PGTask: Introducing the Task of Profile Generation from Dialogues

Anonymous ACL submission

Abstract

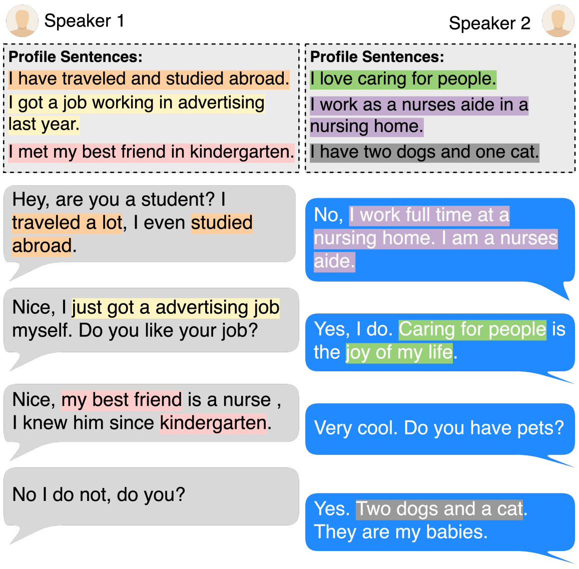
Recent approaches have attempted to personalize dialogue systems by leveraging profile information into models. However, this knowledge is scarce and difficult to obtain, which makes the extraction/generation of profile information from dialogues a fundamental asset. To surpass this limitation, we introduce the Profile Generation Task (PGTask). We contribute with a new dataset for this problem, comprising profile sentences aligned with related utterances, extracted from a corpus of dialogues. Furthermore, using state-of-the-art methods, we provide a benchmark for profile generation on this novel dataset. Our experiments disclose the challenges of profile generation, and we hope that this introduces a new research direction.

1 Introduction

Building conversational systems that mimic human attributes has always been a long-term goal in Natural Language Processing (NLP). Various works have attempted to leverage speaker profile information to improve the consistency of dialogue generation models (Wu et al., 2020; Xu et al., 2022; Cao et al., 2022). However, for dialogue systems, this sort of information is scarce and requires annotation efforts that are expensive to obtain, so there is a need to build methods that automatically gather this knowledge from dialogues.

Zhang et al. (2018) introduced PersonaChat, a dataset comprising a collection of profile sentences (*persona*) that reflect each speaker's individual characteristics and personal facts. These profiles serve as a knowledge base for promoting the consistency between utterances from speakers, and various recent dialogue models have incorporated this information using diverse techniques (Song et al., 2020, 2021; Cao et al., 2022).

Few works have attempted to infer profile information from PersonaChat dialogues. Gu et al. (2021) restructured PersonaChat and built the Persona Detection Task, where the goal was to retrieve the correct persona amongst a set of distractor personas. Although introducing an interesting research path, this task is limited to a set of pre-defined personas, which is not suitable for extracting profile sentences from unseen conversational data. Cao et al. (2022) also manipulate PersonaChat to incorporate model-agnostic personas into the dialogue generation task. Nevertheless, for the profile generation task, PersonaChat is structured in a profile-to-dialogue manner and lacks information about the corresponding profile sentence per turn, which may become a challenge when the task becomes extracting profile information from utterances.



Profile Sentences:
I have traveled and studied abroad.
I got a job working in advertising last year.
I met my best friend in kindergarten.

Profile Sentences:
I love caring for people.
I work as a nurses aide in a nursing home.
I have two dogs and one cat.

Hey, are you a student? I traveled a lot, I even studied abroad.

No, I work full time at a nursing home. I am a nurses aide.

Nice, I just got a advertising job myself. Do you like your job?

Yes, I do. Caring for people is the joy of my life.

Nice, my best friend is a nurse , I knew him since kindergarten.

Very cool. Do you have pets?

No I do not, do you?

Yes. Two dogs and a cat. They are my babies.

Figure 1: An example dialogue where each turn contains the corresponding profile sentence.

In this work, we introduce the PGTask, where the goal is to generate profile sentences given speaker utterances. For this, we create a new dataset, the Profile Generation Dataset (PGDataset)[1], which

[1] Dataset and code are available at https://tinyurl.com/PGDatasetSD.

relates utterances with the respective profile sentences upon the existing PersonaChat corpus. In Figure 1, we can observe several examples of relations between profile sentences and the corresponding speaker's utterance. Notice, however, that the task is more challenging than just finding, within the dialogues, utterances that highly relate to each profile sentence. For instance, the profile sentence "I like all genres of music." is probably at the origin of the utterance "Yes, sometimes I also listen to classical music.", but we cannot extract that profile sentence from that single utterance (the goal of PGTask).

We framed our problem as an entailment classification task and, after human feedback, we reached the final PGDataset. Finally, we provide results from three state-of-the-art models trained and evaluated in the proposed dataset.

2 Building PGDataset

2.1 Modeling Entailment Relations

In the Natural Language Inference (NLI) task, the goal is to classify the relationship between a pair of premise and hypothesis sentences into three classes: entailment (E), neutral (N) and contradiction (C). Welleck et al. (2019) extended the NLI task to the dialogue setting and introduced the Dialogue Natural Language Inference (DNLI) dataset, where the input sentences consist of dialogue utterances from PersonaChat. We adopt this procedure and train a model $\mathcal{M}^{NLI}$ to identify the correct profile sentences for each utterance in a dialogue.

Consider two sentences s_i and s_j that are concatenated into the input $x = \{s_i, s_j\}$. First, we utilize RoBERTa (Liu et al., 2019) to obtain a hidden representation h from the input x. Then, we include a $\mathrm{softmax}$ classifier on top of RoBERTa to obtain the probability distribution over the set of possible classes. Formally, we obtain the probability of label $y \in \{C, N, E\}$ with:

$$\begin{aligned} h &= \text{RoBERTa}(x), \\ p_{\mathcal{M}^{NLI}}(y|x) &= \text{softmax}(Wh), \end{aligned} \tag{1}$$

where W is the learnable parameter matrix from the classification layer. We fine-tune both RoBERTa and W parameters by maximizing the log-probability of the correct label.

We experiment with two different settings where we fine-tune RoBERTa only on DNLI and on MNLI (Williams et al., 2018), a benchmark multi-genre NLI dataset, and DNLI datasets for better generalization. Details are provided in Appendix A. Table 1 shows the results on the test set, where the latter achieves higher accuracy and is selected as the annotation model.

Datasets	Accuracy (%)
DNLI	91.24
MNLI + DNLI	**91.75**

Table 1: Accuracy of fine-tuned RoBERTa for the test set of DNLI.

2.2 Dataset Annotation

In PersonaChat (Zhang et al., 2018), each dialogue carries a set of profile sentences for both speakers. Consider a set of n utterances from a speaker, $U = \{u_1, u_2, ..., u_n\}$, a set of k profile sentences $P = \{p_1, p_2, ..., p_k\}$ from the same speaker, and the dialogue NLI model from Section 2.1. Then, at time step t, we can determine one or more profile sentences s_t related to utterance u_t using:

$$\begin{aligned} s_t = \{p_i \in P : \\ &\operatorname*{arg\,max}_{y \in \{C,N,E\}} (p_{\mathcal{M}^{NLI}}(y|\{u_t, p_i\}) = E\}. \end{aligned} \tag{2}$$

In Equation 2, the profile sentences are gathered by considering the entailed cases between the utterances and the profile sentences, where each utterance could be associated with more than one profile sentence. In Table 2, we provide an extract from the PGDataset.

Utterance	Profile Sentences
I enjoy hanging with my mother she is my best friend.	My mom is my best friend.
I am almost done, I only have two years left in law school.	I have got two more years in college.
	I study law.

Table 2: Two examples from PGDataset.

2.3 Human Annotations

In the profile generation task, the profile must represent a possible extraction from the dialogue utterance, and this correlation's direction between the utterance and the profile sentence must be valid.

To assess the quality of the automatic annotations from our model, we resort to human evaluation.

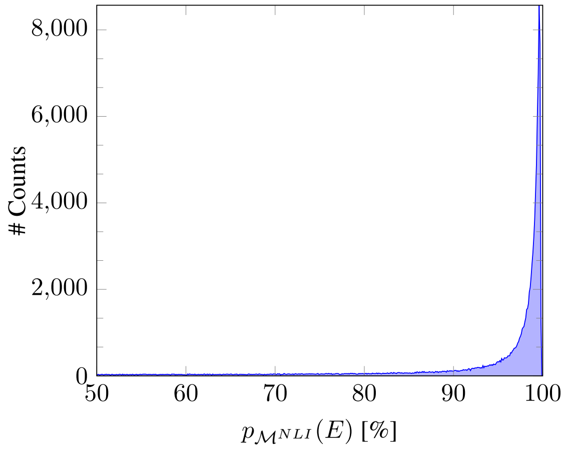


Figure 2: Distribution of the entailment class probability for the entailed cases ($\mu = 93.4$, $\sigma^2 = 1.10$).

For all the pairs classified as entailed in Equation 2, we measure the confidence by inspecting the $\mathrm{softmax}$ probability assigned to the entailment class. Our intuition is that a weak confidence when classifying a profile sentence as entailed corresponds to a weak or incorrect correlation and can be removed from the dataset. In Figure 2, we plot the distribution of the scores from the entailment class for all points obtained from Equation 2.

To determine if a higher confidence value corresponds to a correct example, we randomly select 100 samples from 3 intervals: $[50, 70]$, $]70, 90]$, and $]90, 100]$. We asked 3 expert annotators from our department to "mark with an X if the profile sentence could be extracted from the given utterance". The quality of the samples is measured by the number of marked samples by the annotators (accuracy). The agreement rate between annotators was 86.66% and the average accuracy for each interval was 8.33%, 12.33%, and 51.67%, respectively. The results obtained show that when the confidence of the model grows, the correlation between the profile sentence and the utterance also increases.

After inspecting the results from the annotators, we observed that most of the marked samples had more than 99% confidence. We asked for a second round of annotations with 100 samples but now only for samples with more than 99% confidence. The agreement rate between annotators was 91% and the average accuracy was 87,33%, a significantly higher score compared to the $]90, 100]$ interval. We decided, thus, that PGDataset only considers the samples which the model classified with more than 99% confidence.

Train	# Samples	34355
	Avg. Profile Sentences	1.06
	Avg. Utterance Words	13.13
	Avg. Profile Sentence Words	7.14
Valid	# Samples	4236
	Avg. Profile Sentences	1.06
	Avg. Utterance Words	13.36
	Avg. Profile Sentence Words	7.67
Test	# Samples	3760
	Avg. Profile Sentences	1.06
	Avg. Utterance Words	13.05
	Avg. Profile Sentence Words	7.17

Table 3: Dataset Statistics.

2.4 PGDataset Statistics

In Table 3, we provide the dataset statistics for the gathered samples.

3 Benchmarking the PGTask

In this task, the goal is to generate a profile sentence conditioned on an utterance. Transformer-based decoders have achieved substantial progress in various NLP tasks (Radford et al., 2019). We leverage these models and rely on a causal language modeling (CLM) objective for our profile generation task. More precisely, considering a sentence $s = \{w_1, ..., w_n\}$ composed of n words, in CLM, the maximum likelihood objective over s is:

$$\mathcal{L}_{CLM} = \sum_{i=1}^{n} \log P(w_i | w_1, ..., w_{i-1}). \quad (3)$$

For our task, we are only interested in calculating the loss for the words from the profile sentence conditioned on the utterance. Considering an utterance $u = \{w_1^u, ..., w_m^u\}$ and a profile sentence $p = \{w_1^p, ..., w_k^p\}$, we redefine the objective from Equation 3:

$$\mathcal{L}_{PG} = \sum_{i=1}^{k} \log P(w_i^p | w_1^u, ..., w_m^u, w_1^p, ..., w_{i-1}^p). \quad (4)$$

As seen in Equation 4, the loss is only calculated for the generation of the profile sentences. In the model's input, we separate the utterance and profile sentences using a special token <gen> and, as it can

	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore
W/o FT	`distilgpt2`	0,0559	0,0030	0,0000	0,0000	0,0686	0,0093	0,0580	0,8466
	`gpt2-small`	0,0487	0,0040	0,0000	0,0000	0,0608	0,0063	0,0520	0,8421
	`gpt2-medium`	0,0448	0,0020	0,0000	0,0000	0,0720	0,0031	0,0532	0,8328
W/ FT	`distilgpt2`	0,4442	0,1318	0,0560	0,0000	0,3568	0,1412	0,3539	0,9235
	`gpt2-small`	0,6130	0,3230	0,2062	0,0944	0,5007	0,2831	0,5000	0,9439
	`gpt2-medium`	0,5931	0,2594	0,1530	0,0917	0,4632	0,2414	0,4588	0,9476

Table 4: Generation results for models with and without fine-tuning (FT) on the PGDataset. The results presented are the average score of 5 runs. The scores range between 0 and 1.

exist more than one profile sentence, we add `<sep>` between the profile sentences.

4 Experiments

In this section, we evaluate Transformer decoders on the novel dataset and provide benchmark results for future research. Additional experimental details are provided in Appendix B.1.

4.1 Models

GPT2 This model has achieved state-of-the-art results in various generation tasks (Radford et al., 2019). We consider two different pre-trained versions that differ in size, the `gpt2-small` and `gpt2-medium` (details in Appendix B.2).

DistilGPT2 This is a distilled version of GPT2, where it was trained under the supervision of GPT2 (Hinton et al., 2015). The `distilgpt2` contains about half the size of GPT2 while still achieving competing performance in various NLP tasks.

4.2 Metrics

We follow common practices for text generation and report BLEU (Papineni et al., 2002) and ROUGE (Lin and Hovy, 2002), metrics that, respectively, measure the precision and recall between the generated and the golden text. Additionally, we employ BERT Score (Zhang et al., 2019), an automatic metric that leverages BERT's (Kenton and Toutanova, 2019) contextual embeddings and matches words in candidate and golden sentences using cosine similarity.

4.3 Results

In Table 4, we provide benchmark results for the PGTask. We observe that fine-tuning the models has a great impact on the overall performance, where `gpt2-small` achieves the higher scores in all metrics except BERTScore (for a minimal difference). In Appendix B.3, we provide some generated examples from the evaluated models.

5 Related Work

Recent research has focused on building personalized dialogue systems using profile information. Li et al. (2016) proposed a neural conversational model to capture background information and speaking style from interlocutors in dialogue. Zhang et al. (2018) introduced a dataset composed of personas, which are essentially 3 to 5 profile sentences describing the speaker's profile. Zheng et al. (2019) studied how to include profile information such as age, location, and interests by explicitly incorporating this knowledge into the sequence-to-sequence framework.

Few works have attempted to identify profile knowledge from conversational data. (Gu et al., 2021) introduced a framework for detecting the correct profile amongst a set of distractor profiles. Nevertheless, the authors do not consider the correlation between utterances and profile sentences. (Cao et al., 2022) proposed a data manipulation method to construct distilled and diversified dialogue data containing profile information and leverage it into the dialogue generation task.

6 Conclusion

We propose the PGTask and contribute with PGDataset, a dataset with more than 30 000 pairs of utterances and profile sentences built with the feedback of human annotators. In addition, we train state-of-the-art models and achieve promising results in the proposed task. We hope that this new line of research will help the task of personalizing dialogues, although the task of automatically extracting profiles from dialogues is valuable by itself.

References

Yu Cao, Wei Bi, Meng Fang, Shuming Shi, and Dacheng Tao. 2022. A model-agnostic data manipulation method for persona-based dialogue generation. In

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 7984–8002.

Jia-Chen Gu, Zhenhua Ling, Yu Wu, Quan Liu, Zhigang Chen, and Xiaodan Zhu. 2021. Detecting speaker personas from conversational texts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1126–1136.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and William B 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)*, pages 994–1003.

Chin-Yew Lin and Eduard Hovy. 2002. Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 45–51.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Cite arxiv:1907.11692.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Haoyu Song, Yan Wang, Kaiyan Zhang, Weinan 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.

Haoyu Song, Yan Wang, Weinan Zhang, Xiaojiang Liu, and Ting Liu. 2020. Generate, delete and rewrite: A three-stage framework for improving persona consistency of dialogue generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5821–5831.

Sean Welleck, Jason Weston, Arthur Szlam, and Kyunghyun Cho. 2019. Dialogue natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3731–3741.

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.

Bowen Wu, MengYuan Li, Zongsheng Wang, Yifu Chen, Derek F Wong, Qihang Feng, Junhong Huang, and Baoxun Wang. 2020. Guiding variational response generator to exploit persona. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 53–65.

Xinchao Xu, Zhibin Gou, Wenquan Wu, Zheng-Yu Niu, Hua Wu, Haifeng Wang, and Shihang Wang. 2022. Long time no see! open-domain conversation with long-term persona memory. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2639–2650.

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, Melbourne, Australia. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yinhe Zheng, Guanyi Chen, Minlie Huang, Song Liu, and Xuan Zhu. 2019. Personalized dialogue generation with diversified traits. *arXiv preprint arXiv:1901.09672*.

A Fine-Tuning RoBERTa

We fine-tune a pre-trained `roberta-base`[2] (Liu et al., 2019) with 12 layers, 768 hidden units, 12 attention heads, and 125M parameters on 1 NVIDIA GeForce RTX 3080 to minimize the cross entropy. We use Adam (Kingma and Ba, 2014) optimizer with a learning rate of $5e^{-}5$. The batch size was 32, we train for 20 epochs and early stop after 5 epochs without an increase in the validation accuracy.

B Profile Generation

B.1 Experimental Details

We perform 5 runs for each model on 1 NVIDIA GeForce RTX 3080 using different seed values and calculate the average score for all metrics. Models are trained to minimize the cross entropy using Adam (Kingma and Ba, 2014) optimizer with a learning rate of $5e^{-}5$. For `gpt2-small` and `distilgpt2`, the batch size was 16 while for `gpt2-medium` the batch size was 4 with 4 gradient accumulation steps. We train for 20 epochs with early stopping where the training is stopped after 5 epochs without a decrease in the validation loss. We generate the profile sentences with a maximum length of 50 and perform greedy sampling, i.e., select the next word with the highest probability. All experiments are implemented using the HuggingFace[3] and PyTorch[4] libraries.

B.2 Models

The `gpt2-small`[5] version contains 12 layers, 768 hidden units, 12 attention heads, and 117M parameters and `gpt2-medium`[6] includes 24 layers, 1024 hidden units, 16 attention heads, and 345M parameters. The distilled version `distilgpt2`[7] (Hinton et al., 2015) is smaller than GPT2, where it is composed of 6 layers, 768 hidden units, 12 attention heads, and 82M parameters.

B.3 Generated Examples

As discussed in Section 4.3, the fine-tuned models show promising results in the generation of correct profile sentences. An example for that is presented in Table 5. Here, all models successfully extract the profile sentence, although we could argue that *loving* and *liking* are semantically different.

However, the evaluation scores also show that the systems are still far from always extracting the correct profiles. In Table 6, we show an example where all models generated unrelated profile information and failed to recognize that the speaker dropped off high school.

Dialogue Utterance: sorry, i do not like music, i like reading mystery books.
Golden Profile Sentences: i enjoy reading mysteries.
`distilgpt2`: i love to read mystery novels. **`gpt2-small`:** i love reading mysteries in my free time. **`gpt2-medium`:** i read mystery novels.

Table 5: Generated example #1 from fine-tuned models.

Dialogue Utterance: dropping out of high school was a bad idea. the landlord just called.
Golden Profile Sentences: i dropped out of high school.
`distilgpt2`: i just got a job at the elementary school in new england. **`gpt2-small`:** my parents got a new job. **`gpt2-medium`:** i just graduated high school.

Table 6: Generated example #2 from fine-tuned models.

[2] https://huggingface.co/roberta-base
[3] https://huggingface.co/
[4] https://pytorch.org/
[5] https://huggingface.co/gpt2
[6] https://huggingface.co/gpt2-medium
[7] https://huggingface.co/distilgpt2