An Ensemble of Retrieval-Based and Generation-Based Human-Computer Conversation Systems

Yiping Song,[1] **Cheng-Te Li,**[2] **Jian-Yun Nie,**[3] **Ming Zhang,**[1*] **Dongyan Zhao,**[4] **Rui Yan**[4*]
[1]Institute of Network Computing and Information Systems, School of EECS, Peking University, China
[2]Department of Statistics, National Cheng Kung University, Taiwan
[3]University of Montreal, Canada
[4]Institute of Computer Science and Technology, Peking University, China
{songyiping, mzhang_cs, zhaody, ruiyan}@pku.edu.cn chengte@mail.ncku.edu.tw nie@iro.umontreal.ca

Abstract

Human-computer conversation systems have attracted much attention in Natural Language Processing. Conversation systems can be roughly divided into two categories: retrieval-based and generation-based systems. Retrieval systems search a user-issued utterance (namely a *query*) in a large conversational repository and return a reply that best matches the query. Generative approaches synthesize new replies. Both ways have certain advantages but suffer from their own disadvantages. We propose a novel ensemble of retrieval-based and generation-based conversation system. The retrieved candidates, in addition to the original query, are fed to a reply generator via a neural network, so that the model is aware of more information. The generated reply together with the retrieved ones then participates in a re-ranking process to find the final reply to output. Experimental results show that such an ensemble system outperforms each single module by a large margin.

1 Introduction

Automatic human-computer conversation systems have long served humans. Recently, researchers have paid increasing attention to open-domain, chatbot-style human-computer conversations such as XiaoIce[1] and Duer[2] due to their commercial values. For open-domain conversations, rules and templates-based methods, which have been widely used in specific-domain conversation systems, would probably fail since they hardly can handle the great diversity of conversation topics and flexible representations of natural language sentences. With the increasing popularity of online social media and community question-answering platforms, a huge number of human-human conversation utterances are available on the public Web ([Yan *et al.*, 2016a; Li *et al.*, 2016b]). Previous studies begin to develop data-oriented approaches, which can be roughly categorized into two groups: retrieval systems and generative systems.

When a user issues an utterance (called a *query*), the retrieval-based conversation systems search a corresponding utterance (called a *reply*) that best matches the query in a pre-constructed conversational repository ([Isbell *et al.*, 2000; Ji *et al.*, 2014]). Owing to the abundant web resources, the retrieval mechanism will always find a candidate reply given a query using semantic matching. The retrieved replies usually have various expressions with rich information. However, the retrieved replies are limited by the capacity of the pre-constructed repository. Even the best-matched reply from the conversational repository is not guaranteed to be a good response since most cases are not tailored for the issued query.

To make a reply tailored appropriately for the query, a better way is to generate a new one accordingly. With the prosperity of neural networks powered by deep learning, generation-based conversation systems are developing fast. Generation-based conversation systems can synthesize a new sentence as the reply, and thus bring the results of good flexibility and quality. A typical generation-based conversation model is `seq2seq` ([Sordoni *et al.*, 2015; Shang *et al.*, 2015; Serban *et al.*, 2016a]), in which two recurrent neural networks (RNNs) are used as the encoder and the decoder. The encoder is to capture the semantics of the query with one or a few distributed and real-valued vectors (also known as *embeddings*); the decoder aims at decoding the query embeddings to a reply. Long short term memory (LSTM) ([Hochreiter and Schmidhuber, 1997]) or gated recurrent units (GRUs) ([Cho *et al.*, 2014]) could further enhance the RNNs to model longer sentences. The advantage of generation-based conversation systems is that they can produce flexible and tailored replies. A well-known problem for the generation conversation systems based on "`Seq2Seq`" is that they are prone to choose universal and common generations. These generated replies such as "I don't know" and "Me too" suit many queries ([Serban *et al.*, 2016a]), but they contain insufficient semantics and information. Such insufficiency leads to non-informative conversations in real applications.

*Corresponding authors
[1]http://www.msxiaoice.com/
[2]http://duer.baidu.com/

Category	Pros	Cons
Retrieval	literal human utterances; various expressions with great diversity	not tailored to queries; bottleneck is the size of repository
Generation	tailored for queries; highly coherent	insufficient information; universal sentences

Table 1: Characteristics of retrieved and generated replies in two different conversational systems.

Previously, the retrieval-based and generation-based systems with their own characteristics, as listed in Table 1, have been developed separately. We are seeking to absorb their merits. In this paper, we propose an ensemble of retrieval-based and generation-based conversation systems. Specifically, given a query, we first apply the retrieval module to search for k candidate replies. We then propose a "multi sequence to sequence" (`multi-seq2seq`) model to integrate retrieved replies into the `Seq2Seq` generation process so as to enrich the meaning of generated replies to respond the query. We generate a reply via the `multi-seq2seq` generator based on the query and k retrieved replies. Afterwards, we construct a re-ranker to re-evaluate the retrieved replies and the newly generated reply so that more meaningful replies with abundant information would stand out. The highest ranked candidate (either retrieved or generated) is returned to the user as the final reply.

Experimental results show that our ensemble system consistently outperforms each single component in terms of subjective and objective metrics, and both retrieval-based and generation-based methods contribute to the overall approach. This also confirms the rationale for building model ensembles for conversation systems.

2 Related Work

Most of the free chatting commercial products choose to use retrieval-based methods to establish the conversation systems. Isbell *et al.* (2000) apply information retrieval techniques to search for related queries and replies. Ji *et al.* (2014) and Yan *et al.* (2016a) use both shallow hand-crafted features and deep neural networks for *query-reply* matching. Li *et al.* (2016b) propose a random walk-style algorithm to rank candidate replies. In addition, their model can incorporate additional content (related entities in the conversation context) by searching a knowledge base when a stalemate occurs during human-computer conversations.

Generative conversation systems have attracted increasing attention in the NLP community. Ritter *et al.* (2011) formulate query-reply transformation as a phrase-based machine translation. Zoph and Knight (2016) use two RNNs in encoder and one RNN in decoder to translate a sentence into two different languages into another language. Lately, the renewed prosperity of neural networks witnesses an emerging trend in using RNN for conversation systems ([Sutskever *et al.*, 2014; Vinyals and Le, 2015; Sordoni *et al.*, 2015; Shang *et al.*, 2015; Serban *et al.*, 2016a]). The prevalent structure is the `seq2seq` model ([Sutskever *et al.*, 2014]) which comprises of one encoder and one decoder. However, a known issue with RNN is that it prefers to generate short and meaningless utterances. Li *et al.* (2016a) propose a mutual information objective in contrast to the conventional maximum likelihood criterion. Mou *et al.* (2016) and Xing *et al.* (2016) introduce additional content (i.e., either the most mutually informative word or the topic information) to the reply generator. Serban *et al.* (2016b) applies a variational encoder to capture query information as a distribution, from which a random vector is sampled for reply generation. He *et al.* (2017) uses knowledge base for answer generation in question answering task and Libovicky and Helcl (2017) investigates different attention strategies in multi-source generation.

3 Model Ensemble

3.1 Overview

Figure 1 depicts the overview of our proposed framework, which consists of the following components.

• **Retrieval Module.** We have a pre-constructed repository consisting millions of query-reply pairs $\langle q^*, r^* \rangle$, collected from human conversations. When a user sends a query utterance q, our approach utilizes a state-of-the-practice information retrieval system to search for k best matched queries (q^*), and return their associated replies r^* as k candidates.

• **Generation Module.** We propose the `multi-seq2seq` model, which takes the original query q and k retrieved candidate replies $r_1^*, r_2^*, \ldots, r_k^*$ as input, and generates a new reply r^+. Thus the generation process could not only consider the given query, but also take the advantage of the useful information from the retrieved replies. We call it the **first ensemble** of the retrieval method and generation method.

• **Re-ranker.** Finally, we develop a re-ranker to select the best reply r from the $k+1$ candidates obtained from retrieval-based and generation-based modules. Through the ensemble of retrieval-based and generation-based conversation, the enlarged candidate set enhances the quality of the final result. We call this procedure the **second ensemble**.

3.2 Retrieval-Based Conversation System

The information retrieval-based conversation is based on the assumption that the appropriate reply to the user's query is contained by the pre-constructed conversation datasets. We collect huge amounts of conversational corpora from on-line chatting platforms, whose details will be described in the section of evaluation. Each utterance and its corresponding reply form a pair, denoted as $\langle q^*, r^* \rangle$.

Based on the pre-constructed dataset, the retrieval process can be performed using an the state-of-the-practice information retrieval system. We use a `Lucene` [3] powered system for the retrieval implementation. We construct the inverted indexes for all the conversational pairs at the off-line stages. When a query q is issued, keywords extracted from q and their *tf.idf* values are formulated as the retrieval schema and feed into the retrieval system to search the most relevant q^* in database. Then, the associated r^* of q^* will be returned as the output, resulting in an indirect matching between the

[3] http://lucene.apache.org

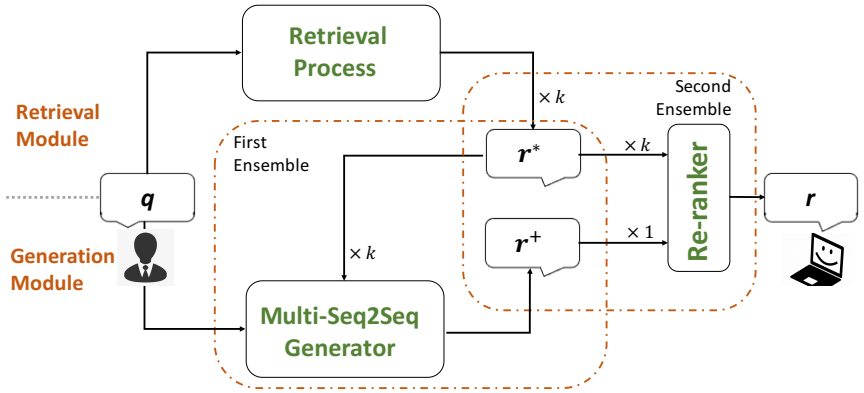

Figure 1: The overall architecture of our model ensemble. We combine the retrieval-based and generation-based conversation systems with two mechanisms. The first ensemble is to enhance the generator with the retrieved candidates. The second is the re-ranking of both candidates.

user's query q and the retrieved reply r^*. The retrieval systems would provide more than one replies and score them according to the semantic matching degree, which is a traditional technic in information retrieval. As the top-ranked one may not perfectly match the query, we keep the top-k replies for further process.

The information retrieval is a relatively mature technique, so the retrieval framework can be alternated by any systems built keep to the above principles.

3.3 Generation-Based Conversation System

A generation-based conversation system is able to synthesize new utterances, which is complementary to retrieval-based methods. The `seq2seq` model ([Sutskever *et al.*, 2014]) , considering the Recurrent Neural Network (RNNs) as the encoder and decoder to transfer source sentence to target sentence, has long been used for generation tasks. The objective function of the `seq2seq` model in our scenario is the log-likelihood of the generated reply r^+ given the query q. Since the reply is generated on the conditional probabilities given the query, the universal replies which have relatively higher probabilities achieve higher rankings. However, these universal sentences contain less information, which impairs the performance of generative systems. Mou *et al.* (2016) also observe that in open-domain conversation systems, if the query does not carry sufficient information, `seq2seq` tends to generate short and meaningless sentences.

Different from the pipeline in `seq2seq` model, we propose the `multi-seq2seq` model (Figure 2), which synthesizes a tailored reply r^+ by using the information both from the query q and the retrieved $r_1^*, r_2^*, \ldots, r_k^*$. `multi-seq2seq` employs $k+1$ encoders, one for query and other k for retrieved r^*. The decoder receives the outputs of all encoders, and remains the same with traditional `seq2seq` for sentence generation. `multi-seq2seq` model improves the quality of the generated reply in two ways. First, the newly generated reply conditions not only on the given query but also the retrieved reply. So the probability of universal replies would decrease since we add an additional condition. The objective function can be written as:

$$\hat{r^+} = \underset{r^+}{\operatorname{argmax}} \{\log p(r^+|q, r_1^*, r_2^*, \ldots, r_k^*)\} \quad (1)$$

Thus the r^+ would achieve higher score only if it has a high concurrency with both q and $r_1^*, r_2^*, \ldots, r_k^*$. Second, the retrieved replies $r_1^*, r_2^*, \ldots, r_k^*$ are the human-produced utterances and probably contain more information, which could be used as the additional information for the generated reply r^+. Hence, the generated reply can be fluent and tailored to the query, and be more meaningful due to the information from the retrieved candidates. To take advantage of retrieved replies, we propose to integrate attention and copy mechanisms into decoding process. Attention helps the decoder to decide which parts of each retrieved reply are useful for current generation step. Copy mechanism directly extracts proper words from encoders, namely both query and retrieved replies, and utilizes them as the output words during the decoding process.

- **Two-level Attention.** `multi-seq2seq` conducts sentence- and character- level attention to make better use of the query and retrieved replies. As multiple replies are of uneven quality, we use sentence-level attention to assign different importance of each retrieved replies. Similarly, multiple words are of uneven quality in a sentence, we use character-level attention to measure different importance to each word in retrieved replies. Specifically, for the sentence-level, we use $k+1$ vectors obtained from the encoders to capture the information of q and the k r^*, denoted as $\boldsymbol{q}$ and $\boldsymbol{r}_1^* \ldots, \boldsymbol{r}_k^*$, which are concatenated as $[\boldsymbol{q}; \boldsymbol{r}^*; \ldots; \boldsymbol{r}_k^*]$. This vector is linearly transformed before fed to the decoder as the initial state. For the character-level, we extend the traditional attention ([Bahdanau *et al.*, 2015]) to multi-source attention to introduce retrieved replies, given by

$$c_i = \sum_{j=1}^{l} \alpha_{i,j} \boldsymbol{h}_j + \sum_{m=1}^{k} \sum_{j=1}^{l_m} \alpha_{i,m,j} \boldsymbol{h}_{m,j} \quad (2)$$

$$\alpha_{i,m,j} = \frac{\exp e_{i,m,j}}{\sum_{j=1}^{l_m} e_{i,m,j}}, e_{i,m,j} = \tanh(\boldsymbol{s}_{i-1} M_a \boldsymbol{h}_{m,j}) \quad (3)$$

where c_i is the context vector at each time step in decoding, which integrates query and all possible words in k retrieved

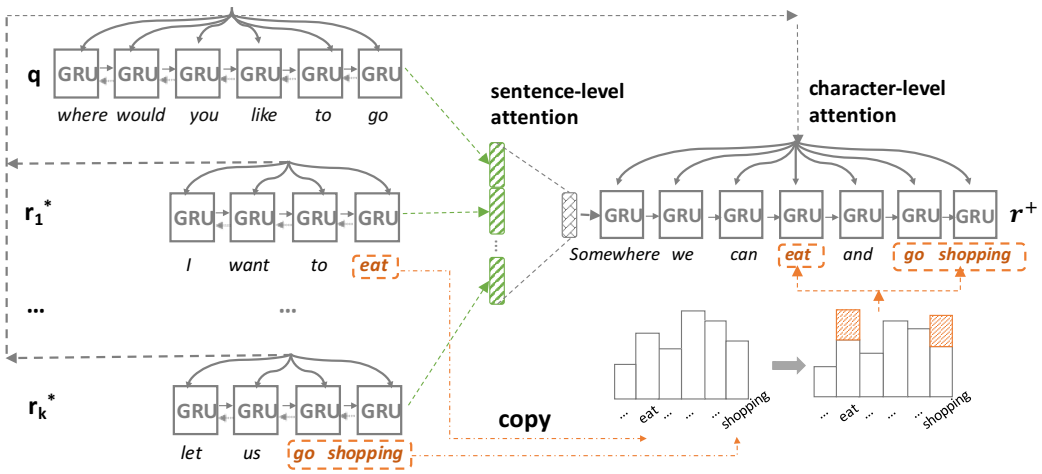

Figure 2: The `multi-seq2seq` model, which takes the query q and k retrieved candidate replies r^* as the input and generate a new reply r^+ as the output.

replies. l is the length of query, $\boldsymbol{h}_j$ is the hidden state of query, l_m is the length of r^*_m, $\boldsymbol{h}_{m,j}$ is the hidden state of r^*_m. $\boldsymbol{s}_i$ is the hidden state of decoder at time step i, $\alpha_{i,m,j}$ is the normalized attention weights for each word. $e_{i,m,j}$ is calculated by a bilinear matching function and M_a is the parameter matrix. Here, we omit the attention of query in equations for easier understanding.

• **Copy Mechanism.** `multi-seq2seq` also uses copy mechanism to explicitly extract words from the retrieved replies. For each word y_t in vocabulary V, the probability $p(y_t|\boldsymbol{s}_t)$ in decoding process is comprised of $k+1$ parts. The first part p_{ori} follows the original probability calculated by GRU/LSTM cells, and the following parts $p_{r^*_m}$ reflect the matching degree between the current state vector $\boldsymbol{s}_t$ and the corresponding states of y_t in encoders, given by,

$$p(y_t|\boldsymbol{s}_t) = p_{ori}(y_t|\boldsymbol{s}_t) + \sum_{m=1}^{k} p_{r^*_m}(y_t|\boldsymbol{h}_{y_t,m}) \quad (4)$$

$$p_{r^*_m}(y_t|\boldsymbol{h}_{y_t,m}) = \delta(\boldsymbol{s}_t M_c \boldsymbol{h}_{y_t,m}) \quad (5)$$

where $\boldsymbol{h}_{y_t,m}$ is the hidden states of retrieved reply r^*_m who responds y_t in decoder, $\delta(\cdot)$ is the sigmoid function, M_c is the parameter for matching $\boldsymbol{s}_t$ and $\boldsymbol{h}_{y_t,m}$. If y_t has not appeared in a retrieved replies r^*_m, the corresponding probabilities $p_{r^*_m}$ would be zero. Here, we do not copy words from query as queries and replies are not sharing the same vocabulary and word embeddings.

Both attention and copy mechanism aim to enrich the generated reply r^+ via useful and informative words extracted from retrieved replies $r^*_1, r^*_2, \ldots, r^*_k$. Figure 2 displays the design of `multi-seq2seq` model. We can see that the generated reply has tight relation with the query, and absorbs the keywords from the retrieved replies.

3.4 Re-ranker

Now we have k retrieved candidate replies r^* as well as a generated one r^+. As all the retrieved candidates are obtained via indirect matching, these replies need a further direct matching with the user-issued query. On the other hand, the generated reply set may contain the influent and meaningless utterances. Hence, we propose the second ensemble to derive the final ranking list by feeding all the candidates into a re-ranker.

We deploy a Gradient Boosting Decision Tree (GBDT) ([Ye *et al.*, 2009]) classifier, and it utilizes several high-level features, as listed in the following.

• *Term similarity.* The word overlap ratio captures the literal similarity between the query and reply. For both query and reply, we transform them into binary word vectors, in which each element indicates if a word appears in the corresponding sentence. We apply the cosine function to calculate the term overlap similarity of the query and the reply.

• *Entity similarity.* Named entities in utterances are a special form of terms. We distinguish *persons*, *locations* and *organizations* from plain texts with the help of named entity recognition techniques. Then we maintain the vectors of recognized entities for both query and its reply and calculate the similarity (measured by cosine similarity) between two entity-based vector representations.

• *Topic similarity.* "Topics" has long been regarded as the abstractive semantic representation ([Hofmann, 2001]). We apply Latent Dirichlet Allocation ([Blei *et al.*, 2003]) to discover the latent topics of the query and reply. The inferred topic representation is the probabilities for the piece of text belonging to each latent topic. By setting the topic number as 1000, which works efficiently in practice, we use the cosine similarity to calculate the topical score.

• *Statistical Machine Translation.* By treating queries and replies as different languages in the paradigm of machine translation, we train a translation model to "translate" the query into a reply based on the training corpora to get the translating word pairs (one word from a query and one word from its corresponding reply) with scores indicating their translating possibilities. To get the translation score for the query and reply, we sum over the translating scores of the word pairs extracted from these two sentences, and conduct normalization on the final score.

Dataset	# of samples
Retrieval (Repository)	7,053,820
Re-ranker (Train)	50,000
Generator (Train)	1,500,000
Validation	100,000
Testing	6,741

Table 2: Statistics of our datasets.

• *Length.* Since too short replies are not preferred, we take the length of replies as a point-wise feature and conduct a normalization to map the value to [0,1].

• *Fluency.* Fluency is to examine whether two neighboring terms have large co-occurrence likelihood. We calculate the co-occurrence probability for the bi-grams of the candidate replies and then take the average value as the fluency feature.

The confidence scores produced by the GBDT classifier are used to re-rank all the replies. The re-ranking mechanism can eliminate both meaningless short replies that are eventually generated by `multi-seq2seq` and less appropriate replies selected by the retrieval system. The *re-ranker* further ensures an optimized effect of model ensemble.

3.5 Model Training

Since our framework consists of learnable but independent components (i.e., `multi-seq2seq` and Re-ranker), the model training is constructed for each component separately. In `multi-seq2seq`, we use human-human utterance pairs $\langle q, r\rangle$ as data samples. k retrieved candidates r^* are also provided as the input when we train the neural network. Standard cross-entropy loss of all words in the reply is applied as the training objective. In the re-ranker part, the training samples are either $\langle q, r\rangle$ pairs or generated by negative sampling.

4 Evaluation

We evaluate our ensemble model in Chinese.

4.1 Experimental Setup

Both retrieval-based and generation-based components require a large database of query-reply pairs, whose statistics is exhibited in Table 2. To construct a database for information retrieval, we collected human-human utterances from massive online forums, microblogs, and question-answering communities, including Sina Weibo[4] and Baidu Tieba.[5] In total, the database contains 7 million query-reply pairs for retrieval. For each query, corresponding to a question, we retrieve k replies ($k = 2$) for generation part and re-ranker.

For the generation part, we use the dataset comprising 1,606,741 query-reply pairs originating from Baidu Tieba. Please note that q and r^* are the input of `multi-seq2seq`, whose is supposed to approximate the ground-truth. We randomly selected 1.5 million pairs for training and 100K pairs for validation. The left 6,741 pairs are used for testing both in generation part and the whole system. Notice that this corpus is different from the corpus used in the retrieval part so that the ground-truth of the test data are excluded in the retrieval module. The training-validation-testing split remains the same for all competing models.

To train our neural models, we implement code based on dl4mt-tutorial[6], and follow Shang *et al.* (2015) for hyper-parameter settings as it generally works well in our model. We did not tune the hyper-parameters, but are willing to explore their roles in conversation generation in future. All the embeddings are set to 620-dimension and the hidden states are set to 1000-dimension. We apply AdaDelta with a mini-batch ([Zeiler, 2012]) size of 80. Chinese word segmentation is performed on all utterances. We keep the set of 100k words for queries and 30K for the retrieval and generated replies due to efficiency concerns. The validation set is only used for early stop based on the perplexity measure.

4.2 Competing Methods

We compare our model ensemble with each individual component and provide a thorough ablation test. All competing methods are trained in the same way as our full model, when applicable, so that the comparison is fair.

• *Retrieval-1, Retrieval-2*. The top and second-ranked retrieved replies from a state-of-the-practice conversation system ([Yan *et al.*, 2016b]), which is a component of our model ensemble; it is also a strong baseline (proved in experiments).

• `seq2seq`. An encoder-decoder framework ([Sutskever *et al.*, 2014]), first introduced as neural responding machine by Shang *et al.* (2015).

• `multi-seq2seq`$^-$. Generation component, which only applies two-level attention strategies.

• `multi-seq2seq` . Generation component, which applies two-level attention and copy strategy.

• *Ensemble(Retrieval-1,Retrieval-2,* `seq2seq`*)*. Ensemble with retrieval and `seq2seq`.

• *Ensemble(Retrieval-1, Retrieval-2,* `multi-seq2seq`*)*. Ensemble with retrieval and `multi-seq2seq` . This is the full proposed model ensemble.

4.3 Overall Performance

• Subjective metric. Human evaluation, albeit time- and labor-consuming, conforms to the ultimate goal of open-domain conversation systems. We ask three educated volunteers to annotate the results ([Shang *et al.*, 2015; Li *et al.*, 2016b; Mou *et al.*, 2016]). Annotators are asked to label either "0" (bad), "1" (borderline), or "2" (good) to a query-reply pair. The subjective evaluation is performed in a strictly random and blind fashion to rule out human bias.

• Objective metric. We adopt BLEU 1-4 for the purpose of automatic evaluation. While Liu *et al.* (2016) further strongly argue that no existing automatic metric is appropriate for open-domain dialogs, we nonetheless include BLEU scores as the expedient objective evaluation, serving as supporting evidence. BLEUs are also used in Li *et al.* (2016a) for model comparison and in Mou *et al.* (2016) for model selection.

The automatic metrics were computed on the entire test set, whereas the subjective evaluation was based on 100 randomly chosen test samples due to the limitation of human resources.

[4]http://weibo.com

[5]http://tieba.baidu.com

[6]https://github.com/nyu-dl/dl4mt-tutorial

Method	Human Score	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Retrieval-1	1.013	**24.06**	10.04	5.232	2.784
Retrieval-2	0.528	4.532	0.655	0.476	0.471
seq2seq	0.880	6.349	0.665	0.111	0.039
Ensemble(retrieval-1, retrieval-2, seq2seq)	1.145	14.15	8.400	7.798	7.619
multi-seq2seq $^-$	0.918	9.290	2.489	1.144	0.566
multi-seq2seq	0.992	9.609	1.674	0.510	0.191
Ensemble(retrieval-1, retrieval-2, multi-seq2seq)	**1.362**	16.99	**11.13**	**10.37**	**9.993**

Table 3: Results of our ensemble and competing methods in terms of average human scores and BLEUs. Inter-annotator agreement for human annotation: Fleiss' $\kappa = 0.2932$ ([Fleiss, 1971]), std $= 0.3926$, indicating moderate agreement.

	Utterance (Translated)	
Query	This mobile phone's photo effect is pretty good.	
Retrieved-1	I really have a crush on it.	
Retrieved-2	Go for it.	
multi-seq2seq	Having a crush on it is not as good as action.	√
seq2seq	Aha.	
Query	Can I see the house tomorrow afternoon?	
Retrieved-1	You can call me!	
Retrieved-2	You can see the house on weekends.	
multi-seq2seq	You can see the house on weekends, please call me in advance.	√
seq2seq	OK.	

Table 4: Examples of retrieved and generated ones. "√" indicates the reply selected by the re-ranker.

We present our main results in Table 3. Table 4 presents two examples of our ensemble and its "base" models. As shown, the retrieval system, which our model ensemble is based on, achieves better performance than RNN-based sequence generation. This also verifies that the retrieval-based conversation system in our experiment is a strong baseline to compare with. Combining the retrieval system, generative system multi-seq2seq and the re-ranker, our model leads to the best performance in terms of both human evaluation and BLEU scores. Our model ensemble outperforms the state-of-the-practice retrieval system by $+34.45\%$ averaged human scores, which we believe is a large margin.

4.4 Analysis and Discussion

RQ1: What is the performance of multi-seq2seq (the **First Ensemble** in Figure 1) in comparison with seq2seq?

From BLEU scores in Table 3, we see both multi-seq2seq $^-$ and multi-seq2seq significantly outperform conventional seq2seq , and multi-seq2seq is slightly better than multi-seq2seq $^-$. These results imply the effectiveness of both two-level attention and copy mechanism. We can also see multi-seq2seq outperforms the second retrieval results in BLEUs. In the retrieval and seq2seq ensemble, 72.84% retrieved and 27.16% generated ones are selected. In retrieval and multi-seq2seq ensemble, the percentage becomes 60.72% vs. 39.28%. The trend indicates that multi-seq2seq is better than seq2seq from the re-ranker's point of view.

RQ2: How do the retrieval- and generation-based systems contribute to re-ranking (the **Second Ensemble** in Figure 1)?

As the retrieval and generation module account for 60.72% and 39.28% in the final results of retrieval and multi-seq2seq ensemble, they almost contribute equally to the whole framework. More importantly, we notice that retrieval-1 takes the largest proportion in two ensemble systems, and it may explain why most on-line chatting platforms choose retrieval methods to build their systems. Besides, multi-seq2seq decreases the proportion of retrieved one in the second ensemble systems.

RQ3: Since the two ensembles are demonstrated to be useful, can we obtain further gain by combining them together?

We compare the full model *Ensemble(Retrieval, multi-seq2seq)* with an ensemble that uses traditional seq2seq, i.e., *Ensemble(Retrieval, seq2seq)*. As indicated in Table 3, even with the re-ranking mechanism, the ensemble with underlying multi-seq2seq still outperforms the one with seq2seq. Likewise, *Ensemble(Retrieval, multi-seq2seq)* outperforms both *Retrieval* and *multi-seq2seq* in terms of most metrics.

Through the above ablation tests, we conclude that both first and second ensemble play a role in our ensemble when we combine the retrieval- and generation-based systems.

5 Conclusion

In this paper, we propose a novel ensemble of retrieval-based and generation-based open-domain conversation systems. The retrieval part searches the k best-matched candidate replies, which are, along with the original query, fed to an RNN-based multi-seq2seq reply generator. Then the generated replies and retrieved ones are re-evaluated by a re-ranker to find the final result. Although traditional generation-based and retrieval-based conversation systems are isolated, we have designed a novel mechanism to connect both modules. The proposed ensemble model clearly outperforms state-of-the-art conversion systems in the constructed large-scale conversation dataset.

Acknowledgments

This paper is partially supported by the National Natural Science Foundation of China NSFC Grant (NSFC Grant

Nos.61772039, 61472006 and 91646202) as well as the Microsoft grant NO. FY17-RES-THEME-031. The author Rui Yan is supported by the National Hi-Tech R&D Program of China (No. 2015AA015403) and the National Science Foundation of China (No. 61672058).

References

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *ICLR*, 2015.

[Blei *et al.*, 2003] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(993-1022), 2003.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv:1406.1078*, 2014.

[Fleiss, 1971] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378, 1971.

[He *et al.*, 2017] Shizhu He, Cao Liu, Kang Liu, and Jun Zhao. Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. *ACL*, pages 199–208, july 2017.

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

[Hofmann, 2001] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1):177–196, 2001.

[Isbell *et al.*, 2000] Charles Lee Isbell, Michael Kearns, Dave Kormann, Satinder Singh, and Peter Stone. Cobot in LambdaMOO: A social statistics agent. In *AAAI*, pages 36–41, 2000.

[Ji *et al.*, 2014] Zongcheng Ji, Zhengdong Lu, and Hang Li. An information retrieval approach to short text conversation. *arXiv:1408.6988*, 2014.

[Li *et al.*, 2016a] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *NAACL-HLT*, pages 110–119, 2016.

[Li *et al.*, 2016b] Xiang Li, Lili Mou, Rui Yan, and Ming Zhang. StalemateBreaker: A proactive content-introducing approach to automatic human-computer conversation. In *IJCAI*, pages 2845–2851, 2016.

[Libovicky and Helcl, 2017] Jindrich Libovicky and Jindrich Helcl. Attention strategies for multi-source sequence-to-sequence learning. *ACL*, pages 196–202, july 2017.

[Liu *et al.*, 2016] Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *EMNLP*, 2016.

[Mou *et al.*, 2016] Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. *COLING*, pages 3349–3358, 2016.

[Ritter *et al.*, 2011] Alan Ritter, Colin Cherry, and William B Dolan. Data-driven response generation in social media. In *EMNLP*, pages 583–593, 2011.

[Serban *et al.*, 2016a] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, pages 3776–3783, 2016.

[Serban *et al.*, 2016b] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. *arXiv:1605.06069*, 2016.

[Shang *et al.*, 2015] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. In *ACL-IJCNLP*, pages 1577–1586, July 2015.

[Sordoni *et al.*, 2015] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. A neural network approach to context-sensitive generation of conversational responses. In *NAACL-HLT*, pages 196–205, 2015.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[Vinyals and Le, 2015] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv:1506.05869*, 2015.

[Xing *et al.*, 2016] Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. Topic augmented neural response generation with a joint attention mechanism. *arXiv:1606.08340*, 2016.

[Yan *et al.*, 2016a] Rui Yan, Yiping Song, and Hua Wu. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In *SIGIR*, pages 55–64, 2016.

[Yan *et al.*, 2016b] Rui Yan, Yiping Song, Xiangyang Zhou, and Hua Wu. "shall i be your chat companion?": Towards an online human-computer conversation system. *CIKM*, pages 649–658, 2016.

[Ye *et al.*, 2009] Jerry Ye, Jyh-Herng Chow, Jiang Chen, and Zhaohui Zheng. Stochastic gradient boosted distributed decision trees. *Proceedings of the 18th ACM conference on Information and knowledge management.*, pages 2061–2064, 2009.

[Zeiler, 2012] Matthew D Zeiler. AdaDelta: An adaptive learning rate method. *arXiv:1212.5701*, 2012.

[Zoph and Knight, 2016] Barret Zoph and Kevin Knight. Multi-source neural translation. In *NAACL-ACL*, pages 30–34, 2016.