Detecting Speaker Personas from Conversational Texts

Jia-Chen Gu[1], **Zhen-Hua Ling**[1]*, **Yu Wu**[2], **Quan Liu**[1,3], **Zhigang Chen**[3], **Xiaodan Zhu**[4]
[1]National Engineering Laboratory for Speech and Language Information Processing,
University of Science and Technology of China, Hefei, China
[2]Microsoft Research Asia, Beijing, China
[3]State Key Laboratory of Cognitive Intelligence, iFLYTEK Research, Hefei, China
[4]ECE & Ingenuity Labs, Queen's University, Kingston, Canada
gujc@mail.ustc.edu.cn, {zhling, quanliu}@ustc.edu.cn,
Wu.Yu@microsoft.com, zgchen@iflytek.com, xiaodan.zhu@queensu.ca

Abstract

Personas are useful for dialogue response prediction. However, the personas used in current studies are pre-defined and hard to obtain before a conversation. To tackle this issue, we study a new task, named Speaker Persona Detection (SPD), which aims to detect speaker personas based on the plain conversational text. In this task, a best-matched persona is searched out from candidates given the conversational text. This is a *many-to-many* semantic matching task because both contexts and personas in SPD are composed of multiple sentences. The long-term dependency and the dynamic redundancy among these sentences increase the difficulty of this task. We build a dataset for SPD, dubbed as `Persona Match on Persona-Chat (PMPC)`. Furthermore, we evaluate several baseline models and propose utterance-to-profile (U2P) matching networks for this task. The U2P models operate at a fine granularity which treat both contexts and personas as sets of multiple sequences. Then, each sequence pair is scored and an interpretable overall score is obtained for a context-persona pair through aggregation. Evaluation results show that the U2P models outperform their baseline counterparts significantly.

1 Introduction

Recently, human-machine conversation has achieved great success (Lowe et al., 2015; Serban et al., 2016; Wu et al., 2017; Madotto et al., 2018; Tao et al., 2019; Zhang et al., 2020; Roller et al., 2021), and has been applied to various scenarios, such as customer service and conversational recommendation engine. It is well-known that a user's persona can help the machine to generate more appropriate responses. Many studies investigate how to predict a response if a persona is given. Zhang et al. (2018); Liu et al. (2020); Song et al. (2021) built dialogue agents to perceive the user's persona and then generated personalized responses. Mazare et al. (2018); Gu et al. (2019); Hua et al. (2020); Gu et al. (2020); Zhu et al. (2021) built matching networks to select a response matching not only the conversational context, but also the user's persona.

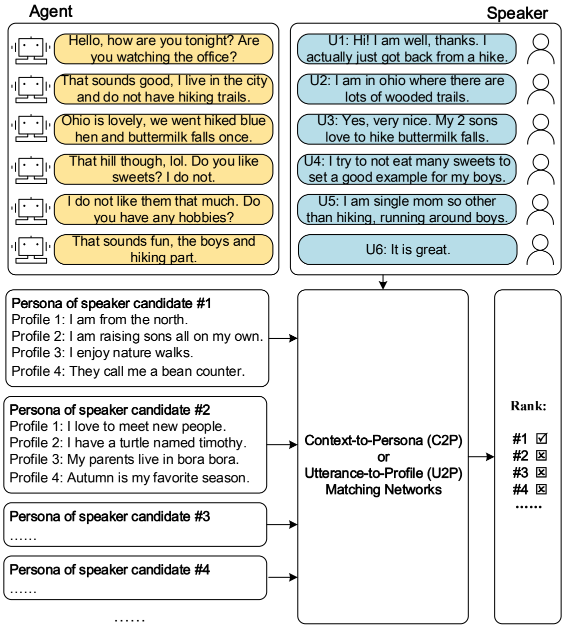

Agent
Speaker
Hello, how are you tonight? Are you watching the office?
That sounds good, I live in the city and do not have hiking trails.
Ohio is lovely, we went hiked blue hen and buttermilk falls once.
That hill though, lol. Do you like sweets? I do not.
I do not like them that much. Do you have any hobbies?
That sounds fun, the boys and hiking part.
U1: Hi! I am well, thanks. I actually just got back from a hike.
U2: I am in ohio where there are lots of wooded trails.
U3: Yes, very nice. My 2 sons love to hike buttermilk falls.
U4: I try to not eat many sweets to set a good example for my boys.
U5: I am single mom so other than hiking, running around boys.
U6: It is great.
Persona of speaker candidate #1
Profile 1: I am from the north.
Profile 2: I am raising sons all on my own.
Profile 3: I enjoy nature walks.
Profile 4: They call me a bean counter.
Persona of speaker candidate #2
Profile 1: I love to meet new people.
Profile 2: I have a turtle named timothy.
Profile 3: My parents live in bora bora.
Profile 4: Autumn is my favorite season.
Persona of speaker candidate #3
……
Persona of speaker candidate #4
……
……
Context-to-Persona (C2P) or Utterance-to-Profile (U2P) Matching Networks
Rank:
#1 ☑
#2 ☒
#3 ☒
#4 ☒
……

Figure 1: Illustration of our proposed SPD task. The matching network judges whether a persona candidate matches with the conversational texts of the speaker.

However, these personas are pre-defined and difficult to obtain before a conversation, because the speaker might not want to fill out a specific table to show its persona due to privacy issues. Hence, the cold-start problem may hinder the persona-aware response prediction in practice. To tackle this issue, our intuition is that the personal information such as hobbies or occupations may be mentioned explicitly or implicitly during a conversation, which can be utilized to identify the speaker's persona. If we can get the persona from

[*] Corresponding author.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 1126–1136
November 7–11, 2021. ©2021 Association for Computational Linguistics

early conversations, it can be utilized for future persona-aware response prediction.

Motivated by this, we propose a new task, named Speaker Persona Detection (SPD), which aims to detect a speaker's persona from the plain conversational text. As illustrated in Figure 1, the agent proactively leads the conversation with a goal to collect the speaker information implicitly. Through the conversation between the speaker and the agent, we can use the speaker's utterances carrying the personal information to search for a best-matched persona in the database. Here, a persona description is composed of several profiles characterizing a person, which is unstructured and common in practice (Zhang et al., 2018; Mazare et al., 2018; Gu et al., 2019; Liu et al., 2020; Gu et al., 2021). Then, the searched persona is utilized for follow-up personalized response generation or selection. The ability to learn speakers' personas can have wide applications in commercial chatbots, recommendation systems and other scenarios that involve conversations. For example, it can be applied to the recommendation system. Speakers first talk to the intelligent agent in their smart devices for several turns. Then the conversational text of the speaker is collected and utilized as a query to search for a best-matched persona description from candidates recorded in the database. Finally, the user preferences linked to the best-matched persona will be recommended to the speaker for providing personalized service.

In our proposed SPD task, a conversation context is composed of multiple utterances and a persona is composed of multiple profiles, which brings three challenges to existing studies on matching. First, the matching in SPD relies on modeling the long-term dependency among conversation utterances. Second, the matching in SPD is established between two sets of sentences which requires a complicated *many-to-many* matching framework. This has not been explored yet, as previous studies have been conducted either between a pair of sentences (i.e., *one-to-one*) (Wang et al., 2013; Bowman et al., 2015; Williams et al., 2018), or between a set of sentences and a single sentence (i.e., *many-to-one*) (Lowe et al., 2015; Lai et al., 2017; Wu et al., 2017). Third, there exists dynamic redundancy among conversation utterances and persona profiles. Specifically, the informativeness of each conversation utterance changes when inferring different profiles in a persona. Besides, some utterances may carry no personal information and some profiles may not be reflected by any utterances in the conversation.

A dataset is built for studying the SPD task. We transform the existing `Persona-Chat` dataset (Zhang et al., 2018) by assuming that conversation sessions are given while personas are what models should predict. We name the transformed dataset `Persona Match on Persona-Chat (PMPC)`. Regarding with SPD methods, context-to-persona (C2P) matching networks, which are established at a coarse granularity by concatenating two sets of sentences respectively, are employed as baselines. Furthermore, we propose utterance-to-profile (U2P) matching networks which are established at a fine granularity by treating each sentence in either contexts or personas individually. They first obtain the representation for each sentence and then derive the representations of contexts and personas through an aggregation operation. In this paper, (1) sentence-encoding-based framework such as the encoder of bag-of-words (BOW) (Joulin et al., 2017), BiLSTM (Hochreiter and Schmidhuber, 1997) and Transformer (Vaswani et al., 2017), and (2) cross-attention-based framework such as ESIM (Chen et al., 2017) and (3) pretraining-based framework such as BERT (Devlin et al., 2019) are considered when building either C2P and U2P models. The experimental results comparing the C2P and U2P models demonstrate the effectiveness of the latter for solving the SPD task, which relies on the *many-to-many* matching between two sets of sentences. In addition to the performance improvement, U2P models yield interpretability by explicitly scoring each utterance-profile pair and performing the aggregation operation.

In summary, our contributions in this paper are three-fold. (1) We propose a new task, Speaker Persona Detection (SPD), and construct a dataset for studying this problem, which make the first attempt to detect speaker personas from conversational texts by persona matching. (2) Many baseline methods have be established for the SPD task. (3) We propose U2P matching networks with a fine granularity to explore the matching between two sets of sentences, which outperform their C2P counterparts with a coarse granularity on the SPD task in our experiments. We hope the task and datasets will invite more research on detecting speaker profiles from conversational texts.

2 Related Work

Speaker Profile in Text Maintaining the consistency between a speaker and its utterances is an important issue in many NLP tasks. A bunch of work investigates how to generate or select a dialogue response which is consistent with a specific speaker for building personalized chatbots. The prior descriptions of the speaker are usually presented by a persona which is composed of multiple profiles. Li et al. (2016) proposed a persona-based neural conversation generation model to capture individual characteristics such as background information and speaking style. Zhang et al. (2018) constructed a `Persona-Chat` dataset using an artificial data collection mechanism based on Mechanical Turk. As a result, neither dialogues nor personas can be fully representative of real user-agent interactions and the dataset coverage remains limited, containing a bit more than 1K different personas. To imitate the real-life scenarios, Mazare et al. (2018) constructed a persona-based dialogue dataset using conversations previously extracted from Reddit, where personas are collected with simple heuristic rules. Gu et al. (2019) proposed a dually interactive matching network which adopted a dual matching architecture for presenting the personalities of dialogue agents in retrieval-based chatbots. Hua et al. (2020); Gu et al. (2020); Zhu et al. (2021) attempt to learn to detect relevant contexts and user profiles in retrieval-based chatbots. Gu et al. (2021) thoroughly explore the impact of utilizing personas that describe either self or partner speakers on the task of response selection in retrieval-based chatbots.

Matching Tasks Matching aims at searching for a best-matched one from a list of candidates. Determining the semantic matching degree or label between two pieces of text is a basic problem in many natural language understanding tasks. The existing matching or classification tasks can be generally categorized into two main categories that establish the relationship either (1) between a pair of sentences (Wang et al., 2013; Bowman et al., 2015; Williams et al., 2018), or (2) between a set of sentences and a single sentence (Lowe et al., 2015; Lai et al., 2017; Wu et al., 2017; Han et al., 2018). We name them *one-to-one* and *many-to-one* matching in this paper.

		Train	Valid	Test
10@1	# distractors (N)	1	9	9
100@1	# distractors (N)	1	99	99
# matched context-persona pairs		18K	2K	2K
Avg. # utterances per context		7.35	7.80	7.76
Avg. # words per utterance		11.67	11.94	11.79
Avg. # profiles per persona		4.50	4.49	4.50
Avg. # words per profile		7.32	7.82	7.56

Table 1: Statistics of the `PMPC`, where 10@1 and 100@1 correspond to using 9 and 99 distractors respectively in the validation and test sets.

Different from the studies mention above, our proposed SPD task is a new *many-to-many* semantic matching task, whose important characteristic is that both contexts and personas are composed of multiple sentences. This discourse-level matching is challenging since the informativeness of an utterance or a profile is dynamically changing, which requires models to filter out the redundant information automatically. Thus, we propose the utterance-to-profile (U2P) matching networks which are effective at filtering out less informative sentences and obtaining a more informative representation for the whole set.

3 Dataset Creation

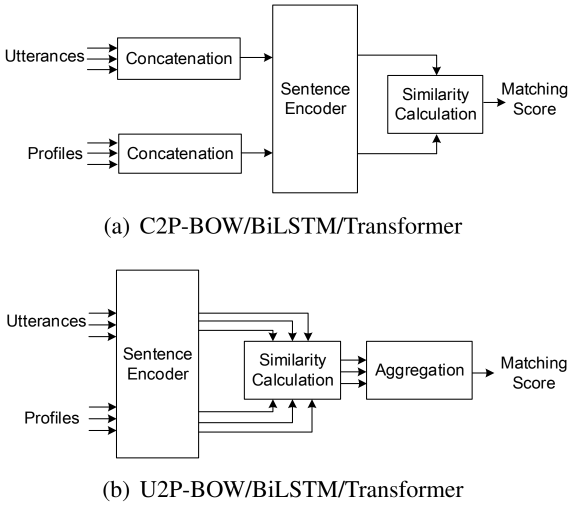
A dataset named `Persona Match on Persona-Chat (PMPC)` is built for studying the SPD task. Luckily, our data construction can be based on an existing `Persona-Chat` dataset (Zhang et al., 2018) which is a high-quality one made available at MILA and Facebook AI. Each dialogue in `Persona-Chat` was created by first assigning each human speaker a random persona and then asking them to chat in pair conditioned on the given personas. The personas in `Persona-Chat` are created by crowd-sourcing and modified by human. Profiles in a persona co-refer to each other, and each profile describes different properties of a coherent persona. Due to the natural dataset creating method, the conversational texts conditioned on the given personas can intuitively reflect characteristics of the speaker, which creates the natural matching relationship between the conversational texts and the persona of a speaker. Thus, we assume that `Persona-Chat` is suitable for studying the SPD task. Since each dialogue in `Persona-Chat` was performed between two speakers, we can consider one of them as human speaker and the other as intelligent agent, and then exchange with each other. In this way, each dialogue in Persona-Chat can provide two matched context-persona pairs. Two versions of personas, including original ones and revised ones were provided by Persona-Chat. The latter is constructed by rephrasing, generalizing, or specializing the original one. The revised version is adopted in this paper to make the SPD task more challenging.

The task of SPD is defined as selecting a best-matched persona from a list of candidates according to the conversational texts of the speaker, as shown in Figure 1. The candidate set is composed of one correct persona and *N* incorrect personas, which we call *distractors*. The context is used as the input conversational texts for detecting speaker personas. The matched persona of the context is adopted to represent the correct persona in the candidate set. Besides, *N* random persona distractors are chosen from the persona pool to form the complete candidate set. Table 1 presents the statistics of the PMPC dataset. Two configurations of *N* (i.e., 9 and 99) are used to construct the validation and test sets. There is no overlap on contexts and personas among the training, validation and test sets in Persona-Chat. We follow this setting by constructing context-persona pairs in the respective sets. Thus, there is no overlap in PMPC as well.

4 Models

We present the formal definition of SPD as follows. Given a dataset $\mathcal{D}$, an example of the dataset can be represented as (c, p, y). Specifically, $c = \{u_1, u_2, ..., u_{n_c}\}$ represents a context with $\{u_m\}_{m=1}^{n_c}$ as its utterances and n_c as the utterance number. $p = \{p_1, p_2, ..., p_{n_p}\}$ represents a persona with $\{p_n\}_{n=1}^{n_p}$ as its profiles and n_p as the profile number similarly. $y \in \{0, 1\}$ denotes a label. $y = 1$ indicates that c and p are a matched pair; otherwise, $y = 0$. Our goal is to learn a matching model $g(c, p)$ from $\mathcal{D}$. For any pair (c, p), $g(c, p)$ measures the matching degree between c and p. We learn $g(c, p)$ by minimizing its cross-entropy loss on $\mathcal{D}$. Let Θ denote the parameters of the matching model. The objective function $\mathcal{L}(\mathcal{D}, \Theta)$ of learning can be formulated as

$$\mathcal{L}(\mathcal{D}, \Theta) = -\sum_{(c,p,y)\in\mathcal{D}} [ylog(g(c,p)) + (1-y)log(1-g(c,p))]. \quad (1)$$



(a) C2P-BOW/BiLSTM/Transformer

(b) U2P-BOW/BiLSTM/Transformer

Figure 2: Model architectures of (a) C2P-BOW/BiLSTM/Transformer and (b) U2P-BOW/BiLSTM/Transformer.

Sentence-encoding-based, cross-attention-based and pretraining-based frameworks are followed to build matching models. Under each framework, C2P and U2P matching networks are designed, adopting the *one-to-one* and *many-to-many* matching respectively. The details of these models are introduced in the following subsections.

4.1 Sentence-Encoding-Based Models

Under this framework, three types of sentence encoders, BOW, BiLSTM and Transformer, are employed. They share the same model architecture except the sentence encoder. First, BOW is employed to explore whether simple *n*-gram overlap could solve this task easily, which can prove the importance of considering the long-term dependency in this task. Second, BiLSTM and Transformer are employed to discuss the impact of chronological (BiLSTM) or parallel (Transformer) encoding on this task. Due to the space limit, we omit the descriptions on basic BOW, BiLSTM or Transformer. Readers can refer to Joulin et al. (2017), Hochreiter and Schmidhuber (1997) and Vaswani et al. (2017) for more details.

4.1.1 C2P-BOW/BiLSTM/Transformer

Figure 2(a) presents the model architecture of C2P-BOW/BiLSTM/Transformer. Each utterance u_m and each profile p_n are first converted to their respective embedding matrices $\mathbf{U}_m = \{\mathbf{u}_{m,i}\}_{i=1}^{l_{u_m}}$ and $\mathbf{P}_n = \{\mathbf{p}_{n,j}\}_{j=1}^{l_{p_n}}$ by looking up a word embedding table, where l_{u_m} and l_{p_n} are the lengths of u_m and p_n. Each $\mathbf{u}_{m,i}$ or $\mathbf{p}_{n,j}$ is an embedding vector. Then, the context representations $\mathbf{C} = \{\mathbf{c}_i\}_{i=1}^{l_c}$ with $l_c = \sum_{m=1}^{n_c} l_{u_m}$ are formed by concatenating the set of utterance representations $\{\mathbf{U}_m\}_{m=1}^{n_c}$. The persona representations $\mathbf{P} = \{\mathbf{p}_j\}_{j=1}^{l_p}$ with $l_p = \sum_{n=1}^{n_p} l_{p_n}$ are formed similarly. Furthermore, the context representations $\mathbf{C}$ and the persona representations $\mathbf{P}$ are encoded by a shared sentence encoder, which is one of BOW, BiLSTM or Transformer. A max pooling operation is followed to derive the vectors $\bar{\mathbf{c}} \in \mathbb{R}^d$ and $\bar{\mathbf{p}} \in \mathbb{R}^d$ for representing the context and the persona. Finally, a similarity matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ and a sigmoid function σ are used to compute the final matching degree $g(c, p)$ as follows,

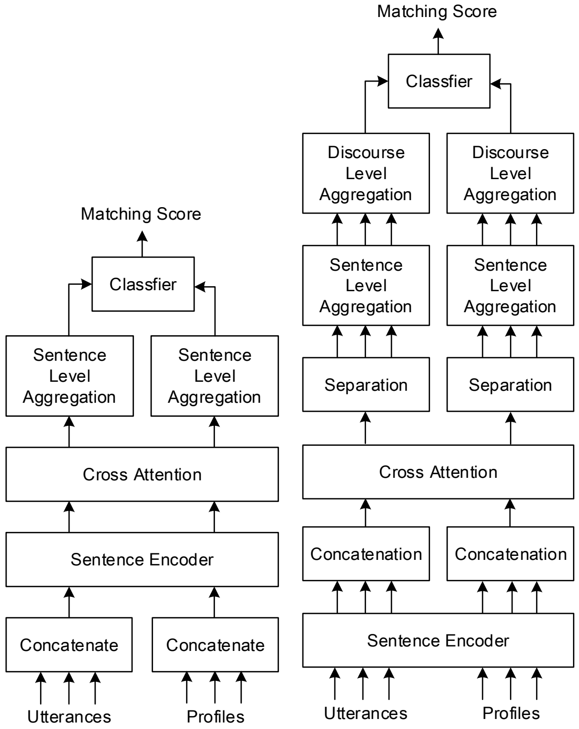
$$g(c, p) = \sigma(\,\bar{\mathbf{c}}^\top \cdot \mathbf{A} \cdot \bar{\mathbf{p}}), \tag{2}$$

where $g(c, p)$ denotes the matching score for a (c, p) pair.

4.1.2 U2P-BOW/BiLSTM/Transformer

Figure 2(b) shows the model architecture of U2P-BOW/BiLSTM/Transformer. In these models, the word representations are constructed in the same way as their C2P counterparts. Instead of concatenating first, each utterance $\mathbf{U}_m$ and each profile $\mathbf{P}_n$ is encoded in parallel and separately by one of BOW, BiLSTM or Transformer encoder. Then, the same pooling operation is applied, and a set of utterance representations $\{\bar{\mathbf{u}}_m\}_{m=1}^{n_c}$ and a set of profile representations $\{\bar{\mathbf{p}}_n\}_{n=1}^{n_p}$ are derived. A similarity score s_{mn} is computed for each utterance-profile pair as follows,

$$s_{mn} = \bar{\mathbf{u}}_m^\top \cdot \mathbf{A} \cdot \bar{\mathbf{p}}_n. \tag{3}$$



(a) C2P-ESIM (b) U2P-ESIM

Figure 3: Model architectures of (a) C2P-ESIM and (b) U2P-ESIM.

In order to obtain the matching score between the whole set of utterances and the whole set of profiles, additional aggregation operations are required. Here, we make an assumption that one utterance can only reflect one profile. Thus, for a given utterance, its matching score with the persona is defined as the maximum matching score between it and all profiles. If no profile provides positive scores for an utterance, this utterance is considered as uninformative and its matching score is set to 0. Finally, we accumulate the matching scores of all utterances and derive the final matching score $g(c, p)$. Mathematically, we have

$$s_m = \max\{\max_n \; s_{mn}, 0\}, \tag{4}$$

$$s = \sum_{m=1}^{n_c} s_m, \tag{5}$$

$$g(c, p) = \sigma(s), \tag{6}$$

More verification about the aggregation methods and our assumption will be discussed in Section of Experiments.

4.2 Cross-Attention-Based Models

ESIM (Chen et al., 2017) is adopted as the basis of models under this framework.

4.2.1 C2P-ESIM

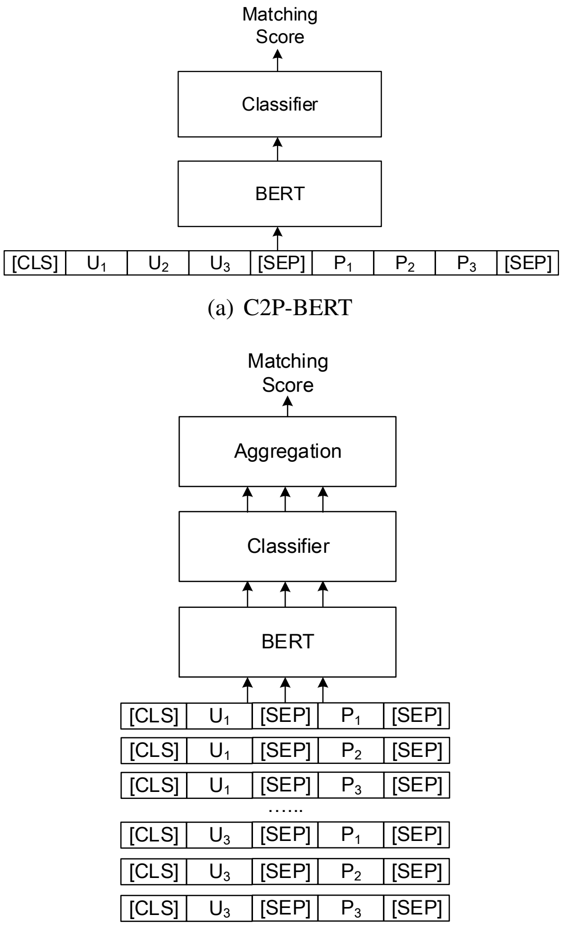
Figure 3(a) presents the model architecture of C2P-ESIM. It shares the same word representation, sentence concatenation and sentence encoder as those in C2P-BiLSTM to derive the encoded context and persona representations $\bar{\mathbf{C}} = \{\bar{\mathbf{c}}_i\}_{i=1}^{l_c}$ with $l_c = \sum_{m=1}^{n_c} l_{u_m}$ and $\bar{\mathbf{P}} = \{\bar{\mathbf{p}}_j\}_{j=1}^{l_p}$ with $l_p = \sum_{n=1}^{n_p} l_{p_n}$. Then, an attention-based alignment is performed to obtain $\mathbf{C}^{mat}$ and $\mathbf{P}^{mat}$. Furthermore, sentence-level aggregation operations are performed to obtain the context and persona embedding vectors $\mathbf{c}^{agr}$ and $\mathbf{p}^{agr}$. Finally, the matching feature vector is formed by concatenating $\mathbf{c}^{agr}$ and $\mathbf{p}^{agr}$, and is sent into a multi-layer perceptron (MLP) classifier for calculating $g(c, p)$.

4.2.2 U2P-ESIM

Figure 3(b) presents the model architecture of U2P-ESIM. Different from C2P-ESIM which encodes long sequences after concatenating context utterances and persona profiles, U2P-ESIM first encodes each sentence in parallel and separately. It first encodes each utterance and profile separately to derive $\{\bar{\mathbf{U}}_m\}_{m=1}^{n_c}$ and $\{\bar{\mathbf{P}}_n\}_{n=1}^{n_p}$. To apply the attention-based alignment between a (c, p) pair, $\{\bar{\mathbf{U}}_m\}_{m=1}^{n_c}$ are concatenated to form the encoded context representations $\bar{\mathbf{C}} = \{\bar{\mathbf{c}}_i\}_{i=1}^{l_c}$ with $l_c = \sum_{m=1}^{n_c} l_{u_m}$. The encoded persona representations $\bar{\mathbf{P}} = \{\bar{\mathbf{p}}_j\}_{j=1}^{l_p}$ are formed similarly with $l_p = \sum_{n=1}^{n_p} l_{p_n}$ The same cross attention used in C2P-ESIM is performed between $\bar{\mathbf{C}}$ and $\bar{\mathbf{P}}$ to obtain $\mathbf{C}^{mat}$ and $\mathbf{P}^{mat}$, which are further converted back into separate utterances $\{\mathbf{U}_m^{mat}\}_{m=1}^{n_c}$ and profiles $\{\mathbf{P}_n^{mat}\}_{n=1}^{n_p}$ according to the lengths of individual sentences. Then, the same sentence-level aggregation in C2P-ESIM is applied to process each $\mathbf{U}_m^{mat}$ and $\mathbf{P}_n^{mat}$ to derive a set of utterance embeddings $\{\mathbf{u}_m^{agr}\}_{m=1}^{n_c}$ and a set of profile embeddings $\{\mathbf{p}_n^{agr}\}_{n=1}^{n_p}$. An additional discourse-level aggregation is performed over $\{\mathbf{u}_m^{agr}\}_{m=1}^{n_c}$ and $\{\mathbf{p}_n^{agr}\}_{n=1}^{n_p}$ to get the embedding vectors $\mathbf{c}^{agr}$ for the whole context and $\mathbf{p}^{agr}$ for the whole persona. Different aggregation strategies are designed for them. As the context utterances are extracted from a conversation and their chronological relationships are maintained, $\{\mathbf{u}_m^{agr}\}_{m=1}^{n_c}$ are sent into another BiLSTM and aggregated by a pooling operation. On the other hand, the profiles in a persona have no chronological orders. Thus, an attention-based aggregation strategy is employed which first projects each profile vector to a scalar and then computes the attention weight for each profile. The aggregated persona representation $\mathbf{p}^{agr}$ is calculated as the weighted sum of $\{\mathbf{p}_n^{agr}\}_{n=1}^{n_p}$. Finally, the matching feature is also the concatenation of $\mathbf{c}^{agr}$ and $\mathbf{p}^{agr}$, which is sent into an MLP classifier to return the matching degree.

4.3 Pretraining-Based Models

BERT (Devlin et al., 2019) is adopted as the basis of models under this framework.



(a) C2P-BERT

(b) U2P-BERT

Figure 4: Model architectures of (a) C2P-BERT and (b) U2P-BERT.

4.3.1 C2P-BERT

Figure 4(a) presents the model architecture of C2P-BERT. In BERT, sentence A and sentence B are concatenated with a `[SEP]` token, which is then sent into BERT and classified through the representation of the `[CLS]` token. Here, utterances are concatenated to form the sentence A, and profiles are concatenated to form the sentence B. The token type schema of C2P-BERT in our current manuscript strictly follows the original BERT to provide fair comparisons. Interactions are performed between the whole context and the whole persona at a coarse granularity through stacked Transformer blocks. Finally, the representation of the `[CLS]` token is sent into an MLP classifier to return the matching degree $g(c, p)$.

4.3.2 U2P-BERT

Figure 4(b) presents the model architecture of U2P-BERT. In order to perform the matching at

Model	$\mathbf{R}_{10}$@1	$\mathbf{MRR}_{10}$	$\mathbf{R}_{100}$@1	$\mathbf{MRR}_{100}$
C2P-BOW	34.7 ± 1.2	54.4 ± 0.9	8.9 ± 0.5	19.5 ± 0.5
U2P-BOW	46.5 ± 1.7	63.3 ± 1.3	16.9 ± 1.2	28.5 ± 1.2
C2P-BiLSTM	38.3 ± 1.2	57.7 ± 0.9	8.1 ± 0.8	19.2 ± 0.9
U2P-BiLSTM	57.4 ± 1.4	71.0 ± 1.4	24.0 ± 1.6	37.5 ± 1.6
C2P-Transformer	49.6 ± 3.7	65.3 ± 2.5	19.0 ± 1.5	30.5 ± 1.1
U2P-Transformer	56.2 ± 1.5	70.6 ± 1.1	22.9 ± 1.3	36.0 ± 1.3
C2P-ESIM	80.7 ± 0.5	87.7 ± 0.4	50.7 ± 1.4	62.8 ± 0.7
U2P-ESIM	81.6 ± 1.0	88.4 ± 0.6	54.5 ± 1.3	66.6 ± 0.7
C2P-BERT	87.4 ± 0.7	91.8 ± 0.4	64.7 ± 1.5	75.4 ± 0.8
U2P-BERT	90.4 ± 0.5	94.3 ± 0.2	79.1 ± 0.9	83.2 ± 0.5

Table 2: Evaluation results (mean ± standard deviation) (%) of different models on the test set of `PMPC` in which each test sample had either 9 or 99 distractors.

a finer granularity, interactions and matching are performed between each utterance and each profile. In detail, for a specific utterance u_m which is used to form the sentence A, it is concatenated with each profile p_n which is used to form the sentence B respectively. The same concatenation operation is performed for each utterance. In this way, we can derive several sets of concatenated sequences. These concatenated sequences are sent into BERT for encoding in parallel. Their output `[CLS]` representations are sent into an MLP classifier to return several sets of similarity score s_{mn}, which denotes the matching degree between u_m and p_n. Finally, additional aggregation operations as introduced by Eq. (4), Eq. (5) and Eq. (6) are applied to get $g(c, p)$ denoting the final matching degree between the whole set of utterances and the whole set of profiles.

5 Experiments

5.1 Evaluation Metrics

We adopted the evaluation metrics popularly used by other text retrieval tasks (Lowe et al., 2015; Wu et al., 2017; Zhang et al., 2018). Each model was tasked with selecting the k best-matched ones from n available candidates, and we calculated the recall of the true positive one among the k selected candidates, denoted as $\mathbf{R}_n@k$. In addition, mean reciprocal rank, denoted as $\mathbf{MRR}_n$, was also calculated, which was the average of reciprocal ranks of retrieval results among n available candidates.

5.2 Training Details

For BOW, BiLSTM, Transformer and ESIM, the word representations were 300-dimensional GloVe embeddings (Pennington et al., 2014) and were not updated during training. All hidden states of the LSTM had 200 dimensions. The MLP at the prediction layer had 256 hidden units with ReLU (Nair and Hinton, 2010) activation. Dropout (Srivastava et al., 2014) with a rate of 0.2 was applied to the word embeddings and all hidden layers. The maximum utterance length, maximum number of utterances in a context, maximum profile length and maximum number of profiles in a persona were set to 20, 8, 15 and 5 respectively for `PMPC`. Zeros were padded if the number of utterances in a context and the number of profiles in a persona were less than the limits. Otherwise, we kept the last 8 utterances in the context and the last 5 profiles in the persona. The Adam method (Kingma and Ba, 2014) was employed for optimization. The learning rate was initialized as 1e-3 and was exponentially decayed by 0.96 every 5000 steps. The validation set was utilized to select the best model for testing. For BERT, we adopted the base version whose number of Transformer block is set to 12, hidden size is set to 768 and number of self-attention heads is set to 12. The learning rate was initialized as 2e-5.

All codes were implemented under the TensorFlow framework (Abadi et al., 2016) and are published along with the dataset to help replicate our results. [1]

5.3 Experimental Results

Table 2 presents the evaluation results of different models on the test set of `PMPC`. Each model was trained for 10 times with identical architectures and different random initializations.

[1] https://github.com/JasonForJoy/SPD

Aggregation Strategy	$\mathbf{MRR}_{10}$	$\mathbf{R}_{10}$@1
Ps-*Max* & Us-*Sum*	71.0 ± 1.4	57.4 ± 1.4
Ps-*Max* & Us-*Max*	70.0 ± 1.3	53.7 ± 1.9
Ps-*Sum* & Us-*Max*	57.3 ± 0.8	37.1 ± 1.0
Ps-*Sum* & Us-*Sum*	67.5 ± 0.7	51.5 ± 1.1

Table 3: Evaluation results (%) of U2P-BiLSTM models with different aggregation strategies on the test set of PMPC ($N = 9$). Ps denotes Profiles and Us denotes Utterances. *Max* and *Sum* denote the aggregation operation used in Eq. (4) and Eq. (5).

	U1	U2	U3	U4	U5	U6
P1	-0.07	-0.35	-0.22	-0.70	-1.05	-0.19
P2	-0.16	0.90	0.72	-0.20	0.38	-0.34
P3	0.83	1.14	1.00	-0.48	0.05	-0.10
P4	-0.92	-1.17	-0.89	-0.64	-2.21	-0.09
s_m	0.83	1.14	1.00	0.0	0.38	0.0

Table 4: Utterance-profile similarity scores for the matched context-persona pair shown in Figure 1. Here, Um and Pn denote the m-th utterance and the n-th profile respectively.

It can be noticed that all U2P models achieved better performance than their C2P counterparts on all metrics. Specifically, U2P-BOW, U2P-BiLSTM, U2P-Transformer, U2P-ESIM and U2P-BERT outperformed their C2P counterparts by margins of 11.8%, 19.1%, 6.6%, 0.9% and 3.0% respectively in terms of $\mathbf{R}_{10}$@1, and by margins of 8.0%, 15.9%, 3.9%, 3.8% and 14.4% respectively in terms of $\mathbf{R}_{100}$@1. These results demonstrate the effectiveness of treating both contexts and personas as sets of multiple sequences and conducting *many-to-many* matching for this task. Comparing BOW, BiLSTM and Transformer, it can be seen that it is actually necessary to model the long-term dependency for this task, as a simple *n*-gram overlap based method could not perform competitively with models considering the long-term dependency. Comparing BiLSTM with Transformer at both coarse and fine granularities, BiLSTM achieved larger improvement than Transformer after employing the U2P framework, which shows its effectiveness especially for models with chronological encoders. Comparing C2P-BERT with U2P-BERT, although C2P-BERT has been the state-of-the-art model for capturing the matching information, U2P-BERT could still provide additional improvement. U2P-BERT achieves only 79.1% $\mathbf{R}_{100}$@1, which shows that this task is difficult and there is a lot of room for further improvement.

5.4 Analysis

Aggregation Method One characteristic of U2P matching networks is that the discourse-level aggregation is necessary to assemble the matching scores for each utterance-profile pair. As introduced by Eq. (4) and Eq. (5), the aggregation operation employs *Max* over the set of profiles to select the best-matched profile for each utterance, and then employs *Sum* over the set of utterances to accumulate the matching scores for all utterances. In order to verify the effectiveness of this strategy, we replaced it with other settings and evaluated them on the test set as shown in Table 3. By comparing the first row with the fourth row, as well as the second row with the third row, we can see that *Max* achieved better performance of aggregating profiles than *Sum*, which supports our assumption that one utterance may reflect only one profile. Then, by comparing the first row with the second row, as well as the third row with the fourth row, we can see that *Sum* achieved better performance of aggregating utterances than *Max*, which indicates that multiple utterances should be considered when deriving the matching score for a context-persona pair.

Case Study We further illustrate the interpretability of the aggregation operation in U2P models by conducting a case study as shown in Table 4, which contains the utterance-profile similarity scores for the matched context-persona pair shown in Figure 1. First, we can see that these matching scores computed by U2P models are reasonable. For example, U1 (*I actually just got back from a hike*) and U5 (*I am single mom*) achieved high similarity scores with P3 (*I enjoy nature walks*) and P2 (*I am raising sons all on my own*) respectively. Second, the s_m values calculated by Eq. (4) indicate that some utterances may be uninformative and the model filtered them out automatically when calculating the final score for a context-persona pair. Scoring each utterance-profile pair at a fine granularity is effective to select the best-matched profile for each utterance and filter out uninformative utterances at the same time.

Model	Time (s)	Parameters
C2P-BOW	7.1	90k
U2P-BOW	8.6	90k
C2P-BiLSTM	17.1	962K
U2P-BiLSTM	12.2	962K
C2P-Transformer	8.3	271K
U2P-Transformer	10.3	271K
C2P-ESIM	36.7	4.1M
U2P-ESIM	22.4	5.7M
C2P-BERT	121.3	110M
U2P-BERT	742.8	110M

Table 5: The inference time over the validation set of PMPC whose configuration of *N* was 9 using different models, together with their numbers of parameters.

Complexity We analysed the time and space complexity difference between the C2P and U2P models.

In order to explore the efficiency difference between C2P and U2P models, we analysed their time complexity by comparing their run-time computation speed. We recorded the inference time over the validation set of PMPC whose configuration of *N* was 9 using a GeForce GTX 1080 Ti GPU. The inference time of each model is measured on a batch basis, instead of on a per-instance basis. The results are shown in Table 5. As we can see that, U2P-BOW was not as efficient as C2P-BOW. So did U2P-Transformer and C2P-Transformer. However, U2P-BiLSTM and U2P-ESIM were faster than their C2P counterparts by 28.7% and 39.0% respectively, although the U2P models relied on additional aggregation operations. The reason may be that the parallel encoding of multiple sequences in U2P networks can improve the efficiency of RNN-based sentence encoders but can not benefit the BOW-based or Transformer-based ones. U2P-BERT took more time than C2P-BERT as the calculation of former is an order of magnitude higher than the latter. However, the empirical improvement of U2P-BERT far outweighed the increased computation cost.

The number of parameters of these C2P and U2P models was used to evaluate the space complexity of different models. The results are shown in Table 5. We can see that C2P-BOW and U2P-BOW contained the same number of parameters. So did C2P-BiLSTM, C2P-Transformer and C2P-BERT with their U2P counterparts. The reason is that the additional aggregation operations in these U2P models consume only the calculation of *Max* or *Sum* function, while do not require additional parameters. U2P-ESIM adopted an additional BiLSTM for discourse-level aggregation, and thus contained more parameters than C2P-ESIM.

These results show that the U2P models outperform their C2P counterparts significantly with comparable time and space complexity.

6 Conclusion

In this paper, we propose the task of Speaker Persona Detection (SPD) and build a PMPC dataset for studying this task. In SPD, the matching from conversational contexts to persona candidates is established between two sets of sentences and requires a new *many-to-many* matching framework. Thus, this paper proposes the utterance-to-profile (U2P) matching networks, which treat both contexts and personas as sets of multiple sequences. Results show that the proposed U2P matching networks outperform their context-to-persona (C2P) counterparts, which treat both contexts and personas as single sequences. In the future, we will explore better aggregation methods and investigate model structures for *many-to-many* matching.

Ethics

The PMPC dataset is transformed from the existing Persona-Chat dataset, personas and conversations of which are constructed by crowd-sourcing. We quote the description of the Persona-Chat dataset (Zhang et al., 2018) as follows: "We asked the crowdsourced workers to create a character (persona) description using 5 sentences. Because the personas are not the real profiles of the Turkers, the dataset does not contain personal information (and they are told specifically not to use any)." Furthermore, the personas and conversations are based on typical topics of daily life that speakers can bring up in conversation, without involving information such as person's attributes, race, etc. Thus, there are no ethical concerns in the Persona-Chat dataset and the PMPC dataset used in this paper.

Acknowledgements

We thank the anonymous reviewers for their valuable comments.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced lstm for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1657–1668.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jia-Chen Gu, Zhen-Hua Ling, Quan Liu, Zhigang Chen, and Xiaodan Zhu. 2020. Filtering before iteratively referring for knowledge-grounded response selection in retrieval-based chatbots. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1412–1422.

Jia-Chen Gu, Zhen-Hua Ling, Xiaodan Zhu, and Quan Liu. 2019. Dually interactive matching network for personalized response selection in retrieval-based chatbots. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1845–1854.

Jia-Chen Gu, Hui Liu, Zhen-Hua Ling, Quan Liu, Zhigang Chen, and Xiaodan Zhu. 2021. Partner matters! An empirical study on fusing personas for personalized response selection in retrieval-based chatbots. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 565–574.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809.

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

Kai Hua, Zhiyuan Feng, Chongyang Tao, Rui Yan, and Lu Zhang. 2020. Learning to detect relevant contexts and knowledge for response selection in retrieval-based dialogue systems. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 525–534.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015.*

Alice Lai, Yonatan Bisk, and Julia Hockenmaier. 2017. Natural language inference from multiple premises. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 100–109.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 994–1003.

Qian Liu, Yihong Chen, Bei Chen, Jian-Guang Lou, Zixuan Chen, Bin Zhou, and Dongmei Zhang. 2020. You impress me: Dialogue generation via mutual persona perception. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1417–1427.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 285–294.

Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2018. Mem2seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1468–1478.

Pierre-Emmanuel Mazare, Samuel Humeau, Martin Raison, and Antoine Bordes. 2018. Training millions of personalized dialogue agents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2775–2779.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, et al. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 3776–3783.

Haoyu Song, Yan Wang, Kaiyan Zhang, Wei-Nan Zhang, and Ting Liu. 2021. BoB: Bert over bert for training persona-based dialogue models from limited personalized data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 167–177.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. One time of interaction may not be enough: Go deep with an interaction-over-interaction network for response selection in dialogues. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1–11.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. 2013. A dataset for research on short-text conversations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 935–945.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.

Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 496–505.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and William B Dolan. 2020. DIALOGPT: Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278.

Yutao Zhu, Jian-Yun Nie, Kun Zhou, Pan Du, and Zhicheng Dou. 2021. Content selection network for document-grounded retrieval-based chatbots. In *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021*, pages 755–769.