End-to-end Task-Oriented Dialogue System with Distantly Supervised Knowledge Base Retriever

Libo Qin, Yijia Liu, Wanxiang Che*, Haoyang Wen and Ting Liu
†Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, China
{lbqin, yjliu, car, hywen, tliu}@ir.hit.edu.cn

Abstract. Task-oriented dialog systems usually face the challenge of querying knowledge base. However, it usually cannot be explicitly modeled due to the lack of annotation. In this paper, we introduce an explicit KB retrieval component (*KB retriever*) into the seq2seq dialogue system. We first use the *KB retriever* to get the most relevant entry according to the dialogue history and KB, and then apply the copying mechanism to retrieve entities from the *retrieved KB* in decoding time. Moreover, the *KB retriever* is trained with distant supervision, which does not need any annotation efforts. Experiments on Stanford Multi-turn Task-oriented Dialogue Dataset shows that our framework significantly outperforms other sequence-to-sequence based baseline models on both automatic and human evaluation.

Keywords: task-oriented dialog systems, sequence-to-sequence, Knowledge Base

1 Introduction

Task-oriented dialogue system, which helps users to achieve specific goals with natural language, attracts more and more research attention. With the sequence-to-sequence (seq2seq) approaches being successfully applied in text generation [21, 1, 10, 15, 14, 21], several works tried to model the task-oriented dialogue as the seq2seq generation of response from the dialogue history [5, 4, 22]. This kind of modeling scheme frees the task-oriented dialogue system from the manually designed pipeline modules and heavy annotation labor for these modules.

Different from typical text generation, the success of serving users in the task-oriented dialogue system largely relies on the success of querying knowledge base (KB). Taking the dialogue in Fig. 1 for example, to answer the driver's query on the gas station, the dialogue system is required to pick out the entry that has the "gas station" (the fourth row in this table). To tackle the KB query challenges, Eric and Manning [4] use an additional copy mechanism to retrieve entities both in KB and dialogue history. Eric et al. [5] further proposed key-value retrieval network which incorporates world knowledge into its dialogue utterances via attention over the key-value entries of the underlying knowledge base.

* Email correspondence.

Address	Distance	POI type	POI	Traffic info
638 Amherst St	3 miles	grocery store	Sigona Farmers Market	car collision nearby
269 Alger Dr	1 miles	coffee or tea place	Cafe Venetia	car collision nearby
5672 barringer street	5 miles	certain address	5672 barringer street	no traffic
200 Alester Ave	2 miles	gas station	Valero	road block nearby
899 Ames Ct	5 miles	hospital	Stanford Childrens Health	moderate traffic
481 Amaranta Ave	1 miles	parking garage	Palo Alto Garage R	moderate traffic
145 Amherst St	1 miles	coffee or tea place	Teavana	road block nearby
409 Bollard St	5 miles	grocery store	Willows Market	no traffic

Driver: Address to the gas station.
Car: Valero is located at 200 Alester Ave.
Driver: OK , please give me directions via a route that avoids all heavy traffic.
Car: Since there is a road block nearby, I found another route for you and I sent it on your screen.
Driver: Awesome thank you.

Fig. 1. An example of a task-oriented dialogue that incorporates a knowledge base.

Besides using soft attention to model the interaction between dialogue history and KB entries, a component that directly retrieves the KB was used in dialogue pipeline[9, 23]. However, such component is generally considered intractable for the seq2seq dialogue system because probabilistically modeling calls for annotated data which are absent in the seq2seq settings. Past decades witness the success of the distant supervision in information extraction [26, 12, 11, 25] , which induces the training signal from a set of heuristic on the existing KB. Inspired by this line of research, we explore the possibility of introducing an explicit KB retrieval component into the seq2seq dialogue system and train this component with distant supervision.

In this paper, we propose a novel seq2seq dialogue system that explicitly queries KB and uses the queried result to generate the response. A KB retrieving component (retriever) is proposed to model the interaction between dialogue history and the KB and its trained with a novel distant supervision algorithm. In practice, KB retriever first gets the most relevant entry given dialogue history and KB, and then perform column attention to get retrieved KB cell based on the selected entry while decoding time. Finally, the retrieved KB cell is then fed into a copy network to generate the final response. Our method represents a shift in perspective compared to existing work, we not only follow the basic method of task-oriented dialog based on seq2seq model, but also explicitly model a KB retrieving component into the basic seq2seq framework. Moreover, the KB retrieving component is trained with a novel distant supervision which doesn't need heavy annotation. Experiments on Stanford Multi-turn, Dialogue Dataset [5] verify the effectiveness of our method by significantly outperforming the baseline in both the automatic and human evaluation.

Our contributions can be summarized as follows:

– We propose a *KB retriever* based seq2seq model in task-oriented dialogue systems, which can greatly improve the ability of entire system interacting with and querying the knowledge base.

– In our framework, the *KB retriever* is trained with a novel distant supervision which does not need heavy human annotation.
– Experiments on a publicly available dataset show that our approach significantly and consistently outperforms all baselines.

2 Related Work

Historically, task-oriented dialog systems have been built as pipelines of separately trained modules. A typical pipeline design contains four components: 1) a user intent classifier, 2) a belief tracker, 3) a dialogue policy maker and a 4) response generator. Recently, the powerful distributed representation ability of neural networks makes task-oriented dialogue system end-to-end possible. Wen et al. [22] built a system that connects classic pipeline modules by a policy network. It queries KB by a database operator which is consistent with the most likely belief state. However, their modules like belief tracker still needs to be trained separately before end-to-end training. Unlike their work, our framework use an explicit KB retriever to extract useful information from a knowledge base, without the need for explicit training of belief or intent trackers. Other dialogue agents can also interface with the database by augmenting their output action space with predefined API calls [13, 27, 2, 9]. While Dhingra et al.[3] applied a soft-KB lookup on an entity-centric knowledge base to compute the probability of that the user knows the values of slots, and has tried to model the posterior distributions over all slots. However, our framework does not require any slots information. Eric and manning [4] use an additional copy mechanism to retrieve entities both in KB and dialogue history. Eric et al.[5] further introduced retrieval from key-value KB based seq2seq model. The key difference between our work and their work is that they query the KB only by attention-based method while our model proposes an explicit *KB retriever* component to query KB into a seq2seq framework. Inspired to those works of the distant supervision in information extraction [26, 12, 11, 25]. we train our *KB retriever* component with distant supervision and collect the training data only by history dialogue and the existing KB, which doesn't need heavy human annotation.

3 Method

In this section, we describe our framework for task-oriented dialogue system. Our framework consists of a *KB retriever* that takes the encoded dialogue history along with the representation of all KB entries as input and returns the most possible KB entry (*retrieved KB*) (§3.3), and an *encoder-decoder* framework that takes the *retrieved KB* and an attentively represented dialogue history and use a copy network [6] to determine the next generated token.

3.1 Problem Definition and Notation

Dialogue History. Given a dialogue between a user (u) and a system (s), we follow Eric and Manning [5, 4] and represent the k-turned *dialogue utterances*

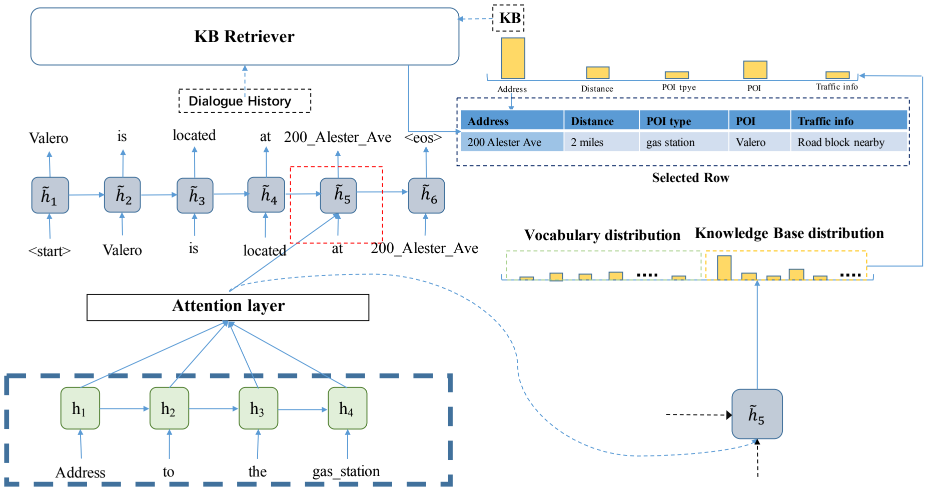

Fig. 2. . Given with dialogue history and KB, the *KB Retriever* return the *Retrieved KB Row*. For each time-step of decoding, the cell state is used to compute an attention over the encoder states and a separate column attention over the column of *Retrieved KB Row*. The attention over the encoder is used to generate a context vector which is combined with the cell state to get a distribution over the normal vocabulary. The hierarchical attention over the column of the KB become the logits for their associated entity in a now augmented vocabulary that we argmax over.

as $\{(u_1, s_1), (u_2, s_2), ..., (u_k, s_k)\}$. At the i^{th} turn of the dialogue, we aggregate dialogue context which consists of the tokens of $(u_1, s_1, ..., s_{i-1}, u_i)$ and use $\mathbf{x} = (x_1, x_2, ..., x_m)$ to denote the whole *dialogue history* word by word, where m is the number of tokens in the dialogue.

Knowledge Base. In this paper, we assume to have the assessment of a relational database-like KB T, which consists of several rows and five columns. Each column is associated with a attribute name f.

Sequence-to-Sequence Task-Oriented Dialogue. We define the seq2seq task-oriented dialogue as finding the most likely response sequence according to the input dialogue history and KB. Formally, it is defined as

$$p(\mathbf{y} \mid \mathbf{x}, T) = \prod_{t=1}^{n} p(y_t \mid y_1, ..., y_{t-1}, \mathbf{x}, T)$$

where y represent an output token.

3.2 Vanilla Sequence-to-Sequence Task-Oriented Dialogue System

Eric and Manning [5] proposed the vanilla seq2seq task-oriented dialogue system. In their model, a long short term memory (LSTM, [7]) is used to encode the dialogue history $\mathbf{x}$. More specifically, the tokens in $\mathbf{x}$ are mapped to vectors with embedding function ϕ^{emb}. The vectors are then fed into LSTM to produce context-sensitive hidden representations $(h_1, h_2, ..., h_m)$, by repeatedly applying the recurrence $h_i = \text{LSTM}\left(\phi^{emb}(x_i), h_{i-1}\right)$.

LSTM is also used to represent the partially generated output sequence $(y_1, y_2, ..., y_{t-1})$ as $(\tilde{h}_1, \tilde{h}_2, ..., \tilde{h}_t)$. For the generation of next token y_t, their model first calculates an attentive representation $\tilde{h'}_t$ of the dialogue history as

$$
\begin{aligned}
u_i^t &= w^T \tanh(W_2 \cdot \tanh(W_1 \cdot [h_i, \tilde{h}_t])) \\
a_i^t &= \text{softmax}(u_i^t) \\
\tilde{h'}_t &= \sum_{i=1}^{m} a_i^t \cdot h_i
\end{aligned}
$$

Finally, a concatenation of the hidden representation of outputted sequence $\tilde{h}_t$ and the attentive dialogue history representation $\tilde{h'}_t$ are projected to the vocabulary space by U as

$$
\begin{aligned}
o_t &= U \cdot [\tilde{h}_t, \tilde{h'}_t] \\
p(y_t \mid y_1, ..., y_{t-1}, \mathbf{x}, T) &= \text{softmax}(o_t)
\end{aligned}
$$

where $\mathcal{V}$ is the vocabulary and $y \in \mathcal{V}$.

Seq2Seq Task-Oriented Dialogue with Copy Net. To enable the network to generate the entry in KB, Eric and Manning [5] also proposed an augmented decoder that decodes over the combination of vocabulary and candidate entries in KB. In [5], the logit o_t is expanded with a KB-attention score v^t as

$$
o_t = U \cdot [\tilde{h}_t, \tilde{h'}_t] + v^t
$$

where o_t's dimensionality is $|\mathcal{V}| + |\mathcal{E}|$. In v^t, lower $|\mathcal{V}|$ is zero and the rest is $|\mathcal{E}|$ attention scores. Our major difference with Eric and Manning [5] is that we don't use attention scores of the whole KB to augment o_t but the scores of one concrete row (*retrieved KB*) of the relational KB.

3.3 KB Retriever

As described in Section 1, our goal is to examine the possibility for directly retrieves the KB. To accomplish this goal, we propose a *KB retriever*.

Dialogue History Representation. We encode the dialogue history by adopting the neural bag-of-words (BoW). Each token in the dialogue history is mapped into a vector by another embedding function $\phi^{emb'}(x)$ and the dialogue history representation E_d is computed as summing these vectors: $E_d = \sum_{i=1}^{m} \phi^{emb'}(x_i)$

KB Encoder. In this section, we describe how we encode the KB table. Each KB cell is represented as the concatenation of the column name embedding $\phi^{\text{name}}(f)$ and the cell value v embedding $\phi^{\text{value}}(v)$. This representation is further fed into a tanh non-linearity and the final representation can be formalized as

$$c = \tanh\left(W^{c}\left[\phi^{\text{value}}(v), \phi^{\text{name}}(f)\right]\right)$$

The representation of a row of KB C_k is denoted as $C_k = [c_{k,1}, ..., c_{k,m}]$, where m represents number of column attributes.

KB Retriever. Past decades witness the success of the memory network [20] in some reasoning tasks [17, 24]. Inspired by those works, we follow the structure of memory network to explore the deep correlation of the dialogue history and the every KB row, hoping to help us reasoning and find the KB row most relevant to the Dialogue history. In practice, we consider the dialogue history as the query which mentioned in [20] and regard KB as the information should be stored in memory. We model the retrieval process as a hierarchical classification over KB, which first select the row, then select the column.

For the row selection. We take the encoding of dialogue history E_d and the table encoding as input, which are fed into multi-hop memory network to get the relevance score of every row in the KB. Finally, we select the row that corresponds to the maximum score. Below we describe how to get the probability distribution of each row through the memory network given the dialog history and table encode. In our model, we give a row entity of KB set $C_1, ..., C_i$ to be stored in memory. The entire set of $[c_{k,1}, ..., c_{k,m}]$ are converted into memory vectors m_k of dimension d computed by embedding each KB cell $c_{k,i}$ and sums the resulting vectors: $m_k = \Sigma_j c_{k,j}$. Then, we compute the match between E_d and each memory m_k by taking the inner product followed by a softmax:

$$p_k = \operatorname{softmax}\left(E_d^T m_k\right)$$

where $\operatorname{softmax}(z_i) = e^{z_i} / \sum_j e^{z_j}$. Defined in this way p is a probability vector over the row entities of KB set. Each C_i has a corresponding output vector z_i given in the simplest case by another embedding matrix M. The response vector from the memory o is then a sum over the transformed inputs z_i , weighted by the probability vector from the input:

$$o = \sum_i p_i z_i$$

In the single layer case, the sum of the output vector o and the dialogue history representation E_d is then passed through a final weight matrix W and a softmax to produce the predicted logits:

$$\tilde{a} = \operatorname{softmax}\left(W\left(o + E_d\right)\right)$$

In our framework, we also explore the multi-hop memory network. The memory layers are stacked in the following way:

– The input to layers above the first is the sum of the output o^k and the input E_d^k from layer k.
– At the top of the network, the input to W also combines the input and the output of the top memory layer: $\tilde{a} = \text{softmax}\left(W\left(o^k + E_d^k\right)\right)$.

where $\tilde{a}$ represents the predicted row logits of KB which is used to query KB. Its dimension size is the number of KB's row rather than the size of word vocabulary which is the difference between our model and [20]. Based on $\tilde{a}$, we select the row with the largest probability value as the *retrieved KB*. Moreover, we use the adjacent type of weight typing to reduce the number of parameters and use Temporal Encoding to improve the performance of the *KB retriever*.

For the column selection. After getting the *retrieved KB*, we perform column attention in decoding time to select column of KB. We use the decoder hidden state $(\tilde{h}_1, \tilde{h}_2, ..., \tilde{h}_t)$ to compute an attention score with the embedding of column attribute name. The attention logits then become the logits of the column be selected based *retrieved KB*. Finally, we use a copy network to determine the next generated token. Similar to [5], the final logit o_t is expanded with a KB-attention score v^t as

$$o_t = U \cdot [\tilde{h}_t, \tilde{h'}_t] + v^t$$

where o_t's dimensionality is $|\mathcal{V}| + |\mathcal{E}|$. In v^t, lower $|\mathcal{V}|$ is zero and the rest is $|\mathcal{E}|$ attention scores. We just use attention scores of one concrete row of the *retrieved KB*, not use the whole KB to augment o_k, which is the key difference between our work and Eric and Manning [5]. This description seeks to capture the intuition that when in response to the query *Address to the gas station* in Fig 1, our *KB retriever* have selected the fourth KB row which includes the correct response entries *Valero* and *200 Alester Ave*. Therefore, our model only put an attention weight on the *retrieved KB* rather than the whole KB, which can improve the performance of response results. We provide a visualization of the whole framework in Fig 2.

4 Data Collection for training the Retriever with Distant Supervision

In this section, we talk about how we collect the training data for the *KB retriever*. Different from other works that need heavy human annotation, we only use the dialogue history and the existing KB to collect our training data.

Given with dialogue history $(x_1, x_2, ..., x_m)$ and the KB C_i (i represents the row index of KB), we can use a simple match algorithm to collect training data. For every row of KB, take the k^{th} row ($[c_{k,1}, ..., c_{k,m}]$) for example, we judge whether the cells$(c_{k,1}, ..., c_{k,m})$ of each row of KB have appeared in the dialogue history $(x_1, x_2, ..., x_m)$. If they match, the counter is incremented by one, and then we get a match score for each row. Finally, we select the row corresponding to the largest match score as our selected row.

Take the dialogue and KB in Fig. 1 for example, we show how we get the match score of the fourth row of KB. First, we initialize every row's match counter to zero. Then, for every cell in the fourth row(200 Alester Ave, 2 miles, gas station, Valero, road block nearby), we find those cells (200 Alester Ave, gas station, Valero) can be matched in the dialogue history. So we change the value of counter to three. After getting each row's match score, we select the row corresponding to the largest match score as our selected row. The intuition is that we believe the knowledge base with the largest number of matched entities in the dialogue history is the supported KB row in most of the time. Through the above steps, we get the training data for training the retriever.

5 Experiments

In this section, we first introduce the details of the experiments and then present results from both automatic and human evaluation. Then we provide results and analyses of automatic evaluation and human evaluation. Besides, we present ablation test to evaluate and analyze the function of different components in our framework.

5.1 Experiment Setting

We choose a KB-rich domain from Stanford Multi-turn Multi-domain Task-oriented Dialogue Dataset [5], which is point-of-interest navigation.

Our framework is trained separately in these two stages, using the same train/validation/test split sets as [5]. We do not map the entities in dialogue into its canonical form as what [5] have done, since our framework extract entities directly from KB. And we evaluate our framework on exact entities as well. In the first stage, we applied three hops and weight typing to train memory network for positioning row of KB. In the second stage, we trained our main framework by an end-to-end approach. Our framework is trained using the Adam optimizer [8]. The learning rate is 10^{-3}. We applied dropout[19] to the input and the output of LSTM, with a dropout rate at 0.75. We add the weight decay on the model. The coefficient of weight decay is $5 * 10^{-6}$. The embedding size and all hidden size are 200. The number of epochs for pretraining memory network is 100 for and the number of hops is 3.

5.2 Baseline Models

We provide several baseline models for comparing the performance of our whole framework:

- **Copy-augmented Sequence-to-Sequence Network.** This model is adapted from [4]. It augments a sequence-to-sequence architecture with encoder attention, with an additional attention-based hard-copy mechanism over the KB entities mentioned in the encoder context.

– **Key-value Retrieval Network.** This model is adapted from [5]. It utilizes key-value forms to represent KBs. Key representations are used for an attention-based value retrieval. Note that in the original paper, they simplified the task by mapping the expression of entities to a canonical form using named entity recognition (NER) and linking.

5.3 Automatic Evaluation

In this section, we provide two different automatic evaluations to compare with other baseline models. The results and analyses are provided in the following sections.

Evaluation Metrics:

– **BLEU.** We use the BLEU metric, commonly employed in evaluating machine translation systems [16], which has also been used in past literature for evaluating dialogue systems both of the chatbot and task-oriented variety [18, 6, 22]. Hence, we include BLEU score in our evaluation (i.e. using Moses multi-bleu.perl script).

– **Entity F1.** We micro-average and macro-average the entire set of system responses and compare the entities in plain text. The entities in each gold system response are selected by a predefined entity list. This metric evaluates the ability to generate relevant entities from the provided KBs and to capture the semantics of the dialogue flow [4, 5].

Results and Analyses. Experiment results are illustrated in Table 1. The results show that our model outperforms other models in all automatic evaluation metrics. Compared to KV Net, we achieve 2.85 improvements on BLEU score and 20.5 improvements on Micro F1. And compared to Copy Net, we achieve 2.88 improvements on BLEU score and 26.3 improvements Macro F1. The results in navigation show our model's capability to generate more natural and meaningful response than the Seq2Seq baseline models.

We also find that the KV Net's results are lower than that reported by [5]. We address this to the differences in the preprocessing, model training and evaluation metrics. In spite of the difference of evaluation metrics that we evaluate on exact entities rather than their canonical forms, the Micro F1 score of our model still outperforms what [5] reported, which is 41.3 in navigation domain and which is evaluated on canonical forms.

Ablation In this section, we perform several ablation experiments to evaluate different components in our framework on the navigation domain. The results are shown in Table 2. The results demonstrate the strong impact that components of our model to the final performance.

Model	BLEU	Micro F1	Macro F1
Seq2Seq with Attention	8.32	17.5	15.6
Copy Net	8.67	23.7	20.8
KV Net	8.70	29.5	24.9
our model	**11.55**	**50.0**	**42.8**

Table 1. Automatic evaluation on test data. Best results are shown in bold. Generally, our framework significantly outperforms other models in all automatic evaluation metrics.

Model	BLEU	Micro F1	Macro F1
our model	**11.55**	**50.0**	**42.8**
-copying	8.9	26.0	22.7
-*KB retriever*	10.42	33.2	30.1

Table 2. Ablation experiment on navigation domain. -copy refers to a framework without copying. -*KB retriever* refers to a framework without *KB retriever*

.

Model	Correct	Fluent	Humanlike
Copy Net	4.01	4.58	4.51
KV Net	4.23	4.68	4.56
our model	**4.66**	**4.81**	**4.78**

Table 3. Human evaluation of responses based on random selected previous dialogue history in test dataset.

Copying mechanism enables our framework to retrieve entities directly from KBs. Without copying mechanism, such retrieval is infeasible and our framework cannot produce values in KBs. The results show that it introduces more variability to the generation process if we do not use copying mechanism.

KB retriever first retrieve the KB row most relevant to Dialogue history is the key difference between our model with other baselines. It can effectively reduce the scale of KB while in decoding time, which can improve the performance of the generation.

5.4 Human Evaluation

In this section, we provide human evaluation on our framework and other baseline models. We generated all responses in test dataset. These responses are based on distinct dialogue history. We hire many human experts, and they were asked to judge the quality of their responses according to correctness, cooperativeness, and humanlikeness on a scale from 3 to 5. And each judgment indicates a relative score compared to the standard response from test data. The results are illustrated in Table 3. The results show that our framework outperforms other baseline models on all metrics. The most significant improvement is from correctness, indicating that our model generates more accurate information that the users want to know.

6 Conclusion

In this work, we explore the possibility of introducing an explicit KB retrieval component (*KB retriever*) into the seq2seq dialogue system. Our framework performed an explicit *KB retriever* to lookup over the knowledge base, and applied the copying mechanism to retrieve entities from the *retrieved KB* while decoding. Besides, the *KB retriver* component is trained with distant supervision, which does not need heavy human annotation. Experiments showed that our model outperforms other competitive Seq2Seq models on both automatic and human evaluation metrics. In the future, we would like to jointly model the *KB retriever* and seq2seq framework in an end-to-end training method.

Acknowledgements

We are grateful for helpful comments and suggestions from the anonymous reviewers. This work was supported by the National Key Basic Research Program of China via grant 2014CB340503 and the National Natural Science Foundation of China (NSFC) via grant 61632011 and 61772153.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Bordes, A., Boureau, Y.L., Weston, J.: Learning end-to-end goal-oriented dialog. arXiv preprint arXiv:1605.07683 (2016)
3. Dhingra, B., Li, L., Li, X., Gao, J., Chen, Y.N., Ahmed, F., Deng, L.: End-to-end reinforcement learning of dialogue agents for information access. arXiv preprint arXiv:1609.00777 (2016)
4. Eric, M., Manning, C.D.: A copy-augmented sequence-to-sequence architecture gives good performance on task-oriented dialogue. arXiv preprint arXiv:1701.04024 (2017)
5. Eric, M., Manning, C.D.: Key-value retrieval networks for task-oriented dialogue. arXiv preprint arXiv:1705.05414 (2017)
6. Gu, J., Lu, Z., Li, H., Li, V.O.: Incorporating copying mechanism in sequence-to-sequence learning. arXiv preprint arXiv:1603.06393 (2016)
7. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation 9(8), 1735–1780 (1997)
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
9. Li, X., Chen, Y.N., Li, L., Gao, J.: End-to-end task-completion neural dialogue systems. arXiv preprint arXiv:1703.01008 (2017)
10. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025 (2015)
11. Min, B., Grishman, R., Wan, L., Wang, C., Gondek, D.: Distant supervision for relation extraction with an incomplete knowledge base. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 777–782 (2013)

12. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2. pp. 1003–1011. Association for Computational Linguistics (2009)
13. Möller, S., Englert, R., Engelbrecht, K., Hafner, V., Jameson, A., Oulasvirta, A., Raake, A., Reithinger, N.: Memo: towards automatic usability evaluation of spoken dialogue services by user error simulations. In: Ninth International Conference on Spoken Language Processing (2006)
14. Nallapati, R., Xiang, B., Zhou, B.: Sequence-to-sequence rnns for text summarization (2016)
15. Nallapati, R., Zhou, B., Gulcehre, C., Xiang, B., et al.: Abstractive text summarization using sequence-to-sequence rnns and beyond. arXiv preprint arXiv:1602.06023 (2016)
16. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 311–318. Association for Computational Linguistics (2002)
17. Pavez, J., Allende, H., Allende-Cid, H.: Working memory networks: Augmenting memory networks with a relational reasoning module. arXiv preprint arXiv:1805.09354 (2018)
18. Ritter, A., Cherry, C., Dolan, W.B.: Data-driven response generation in social media. In: Proceedings of the conference on empirical methods in natural language processing. pp. 583–593. Association for Computational Linguistics (2011)
19. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research 15(1), 1929–1958 (2014)
20. Sukhbaatar, S., Weston, J., Fergus, R., et al.: End-to-end memory networks. In: Advances in neural information processing systems. pp. 2440–2448 (2015)
21. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in neural information processing systems. pp. 3104–3112 (2014)
22. Wen, T.H., Vandyke, D., Mrksic, N., Gasic, M., Rojas-Barahona, L.M., Su, P.H., Ultes, S., Young, S.: A network-based end-to-end trainable task-oriented dialogue system. arXiv preprint arXiv:1604.04562 (2016)
23. Williams, J.D., Asadi, K., Zweig, G.: Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. arXiv preprint arXiv:1702.03274 (2017)
24. Xiong, C., Merity, S., Socher, R.: Dynamic memory networks for visual and textual question answering. In: International Conference on Machine Learning. pp. 2397–2406 (2016)
25. Xu, W., Hoffmann, R., Zhao, L., Grishman, R.: Filling knowledge base gaps for distant supervision of relation extraction. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). vol. 2, pp. 665–670 (2013)
26. Zeng, D., Liu, K., Chen, Y., Zhao, J.: Distant supervision for relation extraction via piecewise convolutional neural networks. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1753–1762 (2015)
27. Zhao, T., Eskenazi, M.: Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. arXiv preprint arXiv:1606.02560 (2016)