU-based sub-layer captures context information. The CNN-based sub-layer promotes the model to extract higher-level semantic units instead of simple semantic information[10]. Notably, we perform the multi-scale convolution which has four different convolution kernel sizes (1, 2, 3, 4). Following [9], we utilize a Transformer-based sub-layer to fetch latent semantic information such as co-reference, which is beneficial to match utterance and response in high-level semantic and is also verified by experiments.

![Fig. 1 The architecture of multiple semantic encoder layer](image)

Semantic matching layer: Our model matches response and multi-turn context under the matching-aggregation framework. The interaction between the conversational context and response provides important information to determine the matching degree. Therefore, we employ a bi-directional
attention flow[11] as an interaction function for the aforementioned hierarchical semantic information. Specifically, for the hierarchical semantic information obtained after multiple semantic encoded layers: \( \{U_{enc}^{k,t}\}_{t=1}^{l_u} \) and \( \{R_{enc}^{k,t}\}_{t=1}^{l_r} \) where \( enc \in \{\text{emb, gru, cnn, trans}\} \), we have:

\[
e_{ij}^{enc} = (u_i^{enc})^T r_j^{enc} \quad (1)
\]

\[
\bar{u}_i^{enc} = \sum_{j=1}^{l_u} \frac{\exp(e_{ij})}{\sum_{m=1}^{l_u} \exp(e_{im})}, i \in \{1,...,l_u\} \quad (2)
\]

\[
\bar{r}_i^{enc} = \sum_{j=1}^{l_r} \frac{\exp(e_{ij})}{\sum_{m=1}^{l_r} \exp(e_{im})}, j \in \{1,...,l_r\} \quad (3)
\]

\( A \) is a linear transforming matrix, and \( e_{ij}^{enc} \) is the interaction score. Moreover, we use the element-wise multiplication and element-wise subtraction operations[12] to obtain semantic interaction matrix of the conversational context and response and:

\[
E_{enc}^k = \text{FFN}(u_i^{enc} - \bar{u}_i^{enc} \odot \bar{r}_i^{enc}; r_i^{enc} - \bar{r}_i^{enc} \odot \bar{r}_i^{enc}) \quad (4)
\]

Particularly, we use \( \text{SemMatch}(u,r) \) to denote all operations of the semantic matching layer.

Residual layer: The purpose of the residual layer is not only to improve the gradient flow to better train the deep neural network but also to make the model comprehensively matches on residual semantic information. To this end, we design two residual techniques including concatenate residual and cross residual. As shown in Fig. 2, we splice the across layer semantic information of utterances and responses into two groups. One group is \( U_{emb}^k, U_{cnn}^k \) and \( R_{emb}^k, R_{cnn}^k \), another group is \( U_{gru}^k, U_{trans}^k \) and \( R_{gru}^k, R_{trans}^k \). As demonstrated in Fig. 2 (a), the concatenate residual technique acquires a concatenate residual interaction matrix:

\[
E_{emb-cnn}^{k,concat} = \text{SemMatch}([U_{emb}^k; U_{cnn}^k],[R_{emb}^k; R_{cnn}^k]),
\]

and we can obtain another residual interaction matrix:

\[
E_{gru-trans}^{k,concat} = \text{SemMatch}([U_{gru}^k; U_{trans}^k],[R_{gru}^k; R_{trans}^k]).
\]

Different from the concatenate residual technique, the cross residual technique performs matching operation directly on the cross-layer semantic information, which means that \( U_{emb}^k \) matches \( R_{cnn}^k \) and \( U_{cnn}^k \) matches \( R_{emb}^k \), \( U_{gru}^k \) matches \( R_{trans}^k \) and \( U_{trans}^k \) matches \( R_{gru}^k \). Recall the interaction score and we extend this to the maximum operation:

\[
e_{ij}^{cross} = \max \{ \Sigma_{enc_p} \Sigma_{enc_q} (u_i^{enc_p})^T r_j^{enc_q} \}
\]

\( enc_p \) and \( enc_q \) represent different types of semantic information separately but cannot be the same at the same time. The intuition behind the maximum operation is that the correlation with a high
matching score should still be high in different semantic matches. Replacing eq.(1) with eq.(5) and performing SemMatch into two groups cross semantic information, we can acquire the cross residual interaction matrix $E_{k,cross}^{emb-cnn}$ and $E_{k,cross}^{gru-trans}$.

![Diagram](https://via.placeholder.com/150)

**Fig. 2** The structure of two residual techniques.

(a) represents the concatenate residual technique while (b) shows the cross residual technique.

Aggregation prediction layer: Following [6], we use GRU as an aggregation method and the interaction matrix is boiled down to a matching vector. Assuming that the semantic interaction sequence is $\{E_k^{enc}\}^u_{k=1}$, where $enc \in \{emb, gru, cnn, trans, residual\}$. Note that $enc = \text{residual}$ represents $\text{emb-cnn-gru-att}$ obtained by concatenate residual techniques or cross residual technique. The GRU integrates the matching information in $E_k^{enc} = \{e_k^{enc}\}^u_{i=1}$, we have $h_k^{enc} = GRU(e_k^{enc}, h_{k,i-1}^{enc}), \forall i \in \{1,2,...,u\}$. We choose the last hidden state $h_k^{enc}$ to represent the k-th interaction matrix. Then, we employ another GRU to aggregate the matching vectors of the context temporal relationship: $f_k^{enc} = GRU(h_k^{enc}, f_{k-1}^{enc}), \forall k \in \{1,2,...,u\}$. Finally, the sub-matching score $g_k^{enc}$ is obtained by feeding different semantic matching vector to the multi-layer perceptron (MLP): $g_k^{enc} = MLP(f_k^{enc})$. The final match score is $g = \sum g_k^{enc}$.

3. Experiment

We conduct experiments on two well-studied multi-turn response selection datasets, the Ubuntu Corpus V1[13] and the Douban Conversation Corpus[6]. The Ubuntu Corpus V1 is a specific domain data set, which contains multi-turn dialogues about Ubuntu system troubleshooting in English. While the Douban Conversation Corpus is a multi-turn Chinese conversation data set crawled from Douban group. Next, we briefly introduce the evaluation metric, along with several state-of-the-art approaches that we compare against. Following the previous studies[6, 13], we choose the information retrieval metrics including Recall at position k in n candidates (Rn@k), Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Precision at 1 (P@1). The competitor baselines could be roughly divided into three categories: Based on basic methods include TF-IDF, LSTM[11], CNN[14]. Based on single-turn matching methods. These methods concatenated the context utterances to get ethical to match a
response including MV-LSTM[15], Match-LSTM[16], Attentive-LSTM[17]. Based on multi-turn matching methods. These methods make use of conversation history in matching including Multi-View[8], SMN[6], DUA[7], DAM[9].

Table 1 reports the evaluation results of HRMN as well as previous models. Overall, HRMN, either with concatenate residual technique or with cross residual technique, outperforms the other models on all metrics and datasets. This exhibits that our model has the ability to select the best-matched response. The three categories of baselines show a consistent trend on both data sets over all metrics: Basic methods < Single-turn matching methods < Multi-turn matching methods. This is a further demonstration that considering not only matching degree between responses and utterances in previous turns but also utterance relationships can improve the performance of the response selection. The state-of-the-art matching model, DAM, performs worse than our models. The reason is that albeit the DAM employs semantic information of all stacked layers for matching, the naïve fusion of semantic information drops the measurement of matching degree, which not fully utilizes the semantic information. This somehow indicates that making full use of the richer semantic information is essential to facilitate selecting the next utterance. One notable point is that the best performance for Ubuntu corpora is HRMN with cross residual technique, while on Douan corpora, the best performance is HRMN with concatenate residual technique. The difference may stem from the nature of the two data, which have different domains (system troubleshooting and social network). Furthermore, the HRMN with two residual technique has a steep improvement on different corpora, which also proves that its compatibility across domains.

|                 | UBUNTU CORPUS | DOUBAN CONVERSATION CORPUS |
|----------------|--------------|-----------------------------|
|                | R2@1 | R10@1 | R10@2 | R10@5 | MAP  | MRR  | P@1 | R10@1 | R10@2 | R10@5 |
| TF-IDF         | 0.659 | 0.410 | 0.545 | 0.708 | 0.331 | 0.359 | 0.180 | 0.096 | 0.172 | 0.405 |
| CNN            | 0.848 | 0.549 | 0.684 | 0.896 | 0.417 | 0.440 | 0.226 | 0.121 | 0.252 | 0.647 |
| LSTM           | 0.901 | 0.638 | 0.784 | 0.949 | 0.485 | 0.527 | 0.320 | 0.187 | 0.343 | 0.720 |
| MV-LSTM        | 0.906 | 0.653 | 0.804 | 0.946 | 0.498 | 0.538 | 0.348 | 0.202 | 0.351 | 0.710 |
| MATCH-LSTM     | 0.904 | 0.653 | 0.799 | 0.944 | 0.500 | 0.537 | 0.345 | 0.202 | 0.348 | 0.720 |
| ATTENTIVE-LSTM | 0.903 | 0.633 | 0.789 | 0.943 | 0.495 | 0.523 | 0.331 | 0.192 | 0.328 | 0.718 |
| SMN            | 0.926 | 0.726 | 0.847 | 0.961 | 0.529 | 0.569 | 0.397 | 0.233 | 0.396 | 0.724 |
| DUA            | -    | 0.752 | 0.868 | 0.962 | 0.551 | 0.599 | 0.421 | 0.243 | 0.421 | 0.780 |
| DAM            | 0.938 | 0.767 | 0.874 | 0.969 | 0.550 | 0.601 | 0.427 | 0.254 | 0.410 | 0.757 |
| HRMN WITH CONCAT | 0.943 | 0.784 | 0.885 | 0.973 | 0.563 | 0.607 | 0.434 | 0.260 | 0.434 | 0.785 |
| HRMN WITH CROSS | **0.944** | **0.787** | **0.886** | **0.973** | **0.563** | **0.607** | **0.434** | **0.260** | **0.434** | **0.785** |

**Tab. 1** Experimental results of HRMN and other comparison models on Ubuntu and Douan Conversation Corpus. The best method in each setting is in bold.

Moreover, we use the Ubuntu Corpus for further analyzing how HRMN works from the impact of multiple encoders as well as the effect of the residual technique. In addition to the efficacy of multiple encoders, we are also curious about two questions: (1) if the multiple encoder layer in HRMN can use...
only one type of encoder such as RNN, CNN, Transformer instead of multiple encoders. (2) the effect of each encoder in multiple encoder layer. Therefore, we respectively utilize only one type of encoder in the multi-encoder layer and compare them with HRMN, denoting the model as HRMN with \{X\}, where X ∈ \{GRU, CNN, Transformer\}. At the same time, the concatenate residual technique is employed. We then remove the encoder layer by layer for HRMN to show the effect of each encoder and denote the models as HRMN−\{X\}. The results are shown in Table 2. For question (1), we can find that using only one type of encoder results in performance degradation, which implies that multiple encoders enjoy the advantage of capturing various semantic information. For question (2), we can conclude that each encoder is useful, and removing one lead to a performance drop.

| UBUNTU CORPUS | R10@1 | R10@2 | R10@5 | R2@1 |
|---------------|-------|-------|-------|------|
| HRMN WITH GRU | 0.940 | 0.775 | 0.877 | 0.966 |
| HRMN WITH CNN | 0.938 | 0.770 | 0.875 | 0.968 |
| HRMN WITH TRANSFORMER | 0.940 | 0.779 | 0.881 | 0.969 |
| HRMN | **0.944** | **0.787** | **0.886** | **0.973** |
| HRMN-TRANSFORMER | 0.941 | 0.779 | 0.880 | 0.971 |
| HRMN-CNN | 0.936 | 0.770 | 0.886 | 0.961 |
| HRMN-GRU | 0.934 | 0.766 | 0.857 | 0.958 |

Tab.2 Performances of the multiple encoders on the Ubuntu Corpus

We next study the effect of residual technique by designing two experiments. In the first experiment, we only remove the residual technique but remain the multi-hierarchies matching and denote HMN, while in the second we only match a response with the utterances at the topmost semantic information, namely, the output of latent relationship encoder sub-layer, and denote MN. As demonstrated in Table 3, residual semantic information is useful to further improve the performance of the model compared with HRMN and HMN. We also find matching at the topmost abstract semantic information results in dramatic performance degradation, which suggests the importance of residuals and further demonstrates the effect of multi-hierarchies matching.

| UBUNTU CORPUS | R2@1 | R10@1 | R10@2 | R10@5 |
|---------------|------|-------|-------|-------|
| MN | 0.934 | 0.760 | 0.851 | 0.959 |
| HMN | 0.942 | 0.781 | 0.882 | 0.971 |
| HRMN | **0.944** | **0.787** | **0.886** | **0.973** |

Tab. 3 Results of residual technique’s effect on Ubuntu Corpus

4. Conclusion
In this paper, we propose the hierarchical residual matching network (HRMN) for the multi-turn response selection task. Our model sheds new insights on how to make full use of various semantic information for measuring the matching degree between conversation context and response. Meanwhile, we present two kinds of residual techniques to overcome the defects of the deep networks. To our knowledge, there is no prior work using these residual techniques in the multi-turn conversation model. Experiments on two well-studied datasets demonstrate our proposed model significantly outperforms the baseline models by a large margin on all metrics.

References

[1] Gao J, Galley M and Li L, et al 2019 Neural approaches to conversational ai Foundations and Trends® in Information Retrieval 13(2-3) 127–298
[2] Mou L, Song Y, Yan R, Li G, Zhang L and Jin Z 2016 Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation arXiv preprint arXiv:1607.00970
[3] Serban I V, Sordoni A, Bengio Y, Courville A and Pineau J 2016 Building end-to-end dialogue systems using generative hierarchical neural network models In: Thirtieth AAAI Conference on Artificial Intelligence
[4] Vinyals O and Le Q 2015 A neural conversational model arXiv preprint arXiv:1506.05869
[5] Ji Z, Lu Z and Li H 2014 An information retrieval approach to short text conversation arXiv preprint arXiv:1408.6988
[6] Wu Y, Wu W, Xing C, Zhou M and Li Z 2016 Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots arXiv preprint arXiv:1612.01627
[7] Zhang Z, Li J, Zhu P, Zhao H and Liu G 2018 Modeling multi-turn conversation with deep utterance aggregation arXiv preprint arXiv:1806.09102
[8] Zhou X, Dong D, Wu H, Zhao S, Yu D, Tian H, Liu X and Yan R 2016 Multi-view response selection for human-computer conversation Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing pp 372–381
[9] Zhou X, Li L, Dong D, Liu Y, Chen Y, Zhao WX, Yu D and Wu H 2018 Multiturn response selection for chatbots with deep attention matching network Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics Vol 1: Long Papers pp 1118–1127
[10] Lin J, Su Q, Yang P, Ma S and Sun X 2018 Semantic-unit-based dilated convolution for multi-label text classification arXiv preprint arXiv:1808.08561
[11] Seo M, Kembhavi A, Farhadi A and Hajishirzi H 2016 Bidirectional attention flow for machine comprehension arXiv preprint arXiv:1611.01603
[12] Chen Q, Zhu X, Ling Z, Wei S, Jiang H and Inkpen D 2016 Enhanced lstm for natural language inference arXiv preprint arXiv:1609.06038
[13] Lowe R, Pow N, Serban I and Pineau J 2015 The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. arXiv preprint arXiv:1506.08909
[14] Kadlec R, Schmid M and Kleindienst J 2015 Improved deep learning baselines for ubuntu corpus dialogs arXiv preprint arXiv:1510.03753
[15] Wan S, Lan Y, Xu J, Guo J, Pang L and Cheng X: Match-srnn 2016 Modeling the recursive matching structure with spatial rnn arXiv preprint arXiv:1604.04378

[16] Wang S and Jiang J 2015 Learning natural language inference with lstm arXiv preprint arXiv:1512.08849

[17] Tan M, Santos Cd, Xiang B and Zhou B 2015 Lstm-based deep learning models for non-factoid answer selection arXiv preprint arXiv:1511.04108