Entity Resolution in Open-domain Conversations

Mingyue Shang, Tong Wang, Mihail Eric, Jiangning Chen, Jiyang Wang
Matthew Welch, Tiantong Deng, Akshay Grewal, Han Wang, Yue Liu
Imre Kiss, Yang Liu, Dilek Hakkani-Tur

Amazon Alexa
{myshang, tonwng, mihaeric, cjiangni, welcmtt, jiyangw, gracdeng,
aksgrewa, wnghn, lyu, ikiss, yangliud, hakkanit}@amazon.com

Abstract

In recent years, incorporating external knowledge for response generation in open-domain conversation systems has attracted great interest. To improve the relevance of retrieved knowledge, we propose a neural entity linking (NEL) approach. Different from formal documents such as news, conversational utterances are informal and multi-turn, which makes it more challenging to disambiguate the entities. Therefore, we present a context-aware named entity recognition model (NER) and entity resolution (ER) model to utilize dialogue context information. We conduct NEL experiments on three open-domain conversation datasets and validate that incorporating context information improves the performance of NER and ER models. Furthermore, we verify that using knowledge sentences identified based on NEL benefits the neural response generation model.

1 Introduction

Building an informative open-domain conversational agent that can naturally interact with humans has been one of recent scientific research topics. Inspired by the development of neural networks, neural generation based conversation systems have made great progress (Sutskever et al., 2014; Vinyals and Le, 2015; Li et al., 2017; Wolf et al., 2019a; Zhou et al., 2020). However, one issue in such approaches is that the neural models often produce universal and less informative responses (Huang et al., 2020). To address this issue, previous work proposed to incorporate external knowledge into the response generation models, such as topics (Xing et al., 2017) and emotions (Zhou et al., 2018a). One line of research investigates the use of external knowledge to enrich the information of the responses (Ghazvininejad et al., 2018; Young et al., 2018; Dinan et al., 2018; Gopalakrishnan et al., 2019; Meng et al., 2020).

Most existing studies retrieve relevant knowledge from a knowledge base using the entities and noun phrases in the input text. Thus, correctly identifying these entities is crucial to find the relevant knowledge for a given dialog context. This typically involves two subtasks: given a user utterance, the system first identifies any named entities it contains (NER task) and then performs entity resolution (ER) to disambiguate the mentioned entities using a knowledge base. Both NER and ER (or NEL) have been well explored in previous studies and demonstrated to perform highly for news or well written text. However, for open domain spoken conversations and human-bot dialog, performance suffers due to ASR errors, incomplete or ungrammatical sentences from users, difference of spoken and written style, and less training data for such tasks.

In this paper, we propose to use neural entity linking (NEL) technologies that leverage both utterance-level and dialog-level context to retrieve relevant knowledge. As shown in the example in Figure 1, dialogues often contain multiple turns and information is dispersed throughout each turn. Thus, a single turn of interaction may be insufficient for entity disambiguation. Therefore, we leverage previous utterances in the dialogue as the context information and propose context-aware models to better solve the NER and ER tasks in open-domain conversation systems. When recognizing and disambiguating entities in a given utterance, we encode dialog context, and adopt the attention mechanism to extract the information related to the current utterance. To verify the effectiveness of context-aware models, in addition to the intrinsic evaluations, i.e., NER and ER standalone performance, we conduct an extrinsic evaluation where NER and ER results are integrated in a knowledge grounded neural response generation model in an open domain conversation system and response quality is evaluated. Our major contributions can
be summarized as follows:

- We propose neural network based context-aware models for NER and ER respectively in open domain conversations.
- Experimental results on different conversation datasets show that our proposed context-aware NER and ER models outperform other state-of-the-art models that do not use context information.
- In an end2end evaluation, we demonstrate that incorporating ER information improves quality of neural response generation models in open domain conversations.

2 Related Work
2.1 Open-domain Conversation System
Inspired by the availability of conversational data and the prosperity of neural networks, building open-domain conversation systems by data-driven approaches has achieved great progress. Previous methods can be roughly divided into two categories, retrieval-based (Zhang et al., 2018; Wu et al., 2019; Tao et al., 2019) and generation-based (Vinyals and Le, 2015; Li et al., 2017; Asghar et al., 2018; Tao et al., 2018). Chen et al. (2017) point out that conventional sequence-to-sequence methods tend to generate trivial responses that lack information and diversity. To address this issue, a line of research proposes to incorporate external knowledge into the generation process. Most of the work in this line retrieves knowledge based on a search or retrieval step first, and followed by further reranking of retrieved relevant knowledge snippets (Ghazvininejad et al., 2018; Young et al., 2018; Zhou et al., 2018b; Gopalakrishnan et al., 2019; Zhao et al., 2020). In our work, we propose neural entity recognition and linking to identify and resolve entities more accurately in order to obtain more relevant knowledge for knowledge grounded response generation.

2.2 Neural Entity Linking
NEL typically involves two tasks: recognizing named entities in a given text and then disambiguating the entity mentions according to the knowledge base (KB). Researchers have shown great success in NER with the help of Convolutional Neural Networks (CNNs), Bidirectional Recurrent Neural Networks (Bi-RNNs), and attention mechanisms along with a CRF decoder (Chiu and Nichols, 2016; Akbik et al., 2018; Ghaddar and Langlais, 2018; Jiang et al., 2019; Baevski et al., 2019; Yamada et al., 2020). Deep neural networks (DNNs) are also dominant in entity resolution tasks. They are used to calculate the semantic similarity between the recognized entity mentions and the entities in the KB (Yamada et al., 2016; Ganea and Hofmann, 2017; Sil et al., 2018; Raiman and Raiman, 2018). However, previous NEL work has mainly focused on news or formal documents, which is different from open-domain dialogues in many aspects. Sentences in open-domain dialogues are more informal, making it more difficult to recognize and disambiguate entities. In addition, since conversations are multi-turn, the semantic information in the current utterance is ambiguous and context needs to be considered. In this paper, we investigate NEL in open-domain conversational data and propose context-aware NER and ER models.
3 Methodology

3.1 Problem Formulation

Our problem can be formulated as follows. Given an open-domain dialogue until a time point $D = c_i, x_i$, where $x_i$ is the current utterance, we define the utterance context $c_i = \{u_1, \ldots, u_k\}$ as the list of utterances prior to $x_i$, and $k$ is the size of the context. For each $x_i$ given $c_i$, an NER model is applied to detect entity mentions in the form of BIO labels. Then for each predicted entity mention, $y_j$, a query is formulated to search a knowledge base to get a list of candidate entities, $\{e_1, \ldots, e_m\}$, where $m$ is the size of the returned entities from the search. An ER model is then used to rank the entities and identify the most relevant entity, $e_t$. Finally, a response, $r_i$, is generated based on $c_i, x_i$, and knowledge sentences obtained from the linked entities $e_t$. Note a knowledge ranking algorithm is applied when there are multiple knowledge sentences corresponding to $e_t$ or there are multiple entity mentions in $x_i$. Figure 1 overviews the pipeline of generating responses with NER and ER modules.

3.2 Context-Aware Named Entity Recognition Model

Figure 2 gives the overall architecture of the context-aware NER model. Following the framework presented by Chiu and Nichols (2016), we employ a bi-directional, long short-term memory (Bi-LSTM) model to extract word features and a conditional random field (CRF) to predict the NER labels.

Suppose we have an utterance $x_i = \{w_1^i, \ldots, w_T^i\}$, where $T$ is the length of $x_i$ and $w_t^i$ is the $t$-th token. After converting each token in $x_i$ to its vector representation through a word embedding table$^2$, the Bi-LSTM layer encodes the sentence into hidden states $h_t^i$, which are the concatenation of $\overrightarrow{h}_t^i$ from the forward LSTM and $\overleftarrow{h}_t^i$ from the backward LSTM. The CRF layer then takes the hidden states as input to predict the label probability.

As discussed earlier, as opposed to news or documents, recognizing and disambiguating the named entities in conversational utterances requires consideration of the context information. Therefore, we employ another Bi-LSTM layer to encode the context utterances from the previous turns, $s_j^t = [\overrightarrow{s}_t^j; \overleftarrow{s}_t^j]$ (1)

where $\overrightarrow{s}_t^j$ is the forward hidden state of the $t$-th token in the context utterance $u_j$ and $\overleftarrow{s}_t^j$ is the backward hidden state.

We use an attention mechanism to model the different impact of the previous utterances in the context:

$$\text{Attention} = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V \quad (2)$$

where $Q, K, V$ refer to the query, key, and value, respectively. Here, the key and value are the context sentences, and the query is the current utterance. To aggregate the context information, a max-pooling operation is performed on the dimension of sentences. Then, the context vector is concatenated with the sentence vector, and then is supplied as the input of the CRF layer.

3.3 Context-aware Entity Resolution Model

Our entity resolution model contains two steps: coarse-grained candidate selection and fine-grained candidate ranking.

**Candidate selection** At this stage we retrieve relevant entities from the KB. We create an Elasticsearch (Gormley and Tong, 2015) index with the entity labels and apply both an exact and a Levenshtein distance based fuzzy match to obtain candidate entities. For each entity mention, we take the top 10 search results, ranked by Elasticsearch, as the candidates for the subsequent reranking step.

$^1$These labels are widely used for NER and indicate a token is Begin, Inside, or Outside an entity mention, respectively.

$^2$Here we adopt the stacked embedding released by Flair (Akbik et al., 2018).
Reranking  At this stage the candidate entities are re-ranked based on the match scores from our context-aware model. We propose to compute the relevance score from the entity, utterance and session levels. The structure of the multi-level reranking model is shown in Figure 3.

Entity-Level Matching: This considers the candidate entity’s label and type attributes, and matches with the entity mention and the predicted type, respectively.

Utterance-Level Matching: This measures the matching degree between the candidate entity’s description and the current utterance based on sentence-level semantic information.

Session-Level Matching: This treats the context and current utterance as a conversation session, and computes its match score with the candidate entity’s description.

For each matching level, we first concatenate the representations from the entity candidate in the KB and the dialog side, and then employ BERT (Devlin et al., 2018) to get their representations. $v_{label}$, $v_{type}$, $v_{utterance}$, $v_{session}$ represent the output of BERT corresponding to the mention label and type (entity-level), utterance-level, and session-level, respectively. We also define the popularity of an entity based on the number of views in the last 60 days, represented as $v_p$. All these features are concatenated and then fed into an MLP layer to predict the ranking score:

$$v = [v_{label}; v_{type}; v_{utterance}; v_{session}; v_p]$$

$$s = MLP(v)$$

To train this model, we minimize the pair-wise hinge-loss, defined as:

$$l_r = max(0, \sigma + s^- - s^+)$$

where $s^+$ is the ranking score of the ground-truth entity and $s^-$ is the ranking score of a negative entity sampled from candidates other than the ground-truth. $\sigma$ is a constant margin and is set to 0.5.

3.4 Response Generation Model

Given the linked entities, we employ a transformer-based response generation model that is trained to leverage the context of a dialogue along with the knowledge relevant at a given turn. More specifically, we first fine-tune a GPT2-medium model using the Wizard of Wikipedia (WOW) dataset (Dinan et al., 2018). WOW is a suitable dataset for fine-tuning as it involves knowledge-grounded conversations dealing with Wikipedia articles, a data source we are using for entity linking in this work.

The GPT2 generation model is fine-tuned in a matter consistent with (Wolf et al., 2019b; Gopalakrishnan et al., 2020). During generation, we are provided a dialogue context, $C = \{c_1, c_2, ..., c_i \}$ containing utterances before $c_i$. We use our linked entities to query the relevant Wikipedia articles, and use the first paragraph of the returned articles, giving us a collection of knowledge sentences, $K = \{k_1, k_2, ..., k_n\}$.

Next, we truncate each knowledge sentence with more than 64 tokens and provide a concatenated input consisting of the dialogue context and the knowledge sentences. We then sample from the language model, one token at a time, using nucleus sampling to form our generated system response.

4 Experiment Setup

4.1 Datasets

We rely on Wikipedia and Wiki data\(^3\) to build the knowledge base for this task. We built a Knowledge Graph (KG) containing over 6M entities including attributes such as Wiki ID, title, type, and introduction. To perform NEL on conversational data, we collect a Multi-turn Open-domain Conversation Dataset (MOC) and ask crowd worker annotators to first annotate NER labels (entity mention and type), and then give ER labels – the ground truth Wikidata ID. Different from the entity labels in regular NER tasks, we define 50 entity types across 8 popular domains in open-domain conversations.

\(^3\)https://www.wikidata.org/wiki/, https://www.wikipedia.org/
including Fashion, Politics, Books, Sports, Music, Science/Technology, Game, Video/Movies. In addition, we created a synthetic dataset that contains ambiguous entities that can only be understood through dialog context. For example, in the utterance "I like Harry Potter", the model needs to understand the context of the utterance to figure out if the user is referring to the movie or the book. We also randomly selected some conversations from Wizard of Wikipedia (WoW), which is a collection of open-domain dialogues grounded on Wikipedia knowledge (Dinan et al., 2018). The statistics of the datasets we used are shown in Table 1.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th>Validation</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOC</td>
<td>5,962</td>
<td>662</td>
<td>1,111</td>
</tr>
<tr>
<td>Synthetic</td>
<td>8,150</td>
<td>905</td>
<td>2,896</td>
</tr>
<tr>
<td>WoW</td>
<td>1,248</td>
<td>216</td>
<td>340</td>
</tr>
</tbody>
</table>

Table 1: Number of utterances of the open-domain conversation data sets used in this study.

### 4.2 Model Setup

All models are implemented in Pytorch (Paszke et al., 2017). For the NER model, we initialize the word embedding with stacked embeddings, including Flair embeddings (Akbik et al., 2018) and FastText embeddings (Bojanowski et al., 2017). The sizes of the word embeddings and hidden state are 300 and 256, respectively. We adopt the SGD optimizer with an initial learning rate of 0.1 and decay rate of 0.5. The batch size is set to 16 and the maximum training epoch is set to 15 with an early stopping strategy. For the ER model, we use Adam as the optimizer and set the learning rate to 0.0005. The hidden size is 762 and the batch size is 8. The maximum sentence length in all the experiments is set to 128.

### 5 Results and Analysis

#### 5.1 NER Results

The performance of the NER models is evaluated using precision, recall and F-1. We consider both the span of an entity and its type. Table 2 shows the results of NER models on three datasets. To compare with our context-aware NER model, we use Flair as the baseline, which is a state-of-the-art NER model on benchmarks in several domains (Akbik et al., 2018). It shows that our context-aware model achieves the best performance on most metrics. In particular, we observe the largest gain of our model using contextual information on the synthetic dataset. This is because that data was created to contain more ambiguous entities and thus requires dialog context to determine entity types.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>P</th>
<th>R</th>
<th>F-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flair</td>
<td>MOC</td>
<td>-</td>
<td>2.7</td>
<td>1.2</td>
</tr>
<tr>
<td>Flair w/ context</td>
<td>MOC</td>
<td>-0.1</td>
<td>2.7</td>
<td>1.2</td>
</tr>
<tr>
<td>Flair</td>
<td>Synthetic</td>
<td>16.0</td>
<td>17.7</td>
<td>16.9</td>
</tr>
<tr>
<td>Flair w/ context</td>
<td>Synthetic</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Flair</td>
<td>WoW</td>
<td>0.7</td>
<td>1.8</td>
<td>1.2</td>
</tr>
<tr>
<td>Flair w/ context</td>
<td>WoW</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2: Results of NER models (relative gains compared to Flair in %).

#### 5.2 ER Results

For the ER task, we evaluate the recall@$n$ values ($n = 1, 3, 5$), which measures the ranking ability of the models. We compare our model with the following two baselines:

**Search.** After performing entity retrieval through Elasticsearch, we rank the candidate entities based on their popularity, i.e., the number of views in last 60 days.

**Ranking.** Similar to our method, here we only use entity and utterance-level matching scores, without dialog context in the ranking model.

Table 3 shows the ER results when ground-truth NER is provided as input. We can see that a ranking model can significantly improve the top entity relevance over the search baseline on all the three datasets. Compared to the non-context ranking model, our proposed context-aware model could further improve the results, especially for R@1.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>R@1</th>
<th>R@3</th>
<th>R@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>MOC</td>
<td>64.5</td>
<td>29.4</td>
<td>2.8</td>
</tr>
<tr>
<td>Rank</td>
<td></td>
<td>65.0</td>
<td>29.7</td>
<td>2.9</td>
</tr>
<tr>
<td>Rank w/ context</td>
<td>MOC</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Search</td>
<td>Synthetic</td>
<td>82.9</td>
<td>22.2</td>
<td>9.4</td>
</tr>
<tr>
<td>Rank</td>
<td></td>
<td>91.0</td>
<td>21.9</td>
<td>10.0</td>
</tr>
<tr>
<td>Rank w/ context</td>
<td>Synthetic</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Search</td>
<td>WoW</td>
<td>82.1</td>
<td>28.8</td>
<td>11.2</td>
</tr>
<tr>
<td>Rank</td>
<td></td>
<td>89.1</td>
<td>29.2</td>
<td>11.2</td>
</tr>
<tr>
<td>Rank w/ context</td>
<td>WoW</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3: Results of ER models (relative gains compared to baseline search in %) using ground-truth NER information.

#### 5.3 End-to-end NEL Results

In Section 5.2, the input of the ER task is the ground-truth NER results. In the practical scenario, the input is the prediction of the NER models.
Therefore, we also evaluate the performance of end-to-end NEL, where the predictions of NER models are used for ER. For performance metrics, we compare the predicted entity with the ground-truth one, and compute precision, recall and F-1. The results are shown in Table 5. Here we observe again that a ranking model can significantly improve results, and the context model yields further gain.

<table>
<thead>
<tr>
<th>NER</th>
<th>ER</th>
<th>Dataset</th>
<th>P</th>
<th>R</th>
<th>F-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flair</td>
<td>Search</td>
<td>MOC</td>
<td>62.1</td>
<td>59.7</td>
<td>60.6</td>
</tr>
<tr>
<td>Flair</td>
<td>Rank</td>
<td>MOC</td>
<td>62.9</td>
<td>63.4</td>
<td>62.8</td>
</tr>
<tr>
<td>Flair</td>
<td>Search</td>
<td>Synthetic</td>
<td>68.2</td>
<td>68.9</td>
<td>68.4</td>
</tr>
<tr>
<td>Flair</td>
<td>Rank</td>
<td>Synthetic</td>
<td>71.0</td>
<td>71.0</td>
<td>71.0</td>
</tr>
<tr>
<td>Flair</td>
<td>Search</td>
<td>WoW</td>
<td>62.0</td>
<td>62.0</td>
<td>62.1</td>
</tr>
<tr>
<td>Flair</td>
<td>Rank</td>
<td>WoW</td>
<td>75.4</td>
<td>72.8</td>
<td>73.9</td>
</tr>
</tbody>
</table>

Table 5: End-to-end experimental results (relative gains in % compared to the end-to-end model of Flair NER and baseline ER search).

5.4 Case Study

Table 4 shows NER and ER results for two example utterances along with their context. We can see when there is an ambiguity in the current utterance, our context-aware model can use context information to correctly recognize the entities and link them to the right entities in KB. In the first example, the named entity is correctly recognized by all the models, however, the model without context failed in the ER task because of insufficient information. In the second case, models without using context information recognize a wrong entity and then link it to a seemingly reasonable but not the most appropriate entity.

5.5 Response Generation Results

We generate outputs for 100 distinct conversational contexts in the WoW data set using using configurations: Baseline GPT2 and GPT2 with NEL. Here, we provide crowd-worker annotators the conversational context along with the generated response, without the associated knowledge extracted through linking. We then ask the workers to evaluate according to two metrics, appropriateness and informativeness, on an ordinal scale from 0-2.

Our results show that in the generated responses, GPT2 with NEL module is superior over baseline GPT2 on both the appropriateness and informativeness metrics, suggesting that our solution can better understand conversation context and is able to generate informative and appropriate responses.

<table>
<thead>
<tr>
<th>Model</th>
<th>Appropr.</th>
<th>Inform.</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPT2 w/ NEL</td>
<td>25.5</td>
<td>53.8</td>
</tr>
</tbody>
</table>

Table 6: Human evaluation of generated responses. (%, relative gains compared to GPT2)

6 Conclusion

In this paper, we investigate NEL in multi-turn open-domain conversations. Considering the characteristic of dialogs, where the meaning of the cur-
rent utterance often varies depending on the context, we design a context-aware NER model and an ER model. Experimental results on three datasets prove that using context information improves the entity recognition and resolution performance. Extrinsic evaluation on response generation also validates the effectiveness of the entity information.

References


Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get the point of my utterance! learning towards effective responses with multi-head attention mechanism. In IJCAI, pages 4418–4424.


