A Document-grounded Matching Network for Response Selection in Retrieval-based Chatbots

Xueliang Zhao[*1], **Chongyang Tao**[*1], **Wei Wu**[2], **Can Xu**[2], **Dongyan Zhao**[1,3] and **Rui Yan**[1,3†]
[1]Institute of Computer Science and Technology, Peking University, Beijing, China
[2]Microsoft Corporation, Beijing, China
[3]Center for Data Science, Peking University, Beijing, China
{xl.zhao,chongyangtao,zhaody,ruiyan}@pku.edu.cn, {wuwei,caxu}@microsoft.com

Abstract

We present a document-grounded matching network (DGMN) for response selection that can power a knowledge-aware retrieval-based chatbot system. The challenges of building such a model lie in how to ground conversation contexts with background documents and how to recognize important information in the documents for matching. To overcome the challenges, DGMN fuses information in a document and a context into representations of each other, and dynamically determines if grounding is necessary and importance of different parts of the document and the context through hierarchical interaction with a response at the matching step. Empirical studies on two public data sets indicate that DGMN can significantly improve upon state-of-the-art methods and at the same time enjoys good interpretability.

1 Introduction

Human-machine conversation is a long-standing goal of artificial intelligence. Recently, building a chatbot for open domain conversation has gained increasing interest due to both availabilities of a large amount of human conversation data and powerful models learned with neural networks. Existing methods are either retrieval-based or generation-based. Retrieval-based methods respond to human input by selecting a response from a pre-built index [Ji *et al.*, 2014; Yan and Zhao, 2018], while generation-based methods synthesize a response with a natural language model [Shang *et al.*, 2015; Li *et al.*, 2015]. In this work, we study the problem of response selection for retrieval-based chatbots, since retrieval-based systems are often superior to their generation-based counterparts on response fluency and diversity, are easy to evaluate.

A key step in response selection is measuring the matching degree between a context (a message with a few turns of conversation history) and a response candidate. Existing methods [Wu *et al.*, 2017; Zhou *et al.*, 2018b] have achieved impressive performance on benchmarks [Lowe *et al.*, 2015; Wu *et al.*, 2017], but responses are selected solely based on conversation history. Human conversations, on the other hand, are often grounded in external knowledge. For example, in Reddit, discussion among users is usually along the document posted at the beginning of a thread which provides topics and basic facts for the following conversation. Lack of knowledge grounding has become one of the major gaps between the current open domain dialog systems and real human conversations. As a step toward bridging the gap, we investigate knowledge-grounded response selection in this work and specify the knowledge as unstructured documents that are common sources in practice. The task is that given a document and a conversation context based on the document, one selects a response from a candidate pool that is consistent and relevant with both the conversation context and the background document. Table 1 shows an example from PERSONA-CHAT, a data set released recently by Facebook [Zhang *et al.*, 2018], to illustrate the task: given two speakers' profiles as documents and a conversation context, one is required to distinguish the true response from the false ones[1].

A's profile	trying new recipes makes me happy. i feel like i need to exercise more. i am an early bird , while my significant other is a night owl. i am a kitty owner.
B's profile	i might actually be a mermaid. i use all of my time for my education. i am very sociable and love those close to me. i enjoy swimming in the ocean , i feel in tune with its inhabitants.
Context	**A**: hi how are you today **B**: i am good . how are you ? **A**: pretty good where do you work ?
True response	i do not work , i am a full time student . what about you?
False response	i have been working as a salesman for more than 10 years.

Table 1: An example of document-grounded dialogue

Intuitively, both documents and conversation contexts should participate in matching. Since documents and contexts are highly asymmetric in terms of information they convey, and there exists complicated dependency among sentences in the documents and utterances in the contexts, challenges of the task include (1) how to ground conversation contexts with documents given that utterances in the contexts are not always related to the documents due to the casual nature of open domain conversation (e.g., the greetings in Table 1); (2) how to comprehend documents with conversation contexts when in-

[*]Equal Contribution.
[†]Corresponding author: Rui Yan (ruiyan@pku.edu.cn).
[1]For space limitation, we only show one false response here.

formation in the documents are rather redundant for proper response recognition (e.g., the description regarding to B's hobby in her profile in Table 1); and (3) how to effectively leverage both information sources to perform matching. To overcome the challenges, we propose a document-grounded matching network (DGMN). DGMN encodes sentences in a document, utterances in a conversation context, and a response candidate through self-attention, and models context grounding and document comprehension by constructing a document-aware context representation and a context-aware document representation via an attention mechanism. With the rich representations, DGMN distills matching information from each utterance-response pair and each sentence-response pair, where whether an utterance needs grounding, which parts of the document are crucial for grounding and matching, and which parts of the context are useful for representing the document are dynamically determined by a hierarchical interaction mechanism. The final matching score is defined as an aggregation of matching signals from all pairs.

We conduct experiments on two public data sets: the PERSONA-CHAT data [Zhang *et al.*, 2018] and the CMU Document Grounded Conversation (CMUDoG) data [Zhou *et al.*, 2018a]. Evaluation results indicate that on both data sets, DGMN can significantly outperform state-of-the-art methods. Compared with Transformer, the best performing baseline on both data, absolute improvements from DGMN on $r@1$ (hits@1) are more than 13% on the PERSONA-CHAT data and more than 5% on the CMUDoG data. Through both quantitative and qualitative analysis, we also demonstrate the effect of different representations to matching and how DGMN grounds conversation contexts with documents.

Our contributions in this work are three-fold: (1) proposal of a document-grounded matching network that performs response selection according to both conversation contexts and background knowledge; (2) empirical verification of the effectiveness of the proposed model on two public data sets; and (3) new state-of-the-art on the PERSONA-CHAT data without any pre-training on external resources.

2 Document-Grounded Matching Network

In this section, we first formalize the document-grounded matching problem, and then introduce our model from an overview to details of components.

2.1 Problem Formalization

Suppose that we have a data set $\mathcal{D} = \{(D_i, c_i, y_i, r_i)\}_{i=1}^{N}$ where $D_i = \{d_{i,1}, \cdots, d_{i,m_i}\}$ is a document that serves as background knowledge for conversation with $d_{i,k}$ the k-th sentence, $c_i = \{u_{i,1}, \cdots, u_{i,n_i}\}$ is a conversation context following D_i with $u_{i,k}$ the k-th utterance, r_i is a response candidate, and $y_i \in \{0, 1\}$ is a label with $y_i = 1$ indicating that r_i is a proper response given c_i and D_i, otherwise $y_i = 0$. The task is to learn a matching model $g(\cdot, \cdot, \cdot)$ from $\mathcal{D}$, and thus for a new triple (D, c, r), $g(D, c, r)$ returns the matching degree between c and r under D.

2.2 Model Overview

We define $g(D, c, r)$ as a document-grounded matching network. Figure 1 illustrates the architecture of the model. In brief, DGMN consists of an encoding layer, a fusion layer, a matching layer, and an aggregation layer. The encoding layer represents D, c, and r via self-attention, and feeds the representations to the fusion layer where D and c are fused into the representations of each other as a document-aware context representation and a context-aware document representation. Based on the representations given by the first two layers, the matching layer then lets each utterance in c and each sentence in D interact with r, and distills matching signals from the interaction. Matching signals in all pairs are finally aggregated as a matching score in the aggregation layer.

2.3 Model Details

We elaborate each layer of the document-grounded matching network in this section.

Encoding Layer

Given an utterance u_i in a context c, a sentence d_j in a document D, and a response candidate r, the model first embeds u_i, d_i, and r as $\mathbf{E}_{u_i} = [\mathbf{e}_{u_i,1}, \cdots, \mathbf{e}_{u_i,l_u}]$, $\mathbf{E}_{d_j} = [\mathbf{e}_{d_j,1}, \cdots, \mathbf{e}_{d_j,l_d}]$ and $\mathbf{E}_r = [\mathbf{e}_{r,1}, \cdots, \mathbf{e}_{r,l_r}]$ respectively by looking up a shared embedding table pre-trained with Glove [Pennington *et al.*, 2014] on the training data $\mathcal{D}$, where $\mathbf{e}_{u_i,k}$, $\mathbf{e}_{d_j,k}$ and $\mathbf{e}_{r,k}$ are representations of the k-th words in u_i, d_j and r respectively, and l_u, l_r, and l_d are lengths of the three sequences. $\mathbf{E}_{u_i}$, $\mathbf{E}_{d_j}$ and $\mathbf{E}_r$ are then processed by an attentive module to encode long-term dependency among words into the representations.

The attentive module simplifies the multi-head attention module in Transformer [Vaswani *et al.*, 2017], and consists of a scaled dot-product attention component and a feed-forward component. Without loss of generality, let $\mathbf{Q} \in \mathbb{R}^{n_Q \times d}$, $\mathbf{K} \in \mathbb{R}^{n_K \times d}$, and $\mathbf{V} \in \mathbb{R}^{n_V \times d}$ denote embedding matrices of a query, a key, and a value respectively, where n_Q, n_K, and n_V are numbers of words in the input sequences, and d stands for embedding size. The scaled dot-product attention component is then defined as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\mathbf{Q}\mathbf{K}^T/\sqrt{d})\mathbf{V}. \quad (1)$$

Intuitively, each entry of $\mathbf{V}$ is weighted by a relevance score defined by the similarity of an entry of $\mathbf{Q}$ and an entry of $\mathbf{K}$, and then an updated representation of $\mathbf{Q}$ is formed by linearly combining the entries of $\mathbf{V}$ with the weights. In practice, we often let $\mathbf{K} = \mathbf{V}$, and thus $\mathbf{Q}$ is represented by similar entries of $\mathbf{V}$. The feed-forward component takes $\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ as input, and transforms it to a new representation by two non-linear projections. A residual connection [He *et al.*, 2016] and a row-wise normalization [Ba *et al.*, 2016] are applied to the result of each projection. For ease of presentation, we denote the whole attentive module as $f_{\text{ATT}}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$. u_i, d_j and r are then represented by attending to themselves through $f_{\text{ATT}}(\cdot, \cdot, \cdot)$:

$$\mathbf{U}_i = f_{\text{ATT}}(\mathbf{E}_{u_i}, \mathbf{E}_{u_i}, \mathbf{E}_{u_i}) \quad (2)$$

$$\mathbf{D}_j = f_{\text{ATT}}(\mathbf{E}_{d_j}, \mathbf{E}_{d_j}, \mathbf{E}_{d_j}) \quad (3)$$

$$\mathbf{R} = f_{\text{ATT}}(\mathbf{E}_r, \mathbf{E}_r, \mathbf{E}_r). \quad (4)$$

Fusion Layer

The fusion layer grounds the conversation context by the document and fuses the information of the context into the docu-

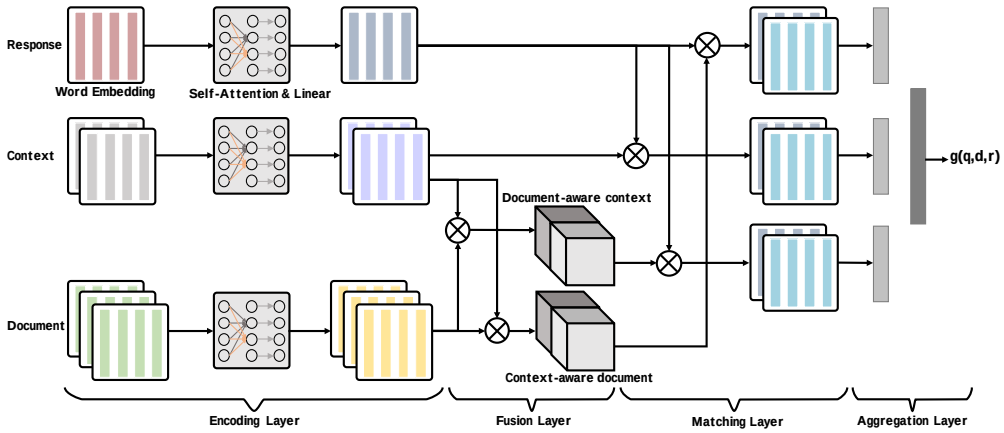


Figure 1: Architecture of the document-grounded matching network.

ment, which results in a document-aware context representation and a context-aware document representation. Formally, the document-aware representation of u_i is given by $\hat{\mathbf{U}}_i = [\hat{\mathbf{U}}_{i,1}, \cdots, \hat{\mathbf{U}}_{i,m}]$, where m is the number of sentences in the document, and $\forall j \in \{1, \ldots, m\}$, $\hat{\mathbf{U}}_{i,j}$ can be formulated as

$$\hat{\mathbf{U}}_{i,j} = f_{\text{ATT}}(\mathbf{U}_i, \mathbf{D}_j, \mathbf{D}_j). \tag{5}$$

Similarly, the context-aware representation of d_j is defined as $\hat{\mathbf{D}}_j = [\hat{\mathbf{D}}_{j,1}, \cdots, \hat{\mathbf{D}}_{j,n}]$, where n is the number of utterances in the context, and $\forall i \in \{1, \ldots, n\}$, $\hat{\mathbf{D}}_{j,i}$ is calculated by

$$\hat{\mathbf{D}}_{j,i} = f_{\text{ATT}}(\mathbf{D}_j, \mathbf{U}_i, \mathbf{U}_i). \tag{6}$$

In $\hat{\mathbf{U}}_{i,j}$, information in d_j provides grounding to u_i, and correlations between d_j and u_i will be distilled to enhance the original representation of u_i. The grounding is performed on a sentence-level rather than on a document-level (i.e., attention with a document vector). This is motivated by the intuition that sentences in a document are differentially important to represent the semantics of an utterance in a context, and the importance should be dynamically recognized through interaction with a response in the matching step. In a similar sense, by letting d_j attend to u_i in $\hat{\mathbf{D}}_{j,i}$ we attempt to highlight important parts of d_j through their correlation with u_i, and thus achieve better document understanding in matching.

As we have analyzed before, utterances in a context are not always related to the background document in chat. To model this intuition, we append $\mathbf{U}_i$ to $\hat{\mathbf{U}}_i$ as $\tilde{\mathbf{U}}_i = [\mathbf{U}_i, \hat{\mathbf{U}}_{i,1}, \cdots, \hat{\mathbf{U}}_{i,m}]$ and determine if an utterance needs grounding with the guide of response r in the following matching layer. Ideally, if an utterance does not need grounding, then only $\mathbf{U}_i$ should participate in matching since other entries of $\tilde{\mathbf{U}}_i$ are noisy. The weights of the entries of $\tilde{\mathbf{U}}_i$ will be learned from training data.

Matching Layer

The matching layer pairs $\mathbf{U}_i$, $\tilde{\mathbf{U}}_i$, $\hat{\mathbf{D}}_j$ with $\mathbf{R}$ as $\{\mathbf{U}_i, \mathbf{R}\}$, $\{\tilde{\mathbf{U}}_i, \mathbf{R}\}$ and $\{\hat{\mathbf{D}}_j, \mathbf{R}\}$ respectively, and extracts matching information from the pairs. Different from existing matching models that are solely based on conversation contexts, $\tilde{\mathbf{U}}_i$ and $\hat{\mathbf{D}}_j$ now contain grounding information from multiple sentences (utterances). Thus, the model needs to dynamically select important sentences (utterances) for grounding and even determine if grounding is necessary. To tackle the new challenges, we propose a hierarchical interaction mechanism. Take $\{\tilde{\mathbf{U}}_i, \mathbf{R}\}$ as an example. For ease of presentation, we define $\mathbf{U}_i = \hat{\mathbf{U}}_{i,0}$. Let $\mathbf{r}_j$ denote the j-th entry of $\mathbf{R}$, then the first level interaction of $\tilde{\mathbf{U}}_i$ and $\mathbf{R}$ happens between $\mathbf{r}_j$ and each $\hat{\mathbf{U}}_{i,k}$, $\forall k \in \{0, \ldots, m\}$, and transforms $\hat{\mathbf{U}}_{i,k}$ into $h_{i,j,k}$ through

$$\omega_{i,j,k,t} = \mathbf{v}_a^\top \tanh(\mathbf{w}_a[\hat{\mathbf{u}}_{i,k,t}; \mathbf{r}_j] + \mathbf{b}_a), \tag{7}$$

$$\alpha_{i,j,k,t} = \frac{\exp(\omega_{i,j,k,t})}{\sum_{t=1}^{l_u} \exp(\omega_{i,j,k,t})}, \tag{8}$$

$$h_{i,j,k} = \sum\nolimits_{t=1}^{l_u} \alpha_{i,j,k,t} \hat{\mathbf{u}}_{i,k,t}, \tag{9}$$

where $\hat{\mathbf{u}}_{i,k,t}$ is the t-th entry of $\hat{\mathbf{U}}_{i,k}$, and $\mathbf{w}_a$, $\mathbf{v}_a$, and $\mathbf{b}_a$ are parameters. Through Eq. (9), the first level interaction tries to play emphasis on important words in each $\hat{\mathbf{U}}_{i,k}$ with respect to $\mathbf{r}_j$. The second level interaction of $\tilde{\mathbf{U}}_i$ and $\mathbf{R}$ then summarizes $[h_{i,j,0}, \ldots, h_{i,j,m}]$ as $h_{i,j}$ by

$$\omega'_{i,j,k} = \mathbf{v'}_a^\top \tanh(\mathbf{w'}_a[h_{i,j,k}; \mathbf{r}_j] + \mathbf{b'}_a), \tag{10}$$

$$\alpha'_{i,j,k} = \frac{\exp(\omega'_{i,j,k})}{\sum_{k=0}^{m} \exp(\omega'_{i,j,k})}, \tag{11}$$

$$h_{i,j} = \sum\nolimits_{k=0}^{m} \alpha'_{i,j,k} h_{i,j,k}, \tag{12}$$

where $\mathbf{w'}_a$, $\mathbf{v'}_a$, and $\mathbf{b'}_a$ are parameters. In the second level interaction, sentences in the document that can bring valuable grounding information for matching will play an important role in the formation of $h_{i,j}$. As a special case, when $\alpha'_{i,j,0}$ is much bigger than other weights, the model judges that u_i does not need grounding from the document. Finally, matching information between $\tilde{\mathbf{U}}_i$ and $\mathbf{R}$ is stored in a matrix $\tilde{\mathbf{M}}_i = [\mathbf{m}_{i,1}, \cdots, \mathbf{m}_{i,l_r}]$. $\forall j \in \{1, \ldots, l_r\}$, $\mathbf{m}_{i,j}$ is calcu-

lated by

$$\mathbf{m}_{i,j} = \text{ReLU}(\mathbf{w}_p \begin{bmatrix} (h_{i,j} - \mathbf{r}_j) \odot (h_{i,j} - \mathbf{r}_j) \\ h_{i,j} \odot \mathbf{r}_j \end{bmatrix} + \mathbf{b}_p), \quad (13)$$

where $\mathbf{w}_p$ and $\mathbf{b}_p$ are parameters, and $\odot$ refers to element-wise multiplication.

Following the same procedure, we obtain $\hat{\mathbf{M}}_j$ as a matching matrix for $\{\hat{\mathbf{D}}_j, \mathbf{R}\}$ where utterances in the context that are helpful for representing d_j are highlighted by r. Since $\mathbf{U}_i$ is only made up of word representations (i.e., one-layer structure), the matching matrix $\mathbf{M}_i$ for $\{\mathbf{U}_i, \mathbf{R}\}$ is calculated by one level interaction parameterized in a similar way as Eq. (7)-(9) and the same function as Eq. (13).

Aggregation Layer and Learning Method

The aggregation layer accumulates matching signals in $\{\mathbf{M}_i\}_{i=1}^n$, $\{\tilde{\mathbf{M}}_i\}_{i=1}^n$, and $\{\hat{\mathbf{M}}_j\}_{j=1}^m$ as a matching score for (D, c, r). Specifically, we construct a tensor from $\{\mathbf{M}_i\}_{i=1}^n$, and then apply a convolutional neural network [Ji *et al.*, 2010] to the tensor to calculate a matching vector $\mathbf{t}$. Similarly, we have matching vectors $\hat{\mathbf{t}}$ and $\tilde{\mathbf{t}}$ for $\{\hat{\mathbf{M}}_j\}_{j=1}^m$ and $\{\tilde{\mathbf{M}}_i\}_{i=1}^n$, respectively. The matching function $g(D, c, r)$ is defined as

$$g(D, c, r) = \sigma([\mathbf{t}; \hat{\mathbf{t}}; \tilde{\mathbf{t}}]\mathbf{w}_o + \mathbf{b}_o), \quad (14)$$

where $\mathbf{w}_o$ and $\mathbf{b}_o$ are wights, and $\sigma(\cdot)$ is a sigmoid function.

Parameters of $g(D, c, r)$ are estimated from the training data $\mathcal{D}$ by minimizing the following objective:

$$-\sum_{i=1}^{N} \Big(y_i \log g(D_i, c_i, r_i) + (1 - y_i) \log(1 - g(D_i, c_i, r_i)) \Big). \quad (15)$$

3 Experiments

We test our model on two public data sets.

3.1 Experimental Setup

The first data we use is the PERSONA-CHAT data set published in [Zhang *et al.*, 2018]. The data is collected by requiring two workers on Amazon Mechanical Turk to chat with each other according to their assigned profiles. Each profile is presented in a form of a document with an average of 4.49 sentences. The profiles define speakers' personas and provide characteristic knowledge for dialogues. For each dialogue, there are both original profiles and revised profiles that are rephrased from the original ones by other crowd workers to force models to learn more than simple word overlap. A revised profile shares the same number of sentences with its original one, and on average, there are 7.33 words per sentence in the original profiles and 7.32 words per sentence in the revised ones. The data is split as a training set, a validation set, and a test set by the publishers. In all the three sets, 7 turns before an utterance are used as conversation history, and the next turn of the utterance is treated as a positive response candidate. Besides, each utterance is associated with 19 negative response candidates that are randomly sampled by the publishers. More statistics of the three sets are shown in Table 2. Following the insights in [Zhang *et al.*, 2018], we train models using revised profiles and test the models with both original and revised profiles.

Statistics	PERSONA-CHAT			CMUDoG		
	Train	Val	Test	Train	Val	Test
# of conversations	8939	1000	968	2881	196	537
# of turns	65719	7801	7512	36159	2425	6637
Av_turns / conversation	7.35	7.80	7.76	12.55	12.37	12.36
Av_length of utterance	11.67	11.94	11.79	18.64	20.06	18.11

Table 2: Statistics of the two data sets.

In addition to PERSONA-CHAT, we also conduct experiments with CMUDoG data set published recently in [Zhou *et al.*, 2018a]. Conversations in the data are collected from workers on Amazon Mechanical Turk and are based on movie-related wiki articles in two scenarios. In the first scenario, only one worker has access to the provided document, and he/she is responsible for introducing the movie to the other worker; while in the second scenario, both workers know the document and they are asked to discuss the content of the document. Since the data size for an individual scenario is small, we merge the data of the two scenarios in the experiments and filter out conversations less than 4 turns to avoid noise. Each document consists of 4 sections and these sections are shown to the workers one by one every 3 turn (the first section lasts 6 turns due to initial greetings). On average, each section contains 8.22 sentences and 27.86 words per sentence. The data has been divided into a training set, a validation set, and a test set by the publishers. In each set, we take 2 turns before an utterance as conversation history and the next turn of the utterance as a positive response candidate. Since the data does not contain negative examples, we randomly sample 19 negative response candidates for each utterance from the same set. Detailed statistics of the data is given in Table 2.

We employ $r@k$ as evaluation metrics where $k \in \{1, 2, 5\}$. For a single context, if the only positive candidate is ranked within top k positions, then $r@k = 1$, otherwise, $r@k = 0$. The final value of the metric is an average over all contexts in test data. Note that in PERSONA-CHAT, $r@1$ is equivalent to hits@1 which is the metric used by [Zhang *et al.*, 2018] for model comparison.

3.2 Baseline Models

The following models are selected as baselines. These models are the ranking models in [Zhang *et al.*, 2018] and [Mazare *et al.*, 2018] which perform much better than the generative models in [Zhang *et al.*, 2018] on the PERSONA-CHAT data.

Starspace. A supervised model in [Wu *et al.*, 2018] that learns the similarity between a conversation context and a response candidate by optimizing task-specific embedding via the margin ranking loss. The similarity is measured by the cosine of the sum of word embeddings. Documents are concatenated to conversation contexts.

Profile Memory. The model in [Zhang *et al.*, 2018] that lets a conversation context attend over the associated document to produce a vector which is then combined with the context. Cosine is used to measure the similarity between the output context representation and a response candidate.

KV Profile Memory. The best performing model in [Zhang *et al.*, 2018] which considers keys as dialogue history and values as the next dialogue utterances and uses a conversation

Models \ Metrics	PERSONA-CHAT						CMUDoG		
	Original Persona			Revised Persona					
	$r@1$	$r@2$	$r@5$	$r@1$	$r@2$	$r@5$	$r@1$	$r@2$	$r@5$
Starspace [Wu *et al.*, 2018]	49.1	60.2	76.5	32.2	48.3	66.7	50.7	64.5	80.3
Profile Memory [Zhang *et al.*, 2018]	50.9	60.7	75.7	35.4	48.3	67.5	51.6	65.8	81.4
KV Profile Memory [Zhang *et al.*, 2018]	51.1	61.8	77.4	35.1	45.7	66.3	56.1	69.9	82.4
Transformer [Mazare *et al.*, 2018]	54.2	68.3	83.8	42.1	56.5	75.0	60.3	74.4	87.4
DGMN	**67.6**	**80.2**	**92.9**	**58.8**	**62.5**	**87.7**	**65.6**	**78.3**	**91.2**
DGMN($\mathbf{t}$)	51.8	66.1	83.3	51.8	66.1	83.3	55.6	69.4	85.4
DGMN($\mathbf{t}$+$\tilde{\mathbf{t}}$)	66.3	78.9	91.7	57.0	71.2	86.9	64.5	78.2	90.8
DGMN($\mathbf{t}$+$\tilde{\mathbf{t}}$-NoGround)	64.2	77.8	91.3	55.8	70.1	86.2	63.5	76.8	90.8

Table 3: Evaluation results on the test sets of the PERSONA-CHAT data and the CMUDoG data. Numbers in bold mean that improvement over the best baseline is statistically significant (t-test, p-value < 0.01).

context as input to perform attention over the keys in addition to the documents. The past dialogues are stored in memory to help influence the prediction for the current conversation.

Transformer. A variant of the model proposed by [Vaswani *et al.*, 2017] for machine translation. The model exhibits state-of-the-art performance on the PERSONA-CHAT data as reported in [Mazare *et al.*, 2018].

All baseline models are implemented with the code shared at https://github.com/facebookresearch/ParlAI/tree/master/projects/personachat and tuned on the validation sets. We make sure that the baselines achieve the performance on the PERSONA-CHAT data as reported in [Zhang *et al.*, 2018] and [Mazare *et al.*, 2018]. Note that we do not include models pre-trained from large-scale external resources, such as the FT-PC model in [Mazare *et al.*, 2018], as baselines, since the comparison is unfair. On the other hand, it is interesting to study if pre-train the proposed model on those large-scale external data (e.g., the Reddit data in [Mazare *et al.*, 2018] with over 5 million personas spanning more than 700 million conversations) can further improve its performance. We leave the study as future work.

3.3 Implementation Details

We set the size of word embedding as 300. In PERSONA-CHAT, the number of sentences per document is limited to 5 (i.e., $m \leq 5$). For each sentence in a document, each utterance in a context, and each response candidate, if the number of words is less than 20, we pad zeros, otherwise, we keep the latest 20 words (i.e., $l_u = l_r = l_d = 20$). In CMUDoG, we set $m \leq 20$ and $l_u = l_r = l_d = 40$ following the same procedure. In the matching layer of DGMN, the number of filters of CNN is set as 16, and the window sizes of convolution and pooling are both 3. All models are learned using Adam [Kingma and Ba, 2015] optimizer with a learning rate of 0.0001. In training, we choose 32 as the size of mini-batches. Early stopping on validation data is adopted as a regularization strategy.

3.4 Evaluation Results

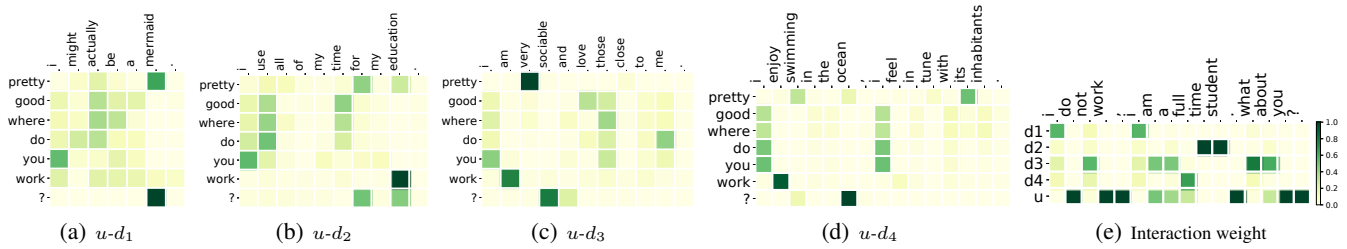
Table 3 reports evaluation results on the two data sets. We can see that on both data sets, DGMN outperforms all baselines over all metrics, and the improvement is statistically significant (t-test, p-value < 0.01). Improvement from DGMN over Transformer on the CMUDoG data is smaller than that on the PERSONA-CHAT data. The reason might be that Transformer can benefit from the wiki documents in CMUDoG that are longer and contain richer semantics than those handcrafted ones in PERSONA-CHAT.

3.5 Discussions

In this section, we investigate how different representations affect the performance of DGMN by an ablation study, visualize the example in Table 1 to illustrate how contexts are grounded by documents in DGMN, and check how the performance of DGMN changes with respect to document length.

Ablation study. First, we calculate a matching score only with the self-attention based context representation and response representation and denote the model as DGMN($\mathbf{t}$) which means only $\mathbf{t}$ is kept in Eq. (14). Then, we take the document-aware context representation into account, and denote the model as DGMN($\mathbf{t}$+$\tilde{\mathbf{t}}$) in which both $\mathbf{t}$ and $\tilde{\mathbf{t}}$ are used in Eq. (14). Based on $\mathbf{t}$+$\tilde{\mathbf{t}}$, we further examine if the special configuration for utterances that do not need grounding matters to the performance of DGMN by removing $\mathbf{U}_i$ from $\tilde{\mathbf{U}}_i$. The model is denoted as DGMN($\mathbf{t}$+$\tilde{\mathbf{t}}$-NoGround). Finally, the context-aware document representation is considered, and we have the full model of DGMN. Table 3 reports evaluation results on the two data sets. We can conclude that (1) all representations are useful for matching; (2) some effect of the context-aware document representation might be covered by the document-aware context representation, as adding the former after the latter does not bring much gain; and (3) although simple, the special configuration for utterances that do not need grounding cannot be removed from DGMN.

Visualization. Second, to further understand how DGMN performs context grounding, we visualize the attention weights in formation of the document-aware context representation (i.e., $\hat{\mathbf{U}}_{i,j}$) and the weights in the second level of interaction (i.e., $\alpha'_{i,j,k}$ in Eq. (11)) with the example in Table 1 in Introduction. Due to space limitation, we only visualize the last utterance of the context. Figure 2 shows the results. It is interesting to see that words like "work" and "education" are highly correlated in the graph, and at the same time, weights between the utterance and irrelevant sentences in the profile, such as "I am very social and love those close to me", are generally small. Moreover, in the second level interaction, while most function words and punctuation point to the utterance itself (i.e., u), the word "student" indicates that information from "i use all of my time for my education." is useful



(a) u-d_1 (b) u-d_2 (c) u-d_3 (d) u-d_4 (e) Interaction weight

Figure 2: Visualization of context grounding. The first four graphs illustrate attention between the last utterance of the context and each sentence in the document. The last one shows $\alpha'_{i,j,k}$ in interaction.

	Original Persona				Revised Persona				CMUDoG			
Doc Length	(0,25]	(25,30]	(30,35]	(35,52]	(0,30]	(30, 35]	(35, 40]	(40,55]	(0, 150]	(150, 250]	(250, 350]	(350,515]
Case Number	1019	2099	2197	2197	2419	2161	1558	1374	1921	2528	980	1208
r@1	67.1	68.8	67.5	66.7	57.8	59.7	59.1	59.0	64.0	65.8	66.2	67.4

Table 4: Performance of DGMN across different length of grounded documents on all data sets.

to recognize the relationship between the response candidate and the context. The example explains why DGMN works well from one perspective.

Performance analysis in terms of document length. Finally, we study the relationship between the performance of DGMN and document length by binning text examples in both data into different buckets according to the document length. Table 4 reports the evaluation results. On the PERSONA-CHAT data, both short profiles and long profiles lead to performance drop, while on the CMUDoG data, the longer the documents are, the better the performance of DGMN is. The reason behind the difference might be that profiles in the PERSONA-CHAT data are handcrafted by crowd workers, and thus semantics among different sentences are relatively independent, while documents in the CMUDoG data come from Wikipedia, and there is rich semantic overlap among sentences. Therefore, short profiles contain less useful information and long profiles contain more irrelevant information, and both will make the matching task more challenging. On the other hand, the longer a wiki document is, the more relevant information it can provide to the matching task.

4 Related Work

There are two groups of methods for building a chatbot. The first group learns response generation models under an encoder-decoder framework [Shang *et al.*, 2015; Vinyals and Le, 2015] with extensions to suppress generic responses [Li *et al.*, 2015; Mou *et al.*, 2016; Xing *et al.*, 2017; Tao *et al.*, 2018]. The second group learns a matching model of a human input and a response candidate for response selection. Along this line, early work assumes that the input is a single message [Wang *et al.*, 2013; Hu *et al.*, 2014]. Recently, conversation history is taken into account in matching. Representative methods include the dual LSTM model [Lowe *et al.*, 2015], the deep learning to respond architecture [Yan *et al.*, 2016], the multi-view matching model [Zhou *et al.*, 2016], the sequential matching network [Wu *et al.*, 2017], the deep attention matching network [Zhou *et al.*, 2018b], and the multi-representation fusion network [Tao *et al.*, 2019]. Our work belongs to the second group. The major difference we make is that in addition to conversation contexts, we also incorporate external documents as a kind of background knowledge into matching.

Before us, a few recent studies have considered grounding open domain dialogues with external knowledge. For example, Ghazvininejad *et al.* (2018) generalize the vanilla Seq2seq model by conditioning responses on both conversation history and external "facts". Zhang *et al.* (2018) release a persona-based conversation data set where profiles created by crowd workers constrain speakers' personas in conversation. Mazare *et al.* (2018) further increase the scale of the persona-chat data with conversations extracted from Reddit. Zhou *et al.* (2018a) publish a data set in which conversations are grounded in movie-related articles from Wikipedia. Dinan *et al.* (2018) release another document-grounded data set with wiki articles covering broader topics. In this work, we study grounding retrieval-based open domain dialog systems with background documents and focus on building a powerful matching model with advanced neural architectures. On the persona-chat data published in Zhang *et al.* (2018) and the document-grounded conversation data set published in Zhou *et al.* (2018a), the model improves upon state-of-the-art methods with large margins.

5 Conclusions

We propose a document-grounded matching network to incorporate external knowledge into response selection for retrieval-based chatbots. Experimental results on two public data sets consistently show that the proposed model can significantly outperform state-of-the-art methods.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2017YFC0804001), the National Science Foundation of China (NSFC Nos. 61672058 and 61876196).

References

[Ba *et al.*, 2016] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[Dinan *et al.*, 2018] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*, 2018.

[Ghazvininejad *et al.*, 2018] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. A knowledge-grounded neural conversation model. In *AAAI*, 2018.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[Hu *et al.*, 2014] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. Convolutional neural network architectures for matching natural language sentences. In *NIPS*, pages 2042–2050, 2014.

[Ji *et al.*, 2010] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. In *ICML*, pages 495–502, 2010.

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

[Kingma and Ba, 2015] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[Li *et al.*, 2015] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *NAACL*, pages 110–119, 2015.

[Lowe *et al.*, 2015] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *SIGDIAL*, pages 285–294, 2015.

[Mazare *et al.*, 2018] Pierre-Emmanuel Mazare, Samuel Humeau, Martin Raison, and Antoine Bordes. Training millions of personalized dialogue agents. In *EMNLP*, pages 2775–2779, 2018.

[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. In *COLING*, pages 3349–3358, 2016.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

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

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

[Tao *et al.*, 2019] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In *WSDM*, pages 267–275, 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

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

[Wang *et al.*, 2013] Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. A dataset for research on short-text conversations. In *EMNLP*, pages 935–945, 2013.

[Wu *et al.*, 2017] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *ACL*, pages 496–505, 2017.

[Wu *et al.*, 2018] Ledell Yu Wu, Adam Fisch, Sumit Chopra, Keith Adams, Antoine Bordes, and Jason Weston. Starspace: Embed all the things! In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[Xing *et al.*, 2017] Chen Xing, Wei Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. Topic aware neural response generation. In *AAAI*, pages 3351–3357, 2017.

[Yan and Zhao, 2018] Rui Yan and Dongyan Zhao. Coupled context modeling for deep chit-chat: towards conversations between human and computer. In *SIGKDD*, pages 2574–2583. ACM, 2018.

[Yan *et al.*, 2016] 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.

[Zhang *et al.*, 2018] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*, 2018.

[Zhou *et al.*, 2016] Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. Multi-view response selection for human-computer conversation. In *EMNLP*, pages 372–381, 2016.

[Zhou *et al.*, 2018a] Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. A dataset for document grounded conversations. *arXiv preprint arXiv:1809.07358*, 2018.

[Zhou *et al.*, 2018b] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*, pages 1118–1127, 2018.