Refine and Imitate: Reducing Repetition and Inconsistency in Persuasion Dialogues via Reinforcement Learning and Human Demonstration

Weiyan Shi\textsuperscript{1}, Yu Li\textsuperscript{2}, Saurav Sahay\textsuperscript{3}, Zhou Yu\textsuperscript{1}
Columbia University\textsuperscript{1} University of California, Davis\textsuperscript{2}, Intel Labs\textsuperscript{3}
ws2634@columbia.edu, yooli@ucdavis.edu
saurav.sahay@intel.com, zy2461@columbia.edu

Abstract

Persuasion dialogue system reflects the machine’s ability to make strategic moves beyond verbal communication, and therefore differentiates itself from task-oriented or open-domain dialogues and has its own unique values. However, the repetition and inconsistency problems still persist in dialogue response generation and could substantially impact user experience and impede the persuasion outcome. Besides, although reinforcement learning (RL) approaches have achieved big success in strategic tasks such as games, it requires a sophisticated user simulator to provide real-time feedback to the dialogue system, which limits the application of RL on persuasion dialogues. To address these issues towards a better persuasion dialogue system, we apply RL to refine a language model baseline without user simulators, and distill sentence-level information about repetition, inconsistency, and task relevance through rewards. Moreover, to better accomplish the persuasion task, the model learns from human demonstration to imitate human persuasion behavior and selects the most persuasive responses. Experiments show that our model outperforms previous state-of-the-art dialogue models on both automatic metrics and human evaluation results on a donation persuasion task, and generates more diverse, consistent and persuasive conversations according to the user feedback. The code is available at \url{https://github.com/wyshi/consistency}.

1 Introduction

Persuasion dialogue systems have become an increasingly important subject in both social science and computational linguistics (Prakken, 2006, 2009; Wang et al., 2019; Asai et al., 2020). Such systems aim to employ conversational strategies to change the audience’s attitude or behaviour, and therefore, are inherently difficult to build with multiple challenges. The first one is that users often expect highly smooth conversation experience from persuasion systems in order to be persuaded (Shi et al., 2020). So the long-standing problems of dialogue repetition and inconsistency can be especially salient in persuasion dialogue systems. Secondly, different from traditional dialogue tasks, the persuasion task is non-collaborative where the user and the system have different goals (Li et al., 2020b), and hence highly intellectual and strategic.

Previous studies have attempted to address the first challenge, the dialogue repetition and inconsistency problems, by changing the object function in supervised learning (Li et al., 2020a) or applying reinforcement learning (RL) (Li et al., 2016; Liu et al., 2018). But these methods either may lead to uninterpretable behaviors, or rely on handcrafted user simulators that are hard to design for persuasion dialogues. To tackle these challenges, we propose to extract a policy directly from the data and let the models learn from its own mistakes without the use of simulators. Leveraging decoding methods such as Nucleus Sampling (Holtzman et al., 2020), the finetuned language model can generate lexically diverse response candidates given the same context. Some candidates are appropriate, while others are repetitive or inconsistent. These good and bad examples are used as positive and negative feedback to the model through meaningful rewards in RL, and help refine the language model.

Besides being diverse and consistent, a good response in persuasion dialogues also needs to accomplish the task: to persuade people. Existing work simply relied on the language models to generate persuasive responses (Li et al., 2020b; Wu et al., 2021b), which could result in uncontrollable task-oblivious replies. To quantify intellectual persuasion activities, we employ imitation learning, and ask human experts to demonstrate the persuasion process. We build a response imitator to imitate these human demonstrations and select the most persuasive responses in our framework.
We evaluate our models on a donation persuasion task (Wang et al., 2019), and deploy the persuasion systems on Amazon Mechanical Turk to interact with real users. The results on both automatic and human evaluations show that our systems achieve better persuasion outcomes (higher donation amount and donation ratio), and generates more diverse, consistent and persuasive responses compared to the baselines.

This work makes multiple contributions. Firstly, we propose the first RL-based persuasive dialogue system framework that achieves state-of-the-art performance on a complex donation persuasion task. Secondly, we design DialGAIL, an RL-based generative algorithm to refine a baseline language model for dialogue generation without the use of user simulators. Additionally, we introduce a human persuasion demonstration dataset that can be used for future research. Previous dialogue research has mostly focused on pure task-oriented dialogues and pure social conversations; but looking forward, it becomes more and more important to pay attention to strategic dialogues that involves both task and social components. We sincerely hope this work could inspire more research and discussions on strategic dialogues in the community.

2 Related Work

Strategic dialogue tasks such as persuasion and negotiation have emerged and attracted more attention recently, given its wide applications in industry and daily life (Lewis et al., 2017; He et al., 2018; Wang et al., 2019; Li et al., 2020b; Shi et al., 2020). These tasks are close to human-human conversations and often contain both a specific task goal and social components to build rapport for better task completion. Previously, Mazzotta et al. (2007) proposed an agenda-based user-adapted persuasion system to build relationship with users and change their eating habit. Yuan et al. (2008) developed a dialogue system for educational debate with strategic heuristics. More recently, Li et al. (2020b) utilized large-scale language models to build a donation persuasion system by generating multiple responses and selecting appropriate candidates with human-defined rules. We take a similar approach to generate candidates but eliminate the manual work for rule design, and teach the model to select task-relevant candidates through human demonstration.

Although large-scale language models have achieved great success in many NLP tasks, these models still suffer from repetition and inconsistency when applied to dialogue tasks. Many previous studies have worked on these issues (Wu et al., 2021b; Li et al., 2020a; Song et al., 2020). For example, Li et al. (2020a) proposed to detect the inconsistency with natural language inference data, and penalize it with unlikelihood loss to achieve more consistent personality in open-domain dialogues. Song et al. (2020) detected and rewrote the contradicting responses to achieve a more consistent personality. Our work tackles these problems with RL to reduce exposure bias in supervised learning and improve the interpretability.

Previous work has also explored RL-based methods in dialogue system building (Li et al., 2016; Liu et al., 2018; Shi et al., 2019a,b). For instance, Li et al. (2016) integrated the goal of coherent into the reward design towards more diverse dialogue generation. Liu et al. (2018) presented a hybrid reinforcement and imitation learning approach to enable the agent to learn from interactions with users in task-oriented dialogues. However, such methods not only rely on hand-crafted user simulators that are inherently hard to build (Shi et al., 2019a) for persuasion systems, but also require meaningful rewards that are difficult to design. In this work, we propose to let the model learn from its own mistakes by generating multiple responses without the use of simulators.

Our work is also closely related to response selection, which focuses on obtaining good context representations to match the context and retrieve the best response from a large collection of human-human conversations. However, such response selection models are highly dependent on the quality and availability of the underlying datasets. To address the data scarcity issue, Henderson et al. (2019) pretrained a response selection model with large conversational corpora, and finetuned it on new domains in task-oriented settings for a better context representation. Instead of retrieving candidates from human dialogues, we adopt the imitation learning approach, and leverage language models’ ability to generate coherent responses, and build a selector to imitate human selection process and choose among the generated candidates.

3 Methods: PersRFI

Our framework is shown in Figure 1. The language model is $p_{θ}$ and there are two steps in the frame-
Figure 1: The overall architecture of our PersRFI model. During training, $p_\theta$ generates $n$ response candidates; 
*Response Detector* annotates them with corresponding status such as “Repetition”; and the response candidates along with the golden human response send feedback to refine $p_\theta$ through the rewards. During testing, the refined $p_\theta^*$ generates $n$ candidates again; *Response Filter* removes the detected repetitive and inconsistent candidates; and *Response Imitator* imitates human demonstrations to select the most persuasive candidate as the final output. The dialogue history consists of the dialogue context and the Profiles.

work. 1) the reinforcement learning (RL) process to refine a baseline language model $q$ for better response generation (i.e., $p_{\theta_0} = q$), and 2) the imitation process to learn from human demonstration and select the best response. During RL training, for each user utterance, $p_\theta$ generates $n$ response candidates, shown in the *Response Candidates* box. Then the *Response Detector* annotates these candidates with corresponding status such as “Repetition” and “Inconsistency”. These labels along with the golden human response provide feedback through the reward function to guide $p_\theta$ to generate nonrepetitive and consistent responses. During test time, we use the refined language model $p_{\theta^*}$ to generate $n$ candidates again, and apply the *Response Filter* to remove the repetitive and inconsistent candidates to further ensure the candidate quality. Finally, the *Response Imitator* takes in the remaining candidates, and imitates the human demonstration to select one persuasive candidate as the final response. To detect repetition and inconsistency, we build *USR Profile* and *SYS Profile* shown in the top right table in Figure 1, where task-relevant information is extracted from the dialogue and stored as $<key: value>$ pairs, such as “want_to_donate: No”. We describe each module below.

### 3.1 Refine with Reinforcement Learning

#### 3.1.1 DialGAIL

One major issue with current RL-based dialogue training is that it requires a sophisticated user simulator to provide real-time feedback to the dialogue system. But in persuasion task, designing a persuadee simulator that can have diverse responses to persuasion is as hard as building the persuasion system itself. To eliminate the user simulator, we extend GAIL (Ho and Ermon, 2016) to dialogues settings and propose DialGAIL. The basic idea is to start with a baseline model, then use it to explore more space by generating multiple responses, and finally provide different rewards to the responses to refine the original model. In this way, DialGAIL extracts a policy directly from the training dialogues and learn from its own mistakes.

Algorithm 1 shows the steps in DialGAIL. We have a baseline model $q$ trained on the persuasion task, and initialize $p_{\theta}$ (the model being refined) with $q$. For each iteration, we sample one dialogue $d$ from the training corpus. For each turn in $d$, $p_{\theta}$ generates $n$ response candidates. Since persuasion strategies such as emotion appeal are found effective in human persuasion conversations (Wang et al., 2019), to encourage more persuasion strategies, we classify the candidates into “Non-Strategy” or “Strategy” with a dialogue-act classifier. Then...
the Response Detector (described later) annotates each candidate with status $a_i \in \{\text{Human Response, Pass/\text{Strategy}, Pass/\text{Non-Strategy}, Repetition, Inconsistency}\}$. With the detected status, candidates receive different rewards based on the following conditions, 1) if it is a ground truth human response (highest reward), 2) if it contains persuasion strategy (medium reward), 3) if it is a repetitive or inconsistent response (lowest reward). The reward values are chosen based on the validation dataset performance and the reward function details for the donation task are in Section A.1. By optimizing the rewards, $p_\theta$ learns from its own repetitive and inconsistent mistakes and generates more diverse, consistent and persuasive responses. Note that although we choose repetition and inconsistency in our persuasion task, DialGAIL is not specific to reducing repetition and inconsistency only. Given corresponding response quality detectors, it can be generalized to improve other sentence-level qualities as well (e.g., naturalness, positive sentiment).

**Algorithm 1 DialGAIL**

1: **Initialize**: Collect human-human dialogues $\mathbb{D}$ 
   Train $q$ with MLE on $\mathbb{D}$ 
   Warm-up $p_\theta$ with $q$, i.e., $p_{\theta_0} = q$ 
   Initialize the Replay Buffer $\mathbb{B}$
2: for $i=1, 2, 3, \ldots$ do
3:   Sample one dialogue $d$ from $\mathbb{D}$
4:   for each turn in $d$ do
5:     $c =$ context, $s^* =$ human response
6:     $p_{\theta_i}$ generates $n$ candidates $\mathbb{S} = \{s_1, s_2, \ldots, s_n\}$
7:     Response Detector annotates $\mathbb{S}$ with corresponding status $\mathbb{A} = \{a_1, a_2, \ldots, a_n\}$
8:     Put the triplet $(c, \{s^*\} \cup \mathbb{S}, \{\text{“Human Response”}\} \cup \mathbb{A})$ into $\mathbb{B}$
9:     Continue the dialogue with $s^*$
10: end for
11: Collect rewards for triplets in $\mathbb{B}$
12: Normalize the collected rewards
13: Update $p_{\theta_i}$ with Eq. (2), and clear $\mathbb{B}$
14: end for

The next step is to train with DialGAIL. To stabilize the RL training process, we apply proximal policy optimization (PPO) (Schulman et al., 2017) following Wu et al. (2021a). PPO performs importance sampling with the likelihood ratio between current and old policies $r(\theta) = \frac{p_{\theta_i}(s|c)}{p_{\theta_{i-1}}(s|c)}$, and optimizes the surrogate in Eq. (1) to maximize the expected rewards. To ensure the generation quality, we use the KL divergence between the language model being refined $p_\theta$ and the baseline $q$ as the maximum entropy regularizer in RL. This KL-term prevents $p_\theta$ from moving too far away from the original model $q$ and potentially losing fluency.

The final objective is shown in Eq. (2), $s$ is the generated response and $s^*$ is the human response:

$$L_{\text{policy}}(\theta) = \min(r(\theta) \hat{A}_{s^*}, \text{clip}(r(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_{s^*}))$$

(1)

$$L(\theta) = \mathbb{E}_{s \sim p_{\theta}(|c)} [L_{\text{policy}}(\theta) + \beta D_{KL}(q | p_\theta)]$$

(2)

### 3.1.2 Repetition and Inconsistency Detection

**Profile Builder**: To apply DialGAIL, we need to detect the repetitive and inconsistent candidates. Previous methods treated this as a classification problem and required manual annotation of the inconsistency status (Welleck et al., 2019). But manual annotations are expensive, and do not generalize across domains. Here we propose to build separate Profiles for both the user and the system to track key contextual information and detect the repetition and inconsistency more automatically. These profiles store $<\text{key}: \text{value}>$ pairs and are dynamically updated as the conversation unfolds. They are similar to dialogue state in task-oriented dialogues, with the key difference that we track both the system and the user in strategic dialogue settings to avoid contradiction with the system’s previous statements. In our task, experts analyze the human-human conversations and design an ontology with high-frequency questions such as “Do you have kids” (have_kids) as the keys in the profiles. For simplicity, we only track five attributes in the top grey table in Figure 1, but ideally new attributes should be added as the conversation continues and we leave this as future work. The Profile Builder uses dialogue-act classifiers to build and update the profiles. For example, if the last system-act is “propose-donation” and the following user-act is “disagree-donation”, the user profile is updated with “<want_to_donate: No>”.

The dialogue-act classifiers use GPT2-small and achieve 0.66 in F1 for system-act and 0.62 for user-act.

**Repetition Detector**: One key observation is that MLE-based baseline language models tend
to repeat high-frequency sentences in the training corpus and usually repeat on the exact lexical level. Therefore, we calculate the Jaccard similarity coefficient between each context sentence \( s_{ctx} \) and each candidate \( s_{cdd} \): 

\[
\text{Ratio}_{\text{rep}}(s_{ctx}, s_{cdd}) = \frac{\text{Unigram}_s \cap \text{Unigram}_{ctx}}{\text{Unigram}_s \cup \text{Unigram}_{cdd}},
\]

as the repetition ratio after normalizing the text. If \( \text{Ratio}_{\text{rep}} \geq 0.5 \), this candidate is considered as repetition. We experimented with other similarity metrics such as sentence embedding (Reimers and Gurevych, 2019) and found that Jaccard similarity is the simplest but the most effective one without much computation overhead, because repetition usually happens on the lexical level in our persuasion task. Such simple detection is also task-independent and can be very easily generalized to other domains. In our final model, 9.0% candidates are labeled as “Repetition”. More details of the repetition detector are in the Appendix.

**Inconsistency Detector.** To detect inconsistency, we apply the Profile Builder on each candidate, extract the value for each key, and compare them against the current Profiles. If the value extracted from the candidate contradicts the current Profiles, it is detected as “Inconsistency”. For example, the candidate “Thanks for your donation” in pink in Figure 1 implies that the user \texttt{want_to_donate:Yes}, which contradicts \texttt{want_to_donate:No} in the current USR Profile and makes it an inconsistent candidate. In our experiments, 6.6% candidates are inconsistent. We also trained a model on the Dialogue Natural Language Inference (DNLI) dataset (Wellleck et al., 2019) to detect inconsistency. However, the DNLI model’s performance is limited, possibly because DNLI is annotated on the PersonaChat (Zhang et al., 2018), which is very different from our persuasion task. We plan to explore domain-adaptation methods (Qian and Yu, 2019) to improve the inconsistency detector in the future.

### 3.3 Imitate with Human Demonstration

Besides being nonrepetitive and consistent, a good response also needs to move the conversation forward towards the task goal to persuade people to donate. However, intellectual activities such as persuasion or negotiation are difficult to quantify and optimize without imitation. Therefore, we perform behavior cloning (Bain and Sammut, 1995) and ask humans to demonstrate the persuasion process for the model to imitate. One human expert was employed to interact with our model for 10 conversations and was presented \( n = 10 \) candidates for each turn. Since it is subjective to determine each candidate’s persuasive level, to avoid bias towards different persuasive messages, the human expert was asked to select all acceptable responses given the context, rather than rating or ranking the candidates, which made the process easier and faster. In total, we collected 1,077 utterances (861 for training, 216 for validation) with binary labels (0 = not selected, 1 = selected) from the expert, with the labor time being only 3 hours. We didn’t employ more people in this process because we wanted to explore the potential of human demonstration. The experiments show that even with such small amount of data collection effort, human demonstration still helps significantly.

With the human demonstration data, we build the Response Imitator, a binary classifier to imitate the human selection process. It takes in all “Pass” candidates that pass the Response Filter and decide if a particular candidate is persuasive and should be selected. This classifier achieves 79.4% in accuracy on the validation set. In our final model, 60.1% candidates are selected.

It is worth noting that the Response Imitator is fundamentally different from the “next sentence prediction” (NSP) classifier used in many studies (Devlin et al., 2019; Wolf et al., 2019). Previous research shows that NSP doesn’t help much in dialogue generation (Li et al., 2020b), partly because in NSP, random sentences from the training data are assigned as negative examples. But in our response selection setting, the negative examples are generated by the language model under the same context, and therefore are semantically much closer to each other and much harder to distinguish. This makes the Response Imitator help more than the auxiliary NSP task in dialogue response generation.
even with small amount of human effort.

4 Experiments

4.1 Dataset

We conduct our experiments on the PERSUASION-FORGOOD dataset (Wang et al., 2019). It has 1,017 rich human-human persuasion conversations, where one user persuades the other user to donate to Save the Children.1 In the human-human setting, the average donation is $0.35 with a persuadee donation probability of 0.54. Basic statistics of the dataset is shown in Table 5 in the Appendix.

4.2 Baselines

MISSA (Li et al., 2020b) is a transformer-based dialogue model (Wolf et al., 2019) for strategic tasks with human-designed response filters, and jointly trains three tasks (language modeling, dialogue-act prediction and next sentence prediction).

ARDM (Wu et al., 2021b) uses two GPT2-medium models to model the user and the system separately, and jointly trains them to better capture different speakers’ language styles. It achieves state-of-the-art results on the persuasion task, so we initialize $p_0$ with ARDM and refine it with DialGAIL.

4.3 Evaluation Metrics

We evaluate the models from two aspects: response quality (measured by nonrepetitiveness, consistency, and fluency) and persuasion outcome (measured by persuasiveness, donation amount and donation probability). We conduct both automatic and human evaluations to assess the models.

**Automatic Metrics.** We use perplexity (PPL) to measure the models’ generation quality. To evaluate the candidate quality, we estimate the models’ probability to run out of candidates (OOC), the percentage of candidates that 1) are nonrepetitive and consistent and thus pass the Response Filter (Pass); 2) are persuasive and selected by the Response Imitator (Slct.); 3) have persuasion strategies (Strag.), and also the average sentence length (Len.).

**Human Evaluation.** We deployed the persuasive dialogue models on Amazon Mechanical Turk with ParlAI (Miller et al., 2017) to interact with human users. Each model interacted with 50 unique users to persuade them to donate part of their task earnings to Save the Children. Each user was allowed to do the task only once to avoid bias. After the conversation, the users were asked to input their donation amount (Dnt.) privately, and rate the conversation on nonrepetitiveness (Nonrep.), consistency (Const.), fluency (Fluc.), persuasiveness (Pers.), and overall experience (All.) on five-scale. Higher scores indicate better performances. We estimated the donation probability (DntP.) with the percentage of people who made a donation.

4.4 Quantitative Results

The automatic and human evaluation results are shown in Table 1 and 2 respectively. PersRFI refers to our final model refined with DialGAIL (R) plus Response Filter (F) and Response Imitator (I); PersRFI - RL refers to PersRFI minus refining with RL, which uses the baseline ARDM with the Response Filter and the Response Imitator. PersRFI - RL - Demo refers to PersRFI without RL refining and human demonstrations to train the Response Imitator, which is ARDM with the Response Filter only. We performed one-tailed t-test between ARDM and our three models.

In automatic evaluation in Table 1, we find that refining the model with DialGAIL achieves a lower perplexity (12.38 vs 12.45), indicating a better generation quality compared to the MISSA and ARDM baselines. PersRFI also generates more candidates with persuasion strategies than ARDM (51.2% vs 49.2%). Furthermore, PersRFI encourages longer generation and increases the average sentence length from 15.03 to 19.89 significantly.

In human evaluation in Table 2, PersRFI outperforms all the baselines on all metrics. For response quality, it achieves the highest consistency score (4.17) and fluency score (4.41). For persuasion outcome, it also receives the highest persuasiveness score (2.98) with a significantly higher average donation ($0.53) than the baselines. The donation amount and donation probability are even higher than the human results in PERSUASION-FORGOD (average donation=$0.35, donation probability=0.54). We notice that the persuasiveness scores of all models are relatively low compared to other metrics, indicating that persuasion is indeed a hard task and worth studying. All these results suggest that applying DialGAIL to refine the language model and imitating human demonstration to select the response are effective on all levels.

We report the Ablation study results in the lower half of Table 1 and 2, and find Response Filter alone (PersRFI - RL - Demo) doesn’t improve the model much, probably because the candidates

---

1https://www.savethechildren.org/
### Table 1: Automatic evaluation results. OOC: Out-of-candidate. Pass: Good candidates that are nonrepetitive and consistent and therefore pass the Response Filter. Slct.: Candidates with persuasion strategies. The baselines only generate one response, so metrics that involve multiple candidates such as OOC do not apply and are left blank. *p<0.05, **p<0.01.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MISSA (Li et al., 2020b)</td>
<td>19.91</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>47.6%</td>
<td>16.62</td>
</tr>
<tr>
<td>ARDM (Wu et al., 2021b)</td>
<td>12.45</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>49.2%</td>
<td>15.03</td>
</tr>
<tr>
<td>MISSA (Li et al., 2020b)</td>
<td>19.38</td>
<td>0.2%</td>
<td>84.4%</td>
<td>60.1%</td>
<td>51.2%</td>
<td>19.36**</td>
</tr>
<tr>
<td>ARDM (Wu et al., 2021b)</td>
<td>12.17</td>
<td>0.4%</td>
<td>85.3%</td>
<td>59.2%</td>
<td>49.6%</td>
<td>18.29**</td>
</tr>
<tr>
<td>PersRFI (Ours)</td>
<td>3.50</td>
<td>4.17</td>
<td>4.41</td>
<td>2.98**</td>
<td>3.61</td>
<td>0.53*</td>
</tr>
<tr>
<td>PersRFI - RL (w/o RL)</td>
<td>3.78**</td>
<td>3.98</td>
<td>4.37</td>
<td>2.72</td>
<td>4.11*</td>
<td>0.62**</td>
</tr>
<tr>
<td>PersRFI - RL - Demo (w/o RL w/o Demo)</td>
<td>3.25</td>
<td>3.84</td>
<td>4.39</td>
<td>2.73</td>
<td>3.75</td>
<td>0.38</td>
</tr>
</tbody>
</table>

### Table 2: Human evaluation results. Nonrep.: Nonrepetitiveness. Const: Consistency. Fluc.: Fluency. Pers.: Persuasiveness. All.: Overall experience. Dnt.: Average donation. DntP.: Donation probability. *p<0.05, **p<0.01.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MISSA (Li et al., 2020b)</td>
<td>-</td>
<td>3.78</td>
<td>3.74</td>
<td>-</td>
<td>-</td>
<td>$0.41</td>
<td>0.50</td>
</tr>
<tr>
<td>ARDM (Wu et al., 2021b)</td>
<td>3.17</td>
<td>3.95</td>
<td>4.17</td>
<td>2.33</td>
<td>3.61</td>
<td>$0.33</td>
<td>0.50</td>
</tr>
<tr>
<td>MISSA (Li et al., 2020b)</td>
<td>3.50</td>
<td>4.17</td>
<td>4.41</td>
<td>2.98**</td>
<td>4.0</td>
<td>$0.53*</td>
<td>0.61</td>
</tr>
<tr>
<td>ARDM (Wu et al., 2021b)</td>
<td>3.78**</td>
<td>3.98</td>
<td>4.37</td>
<td>2.72</td>
<td>4.11*</td>
<td>$0.62**</td>
<td>0.72*</td>
</tr>
<tr>
<td>MISSA (Li et al., 2020b)</td>
<td>3.25</td>
<td>3.84</td>
<td>4.39</td>
<td>2.73</td>
<td>3.75</td>
<td>$0.38</td>
<td>0.57</td>
</tr>
</tbody>
</table>

that pass the filter are still randomly selected and therefore not persuasive. However, Response Imitator makes significant contributions to reducing repetition and improving the overall experience, and also obtains the highest average donation amount ($0.62) and the highest donation probability (0.72). This confirms that even small amount of human demonstrations can be very helpful in accomplishing complex tasks such as persuasion. Finally, adding RL further improves the model’s persuasiveness (2.98 vs 2.72) and consistency (4.17 vs 3.98), decreases the out-of-candidate (OOC) probability (0.2% vs 0.4%) and leads to longer candidates (19.36 vs 18.29) with more strategies (51.2% vs 49.6%), indicating a better generation quality.

### 4.5 Qualitative Results

For qualitative evaluation, we present two dialogues examples from PersRFI and PersRFI - RL in Table 3. The top dialogue from PersRFI received all five ratings with a donation of $0.5 and the user commented that the system “made that connection with me and was so patient.” The responses with persuasion strategies are highlighted. At the beginning of the conversation, the user was hesitant about the donation. Then the model started to persuade with various strategies. It first provided more detailed information about the organization (credibility appeal), then tried to arouse the user’s feelings (emotion appeal), proposed a small donation request (foot-in-the-door) afterwards, and eventually successfully persuaded the user to make a donation. Compared to PersRFI, the bottom dialogue from PersRFI - RL have shorter responses with fewer strategies; after the user rejected the donation, the model didn’t try hard to persuade with different strategies and led to $0 donation. These results qualitatively show that PersRFI is able to generate richer, more coherent, and consistent responses with different persuasion strategies. There are more dialogue examples from other models in Section A.3 in the Appendix.

### 5 Discussion and Future Work

The proposed PersRFI framework involves two major steps: 1) refine a baseline model with DialGAIL, and 2) imitate only small amount of human demonstrations. While previous RL approaches focused more on token-level generation, DialGAIL infuses sentence-level qualities into the reward function and therefore may be used to improve sentence-level qualities beyond repetition and inconsistency. This gives task designers the freedom to design and plug in customized task-specific detectors into the PersRFI framework. Powered by the generalizable DialGAIL and small effort in human demonstration collection, PersRFI can be easily generalized to other dialogue tasks. In our persuasion task, the Inconsistency Detector still requires some manual
work on designing the profile ontology. We plan to apply dialogue relation extraction models (Yu et al., 2020) and reading comprehension (Sun et al., 2019) models to extract high-frequency questions to further automate this process in the future.

### 6 Conclusions

Persuasion dialogue system is an important topic in as it measures the machine’s ability to take strategic actions in conversations towards a persuasion goal. But the current conversational systems still suffer from repetition, inconsistency and task-oblivious responses, which will hinder the persuasion success. To address these issues, we propose DialGAIL to refine a baseline language model and extract a policy directly from the data without user simulators by learning from its own mistakes. Moreover, to better accomplish the persuasion task, we provide human demonstration for the model to imitate human persuasion activity. Experiments show that our PersRFI framework achieves state-of-the-art performance in a donation persuasion task, and produces more diverse, consistent, and persuasive conversations with small amount of human

---

**Table 3:** Dialogues from PersRFI and PersRFI - RL with ratings. PersRFI attempts to persuade with various strategies; utterances with strategies are highlighted (in the order of credibility appeal, emotion appeal and foot-in-the-door). Compared to PersRFI, the responses from PersRFI - RL are shorter with fewer persuasion strategies.

<table>
<thead>
<tr>
<th>Rating:</th>
<th>nonrepetitiveness=5, consistency=5, fluency=5, persuasiveness=5, overall=5, donation=$0.5.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model:</strong> PersRFI</td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rating:</th>
<th>nonrepetitiveness=5, consistency=5, fluency=4, persuasiveness=5, overall=5, donation=$0.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model:</strong> PersRFI - RL (without RL)</td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
<tr>
<td></td>
<td>USR</td>
</tr>
<tr>
<td></td>
<td>SYS</td>
</tr>
</tbody>
</table>

---
efforts. Looking into the future, strategic dialogues with both task and social contents will become more and more important, and it is our sincere hope that this work could inspire more research and discussion in strategic dialogue tasks such as persuasion and negotiation in the community.

7 Ethical Considerations

Persuasion is a double-edged sword and has been used for both good and evil. Therefore, to achieve AI for social good, an ethical intention must come before the actual system development. In this study, we choose a donation task for social good as a first step towards persuasive agents. At task completion, we collected a donation of $98.76 for Save the Children. Second, the lack of world knowledge remains a challenge for generative models and could lead inaccurate information, e.g., the underlined utterance in Table 3 is not accurate, and thus we must perform more fact-checking in the future. Furthermore, in real human-computer interactions, it is important to inform the users of the agent’s identity. Therefore, we conveyed the chatbot identity and the persuasion research purpose to the users clearly at the end of every conversation, and provided options for the users to directly communicate with the human team behind the system for any questions.

Acknowledgments

This work was supported by an Intel research gift. We thank many excellent Mechanical Turk contributors for participating in our task.

References


Yu Li, Kun Qian, Weiyan Shi, and Zhou Yu. 2020b. End-to-end trainable non-collaborative dialog system. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second In-


A Appendix

A.1 Training Details

Reward Function Details The reward function is shown in Eq. (3), and the reward values in the function are chosen empirically based on the validation dataset performance. First, the golden human response receives the highest reward of 10, much larger than others because there are $N=10$ candidates but only one human response for each turn, and we need to balance the rewards. Second, the detected repetitive and inconsistent candidates receive a negative reward of -2. Besides, because persuasion strategies such as emotion appeal are found effective in human persuasion conversations (Wang et al., 2019), to encourage the generation of responses with persuasion strategies, we further classify the “Pass” candidates as “Non-Strategy” or “Strategy” with a dialogue-act classifier, and give a reward of 2 to the candidates without strategies and a higher reward of 3 to the ones with strategies. A constant penalty of -3 is applied to sentences longer than 50 tokens. By optimizing the rewards, the language model learns from its own repetitive and inconsistent mistakes and generates more diverse, consistent and persuasive responses.

$$R_s = \begin{cases} 10 & s \in \text{Human Responses} \\ 3 & s \in \{\text{Pass} \land \text{Strategy}\} \\ 2 & s \in \{\text{Pass} \land \text{Non-Strategy}\} \\ -2 & \text{otherwise} \end{cases} \quad (3)$$

Repetition Detector details If $\text{Ratio}_{\text{rep}} \geq 0.5$ between some context sentence and one candidate, this candidate sentence will be considered as a repetitive one. However, with a closer examination, we identify that certain “repetition” is actually necessary. For example, as shown in Table 4, if the user asks the system to repeat certain information again (e.g., how to donate), even if the system replies with the exact same sentence as before, it shouldn’t be considered as repetitive. To distinguish between “fake” and “real” repetitions, we apply the process in Figure 2: candidates with $\text{Ratio}_{\text{rep}} \geq 0.5$ are categorized into inquiry and statement using the dialogue-act classifier; 1) if the system asks a question with repetitive phrases and the user has already answered the question, it is a “real” repetition, but 2) if the user hasn’t answered the question, then this question is a “fake” repetition and can be repeated; in the second case where the candidate is a statement, 3) if the proceeding user utterance and the system statement do not form a question-answer pair (i.e. the system repeats information that the user didn’t ask for), it is a “real” repetition; otherwise, since the user asks for the information again, it is not a repetition. After this process, 9.0% candidates in our model are labeled as “Repetition”. Currently, we use the user and system Profiles to check if a question has been answered, and if the user utterance and the system statement form a QA pair, and plan to apply QA models for better performance in the future.

<table>
<thead>
<tr>
<th>Role</th>
<th>Utterance</th>
</tr>
</thead>
<tbody>
<tr>
<td>USR</td>
<td>How can I donate?</td>
</tr>
<tr>
<td>SYS</td>
<td>The donation will be directly deducted from your task payment.</td>
</tr>
<tr>
<td>USR</td>
<td>Can you remind me again how to donate?</td>
</tr>
<tr>
<td>SYS</td>
<td>The donation will be directly deducted from your task payment.</td>
</tr>
</tbody>
</table>

Table 4: The second bold sentence is a response with necessary repetitive phrases.

![Figure 2: The procedure to detect real repetitions.](image)

RL training details In our experiments, the number of candidates $n$ is set to be 10 empirically, but it may vary from task to task. RL training process can be unstable and delicate. Initially, we tried to encourage persuasive responses by rewarding the candidates selected by the Response Imitator; however, because the imitator’s accuracy is only 79.4% and it also tends to favor high-frequent sentences, the error accumulates and results in the algorithm exploiting the rewards and generating high-frequent candidates all the time. Therefore, we chose to reward the “Pass” candidates only, with the observation that more “Pass” candidates would lead to more persuasive utterances. Besides, we found that in spite of the KL constraint, the more steps we train, the further $p_\theta$ moves, and this causes...
the model’s validation perplexity to decrease first and then increase. Therefore, we only trained the model for 35 epochs (i.e. 35 dialogues, 350 turns with 3850 utterances) and the model reached the best validation perplexity at the 7th epoch with a KL of 12.59. The change in rewards with the training steps is shown in Figure 3. Adam (Kingma and Ba, 2015) was used for optimization with an initial learning rate of 2e-5.

Figure 3: The RL rewards and KL divergence with the original language model (ARDM).

### A.2 Dataset Details

<table>
<thead>
<tr>
<th>PERSUASIONFORGOOD</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Dialogues</td>
<td>1,017</td>
</tr>
<tr>
<td>Average Turns per Dialogue</td>
<td>10.43</td>
</tr>
<tr>
<td>Average Tokens per Utterance</td>
<td>19.36</td>
</tr>
<tr>
<td>Average Donation</td>
<td>$0.35</td>
</tr>
<tr>
<td>Donation Probability</td>
<td>54%</td>
</tr>
</tbody>
</table>

Table 5: Dataset Statistics of the PERSUASIONFORGOOD dataset.

### A.3 More dialogue examples

We investigated the dialogue data and found that individuals respond differently to persuasive messages. Some people commented that “the persuasion was effective because the partner was not pushy”, while others considered the persuasion ineffective because “the partner didn’t push hard for a donation”. Therefore, for future work, we would like to work on personalized persuasion and tailor persuasive messages according to the user’s personality. Table 6, 7, 8 and 9 show dialogue examples from different models.
If I had to make a decision right now, I would say I would donate $0. It would be so nice to have that. Just to confirm, how much would like to donate to the charity now?

Hey, I’m doing good.

They are a non-governmental organization that promotes children’s rights, provides relief and helps support children in developing countries. Do you want to make a donation to Save the Children? You can choose any amount from $0 to all your task payment ($2) to donate, which will be directly deducted from your payment.

Yes children are our future! And truth be told no one deserves to go to bed hungry.... Looks like the task is over, but enjoy the rest of your day :-)

I agree, I need more people like you!

Thanks. Take care.

Thank you so much, have a wonderful day.

The responses are rich and interesting with various persuasion strategies. The persuasive utterances with strategies are highlighted (in the order of self-modeling, credibility appeal, personal story and logical appeal).

**Table 6:** Another dialogue example from our PersRFI model. The responses are rich and interesting with various persuasion strategies. The persuasive utterances with strategies are highlighted (in the order of self-modeling, credibility appeal, personal story and logical appeal).

**Table 7:** Another dialogue example from our PersRFI - RL model. Compared to PersRFI, the responses are shorter and less persuasive with occasional inconsistency (e.g., “It is very easy to persuade people to donate”). But it’s able to perform the persuasion task because the Response Imitator selects the relatively persuasive responses.
Table 8: One dialogue example from our PersRFI - RL - Demo model. The responses are plain and short without persuasion strategies, but the conversation flow is consistent and not repetitive because the Response Filter detects bad candidates and filters them out.

Table 9: One dialogue example from the baseline ARDM. The sentences are very repetitive and not consistent with the context.