

Prompting for a conversation: How to control a dialog model?

Josef Valvoda^{*} Yimai Fang David Vandyke
University of Cambridge Apple
jv406@cam.ac.uk
{yimai_fang, dvandyke}@apple.com

Abstract

Dialog modelling faces a difficult trade-off. Models are trained on a large amount of text, yet their responses need to be limited to a desired scope and style of a dialog agent. Because the datasets used to achieve the former contain language that is not compatible with the latter, pre-trained dialog models are fine-tuned on smaller curated datasets. However, the fine-tuning process robs them of the ability to produce diverse responses, eventually reducing them to *dull* conversation partners. In this paper we investigate if prompting can mitigate the above trade-off. Specifically, we experiment with conditioning the prompt on the query, rather than training a single prompt for all queries. By following the intuition that freezing the pre-trained language model will conserve its expressivity, we find that compared to fine-tuning, prompting can achieve a higher BLEU score and substantially improve the diversity and novelty of the responses.

1 Introduction

Prompting large language models (LLM) has recently demonstrated an impressive performance on a number of natural language processing (NLP) tasks such as machine translation (Radford et al., 2019), summarisation (Li and Liang, 2021) or question answering (Schick and Schütze, 2021a). Prompts are tokens which are appended or prepended to the input of a language model. They are employed to induce the model into generating useful information, while keeping the model weights frozen. Soft prompts, continuous trainable vectors prepended to the model input, have in particular proven useful for a number of tasks (Liu et al., 2021a). While requiring fine-tuning of only a relatively small number of parameters, they excel in a few-shot setting. Furthermore, as the underlying LLM’s grow in parameter size,

^{*}Work done while at Apple.

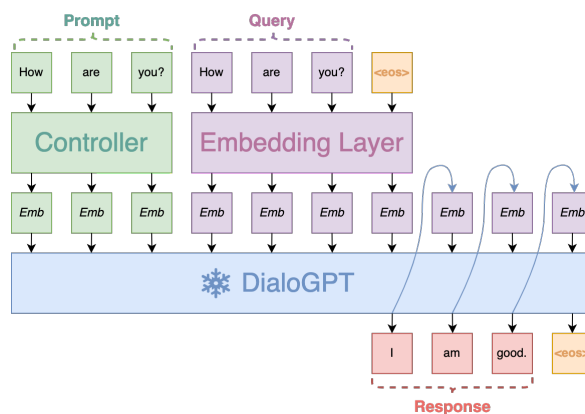


Figure 1: Dynamic prompt conditions the *prompt* on the *query* using the *controller*; a Transformer encoder.

they become competitive even in the full data setting (Lester et al., 2021). Simultaneously, prompts remove the burden of storing the full copy of a fine-tuned LLM for every task, which becomes increasingly useful as the LLM size grows. Crucially for us, prompting preserves the LLM parameters which should help retain their general language abilities for a downstream task.

Dialog modelling is the task of generating a response given the previous dialog turn, the query (Li et al., 2016b). Dialog models are typically trained using the maximum-likelihood estimation (MLE) objective. However, MLE-trained models have a high propensity to provide dull responses, such as “I don’t know” (Sordoni et al., 2015; Serban et al., 2016; Zhao et al., 2017). While state-of-the-art models, such as DialogPT (Zhang et al., 2020), can overcome this issue by training large models on massive amounts of data, a trade-off emerges. On the one hand, these models are expressive by virtue of the large datasets they are trained on. On the other hand, the same scale of training data and model parameters is responsible for the lack of control over the content of their responses.

Since dialog models are not useful without the ability to control their responses, in this paper, we

turn to prompting as a possible method of exerting such control. Instead of fine-tuning the entire model, we keep the weights of the model intact, tuning only the prompts (and word embeddings) in an effort to preserve the models’ expressivity. Furthermore, we develop a **dynamic-prompt**,¹ which conditions the prompt on the query in an effort to dynamically induce the response by having a different prompt for every turn of the conversation, see Fig. 1. The intuition is to separate the task of language generation, which LLM’s are very good at, from the task of selecting appropriate responses, which LLM’s struggle to learn from the limited examples they are fine-tuned on.

In our experiments with DailyDialog dataset, we find that the dynamic prompt outperforms both fine-tuning and the soft-prompting of DialoGPT in terms of a BLEU4 score. The best dynamic prompt model achieves 0.12 BLEU4 compared to 0.11 and 0.08 of fine-tuning and soft-prompting respectively. Furthermore, the dynamic prompt finds the best trade-off between the BLEU score and novelty as well as the diversity of responses, maintaining above 0.90 novelty and diversity as the BLEU4 score increases, while the other models can not. Finally, we find that dynamic-prompting on GPT-2 achieves the best BLEU4 results, improving over prompted DialoGPT by an additional 19% and casting doubt about the utility of dialog-specific pre-training when it comes to prompting.

2 Prompting for Dialog Generation

Fine-tuning a pre-trained model on a small dialog dataset degrades the novelty of the responses (Sordani et al., 2015). Prompting opens up a possibility of exerting control over the model behaviour, without touching the majority of the weights that might be useful for generating novel responses (Lester et al., 2021). However, typical prompting methods are restrained in using a single prompt for a single task. In this section, we motivate the dynamic prompt as a natural next step in generalising the prompting paradigm for dialog modelling.

¹Our work is concurrent to Gu et al. (2021) who develop a similar prompting method. However, our research is motivated by a different question from theirs. Specifically, we investigate how prompting can help with inducing creative responses, which we measure on the novelty and diversity metrics introduced below. Their work, on the other hand, focuses on improving performance on traditional metrics such as BLEU, NIST, METEOR and ROUGE-L.

Prompting. Auto-regressive models with billions of parameters trained on a language modelling objective, such as OpenAI’s GPT-3 (Brown et al., 2020), have demonstrated a strong few-shot performance without the need to update the model parameters. Instead of fine-tuning the model with target task examples, a manually designed prompt, for example in the form of a natural language sentence describing the task, is fed to the model to solicit the desired response. For instance, to induce a translation the model might be told to: *Translate English to French*. The description is followed up with pairs of examples of English sentences and their French translations. To derive a new translation the model is fed an English sentence alone and it is left to infer the French translation. However, not all human-designed prompts elicit the desired response. In practice ensembles of many prompts have gone some way in improving the performance, but still necessitate humans in the loop to design the said prompts (Schick and Schütze, 2021a).

Soft-prompting. To address this short-coming, recent work has found that prefixing and fine-tuning vocabulary tokens is a more expressive solution, which can learn the prompt from the data directly, without a need of a human prompt designer (Liu et al., 2021a). The soft prompts are not constrained by encoding existing tokens in the vocabulary and can freely encode parameters to facilitate the task. Empirically, soft prompts achieve better performance than their *hard* prompt counterparts and in a few-shot setting outperform fine-tuning of the entire model.

Dynamic-prompting. Soft prompts are restricted in utilising only a single prompt of a constant number of tokens for each task. We hypothesise that there are many tasks where the desired response for every input will be hard to solicit through a single shared general prompt. For example, the task of dialog generation has a general requirement to generate semantically and syntactically coherent, engaging responses. However, an individual response might have specific properties that are only relevant within the context of the query. While soft prompts are likely to solicit the more general tasks, such as machine translation or question answering, possibly because they exist in the training data to begin with, the more nuanced requirements might be heavily context-dependent. Therefore,

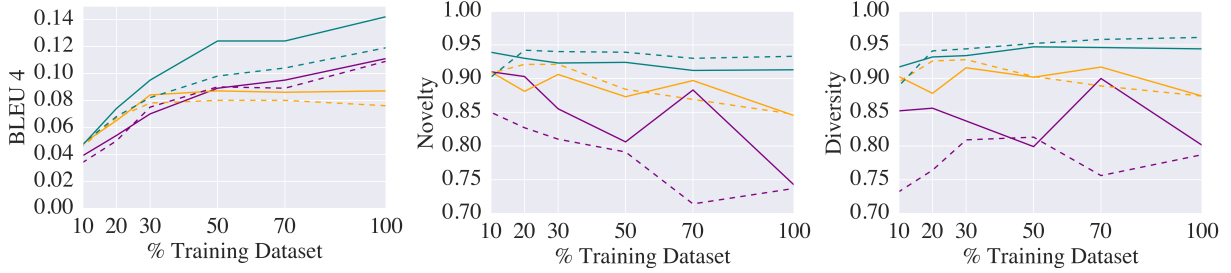


Figure 2: BLEU4, novelty and diversity scores for **fine-tuning**, **soft-prompting** and **dynamic-prompting**. Full line is GPT-2, dashed line is DialoGPT.

we condition the prompt on the context to enable learning different prompts for different queries. We describe the method in detail below.

3 Method

Notation. Let a dialogue be denoted as $x_1 \cdots x_N$ where N is the sequence length. We denote dialogue history as $S = x_1, \dots, x_m$ and target sequence as $T = x_{m+1}, \dots, x_N$. Now we can compute the conditional probability $P(T | S)$ as a product of a series of conditional probabilities:

$$p(T | S) = \prod_{n=m+1}^N p(x_n | x_1, \dots, x_{n-1}) \quad (1)$$

This probability can be parametrized by an autoregressive language model (in our case DialoGPT).

Fine-tuning. For our experimental baseline, we simply fine-tune pre-trained DialoGPT on the DailyDialog dataset, see § 4.

Soft-prompting. For soft-prompting, we prepend a prompt $F = z_1, \dots, z_m$ at the beginning of S . Prompt tokens are randomly initialised word embeddings appended to the DialoGPT vocabulary. In our experiments, we set the length of the soft prompt to the length of S . Now, to compute the probability of the target given the dialogue history and prompt, we compute:

$$p(T | S, F) = \prod_{n=m+1}^N p(x_n | z_1, \dots, z_m, x_1, \dots, x_{n-1}) \quad (2)$$

We instantiate the pre-trained DialoGPT, but this time we freeze all of the model parameters, except the word embedding weights. While in a typical soft-prompting experiment only the prompt embeddings are tuned, we found that for our dialog

setting the performance suffers considerably when the model is constrained to train only the parameters of prompt embeddings, which is why we relax this requirement. Furthermore, we do not calculate the loss for the logits corresponding to the prompt. This is because we don't know the ground truth of what the prompt should be.

Dynamic-prompting. Finally, we extend the soft-prompting paradigm by conditioning the prompt F on the query S to find a unique prompt for every query, see Fig. 1. We use an autoregressive Transformer encoder to generate the prompt embeddings:

$$\mathbf{h}_1, \dots, \mathbf{h}_m = \text{Transformer}(x_1, \dots, x_m) \quad (3)$$

Now, we use \mathbf{h} as our prompt token embeddings and prepend them to the query embeddings inside the DialoGPT model. The model is otherwise trained the same way as the soft-prompting model above. The Transformer encoder is trained jointly with the embeddings.

4 Experimental Setup

Datasets. We conduct our experiments on the DailyDialog dataset consisting of 13,118 dialogues, split into 11,118/1,000/1,000 training/validation/test sets (Li et al., 2017). We only focus on single-turn dialog modelling and process the dataset so that our pairs of queries and responses correspond to every two steps in a conversation. Inspired by Li and Liang (2021), we train our models on 10%, 20%, 30%, 50%, 70%, and 100% of the DailyDialog dataset to observe how the number of training samples affects the performance. We keep the validation and test sets full-sized for all experiments. For full experimental details, see App. B.

Metrics. We follow Li et al. (2017) and evaluate the models using BLEU4 score. Since we do not

want the model to simply repeat the responses it has memorised from the training data, i.e. the dull response issue, we additionally introduce two new metrics: **novelty** and **diversity**. We define novelty as the proportion of model outputs on the test set that are not found in the training set. Given a test set, diversity is defined as the number of unique model outputs divided by the total number of outputs. A good dialog model should be able to achieve a high BLEU4 score, while maintaining a high level of novelty and diversity in its responses.

Models. We experiment with two models, DialoGPT (Zhang et al., 2020), a state-of-the-art dialog model, and GPT-2 (Radford et al., 2019) the model DialoGPT is fine-tuned from. We choose the latter model to gain insight into how useful in-domain pre-training of DialoGPT is for prompting.

5 Results

Our main results are contained in Fig. 3. First, we compare the performance of prompting vs fine-tuning on the DialoGPT model. We find that under the full data setting the dynamic prompt model outperforms the soft prompt model and even the fine-tuned model on all three metrics. With a BLEU4 of 0.12 dynamic prompt is 9% better than fine-tuning (0.11) and 15% better than a soft prompt (0.08). The difference is even more dramatic when we compare the corresponding diversity and novelty scores. Dynamic prompt achieves a novelty of 0.93 and diversity of 0.96, an improvement of 9% (0.85) and 10% (0.87) respectively over the soft prompt, and an even more pronounced improvement of 26% (0.74) and 22% (0.79) respectively over fine-tuning the model.

Next, we observe that soft-prompting is competitive in the low data setting and outperforms fine-tuning. However, with more than 30% of the training data available, soft-prompting stops improving altogether and begins to under-perform the other models. Dynamic prompt on the other hand maintains the best BLEU4 score no matter the amount of training data.

Now we turn to the question of novelty and diversity degradation. We observe that the dynamic prompt does not suffer from this issue nearly as much as the other models. In contrast, for the soft-prompted and fine-tuned models, the better the BLEU4 score gets the lower the novelty drops. While soft prompt already mitigates this drop-off, degrading slower than DialoGPT, dynamic prompt

does not suffer from this effect and maintains above 90% novelty throughout. Similarly, for diversity, the dynamic prompt maintains around 0.95 score for any percentage of training data, while the soft prompt drops down over time and fine-tuning hovers around 0.80.

Finally, we can turn to the comparison of DialoGPT and GPT-2. Against our expectations, DialoGPT under-performs GPT-2 on BLEU4 despite the former having been pre-trained on dialog-specific data. The performance improvement is most notable in the case of the dynamic prompt, where GPT-2 (0.14) achieves 19% higher performance than DialoGPT (0.12). On the other hand, fine-tuning either model leads to nearly identical performance. You can find our discussion of this phenomenon in App. A, all our results in App. C and example outputs in App. D.

6 Related Work

Our work builds on two strains of thought. First, we deal with the problem of dull responses in dialog modelling. Related work includes the use of reinforcement learning (Li et al., 2016b), latent variables (Cao and Clark, 2017) and decoding techniques to mitigate this issue (Li et al., 2016a).

Second, we build on the idea of continuous prompting, which was developed concurrently by Lester et al. (2021) and Li and Liang (2021). There are many variations of the prompting paradigm. For instance, we fine-tune the embeddings along with the prompts, but Liu et al. (2021b) tune the prompt with the full model. Schick and Schütze (2021b) on the other hand tune only the model, while keeping the prompt fixed.

7 Conclusion

We find dynamic-prompting bests fine-tuning by generating more novel and diverse responses with a higher BLEU score while training only a small portion of the DialoGPT/GPT-2 parameters. Thus proving itself as a useful method for mitigating dialog modelling issue of *dull* responses. Prompting the general purpose GPT-2 achieves much higher performance than prompting the specialist DialoGPT model, suggesting that for pre-training data, diversity is more valuable for dialog modelling than dialog-specific information.

References

- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Kris Cao and Stephen Clark. 2017. [Latent variable dialogue models and their diversity](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 182–187, Valencia, Spain. Association for Computational Linguistics.
- Xiaodong Gu, Kang Min Yoo, and Sang-Woo Lee. 2021. [Response generation with context-aware prompt learning](#). *arXiv preprint arXiv:2111.02643*.
- Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. [The power of scale for parameter-efficient prompt tuning](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Jiwei Li, Will Monroe, and Dan Jurafsky. 2016a. [A simple, fast diverse decoding algorithm for neural generation](#). *CoRR*, abs/1611.08562.
- Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016b. [Deep reinforcement learning for dialogue generation](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas. Association for Computational Linguistics.
- Xiang Lisa Li and Percy Liang. 2021. [Prefix-tuning: Optimizing continuous prompts for generation](#). 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 4582–4597, Online. Association for Computational Linguistics.
- Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. [DailyDialog: A manually labelled multi-turn dialogue dataset](#). In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.
- Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021a. [Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing](#).
- Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. [GPT understands, too](#). *arXiv preprint arXiv:2111.02643*.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).
- Michael Rescorla. [The language of thought hypothesis](#). In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*.
- Timo Schick and Hinrich Schütze. 2021a. [Exploiting cloze-questions for few-shot text classification and natural language inference](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.
- Timo Schick and Hinrich Schütze. 2021b. [It’s not just size that matters: Small language models are also few-shot learners](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.
- Iulian V. Serban, Alessandro Sordani, 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*, AAAI’16, page 3776–3783. AAAI Press.
- Alessandro Sordani, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. [A neural network approach to context-sensitive generation of conversational responses](#). In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205, Denver, Colorado. Association for Computational Linguistics.
- Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill 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, Online. Association for Computational Linguistics.
- Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. [Learning discourse-level diversity for neural dialog models using conditional variational autoencoders](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), pages 654–664, Vancouver, Canada. Association for Computational Linguistics.

A Appendix A: Discussion

Liu et al. (2021a) designed their soft prompt based on the observation that context can control the LLM without the need to change its parameters. They demonstrate that the context might be hard to find among the existing word embeddings and instead learn it via back-propagation. We go a step further and demonstrate that when it comes to dialog generation it is harder to learn a single prompt than to learn a function that conditions the prompt on the query. We believe that this is because there is no single context, represented by words or embeddings, that can capture a complex task such as a conversation. This becomes intuitive if we separate the task of language generation from the task of making a conversation. The conversation is an interaction of the query with the state of the conversation agent. Without the ability to respond to the query the agent/language model is simply saying whatever is the most probable response. A prompt can induce some general modification over this response, but there is no process by which the agent can react to the query. In our dynamic prompt setting, the agent, represented by the controller module, is allowed to learn how to respond to each query.

The separation of thought from language production is of course not a new idea. Under the language of thought hypothesis, *mentalese* is the language of thought (Rescorla). While similar to natural language in its compositional structure, mentalese is separate from language itself. From this perspective, building models that separate the task of learning a language from that of learning how to use language makes perfect sense. Following this intuition, it also makes sense to pre-train the language model on general language, rather than a specific task. We believe that in our setting, the superior language ability of GPT-2 allows for a better Controller in the dynamic prompt setting.

B Appendix B: Experimental Details

We implement our models using the Pytorch and Huggingface libraries. We experiment with total of 36 configurations, testing three learning methods (Fine-tuning, Soft-prompting and Dynamic-prompting) on two LLMs (DialogPT and GPT-2) and six data regimes (10%, 20%, 30%, 50%, 70%,

and 100%). In each configuration, we run 12 trials of different learning rates $\in [3 \times 10^{-6}, 0.009]$, with a batch size of 8, for a maximum of 300 epochs (with early stopping after 100 epochs), and select the best model by validation BLEU. Each trial is done on a single GPU with 32GB memory, and the maximum training time is 14 days. To generate the model output, we always use only greedy decoding.

C Appendix C: Full Results

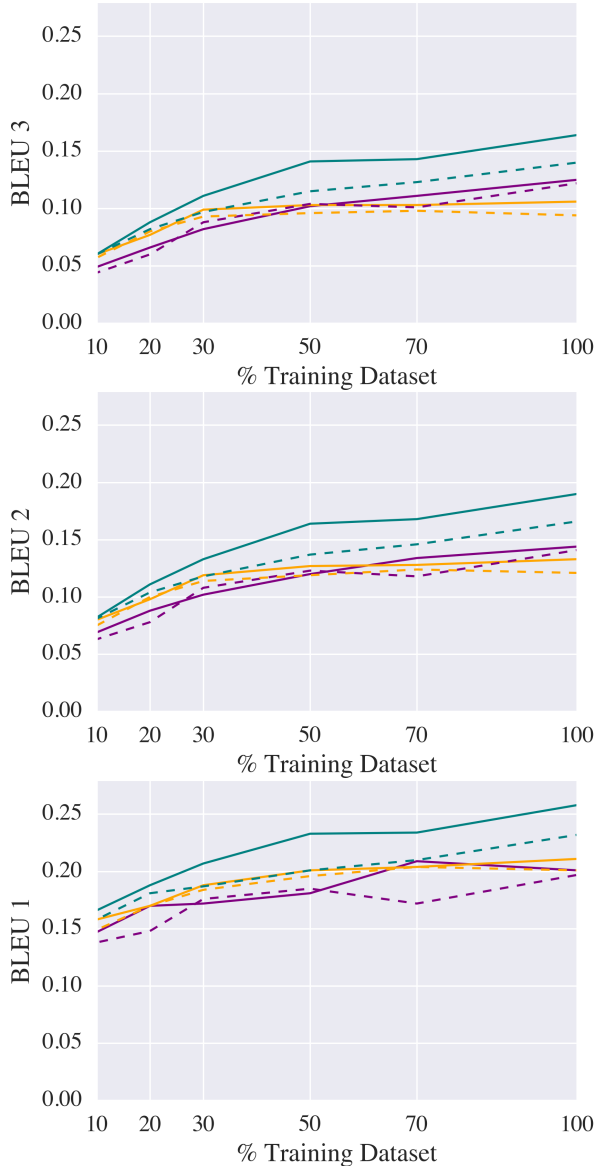


Figure 3: BLEU scores for **fine-tuning**, **soft-prompting** and **dynamic-prompting**. Full line is GPT-2, dashed line is DialogPT.

Data	Model	BLEU1	BLEU2	BLEU3	BLEU4	Novelty	Diversity
10%	fine-tuning	0.138	0.063	0.044	0.034	0.850	0.732
	soft-prompting	0.149	0.075	0.057	0.046	0.908	0.895
	dynamic-prompting	0.158	0.081	0.060	0.048	0.902	0.889
20%	fine-tuning	0.148	0.078	0.060	0.050	0.827	0.764
	soft-prompting	0.170	0.100	0.080	0.067	0.921	0.926
	dynamic-prompting	0.181	0.104	0.082	0.068	0.942	0.941
30%	fine-tuning	0.176	0.108	0.088	0.075	0.810	0.809
	soft-prompting	0.184	0.114	0.093	0.078	0.921	0.928
	dynamic-prompting	0.187	0.118	0.097	0.082	0.940	0.944
50%	fine-tuning	0.185	0.123	0.104	0.090	0.791	0.813
	soft-prompting	0.196	0.119	0.096	0.080	0.884	0.903
	dynamic-prompting	0.201	0.137	0.115	0.098	0.939	0.952
70%	fine-tuning	0.172	0.118	0.101	0.089	0.714	0.756
	soft-prompting	0.204	0.124	0.098	0.080	0.869	0.889
	dynamic-prompting	0.210	0.146	0.123	0.104	0.930	0.958
100%	fine-tuning	0.197	0.141	0.122	0.109	0.737	0.787
	soft-prompting	0.201	0.121	0.094	0.076	0.847	0.874
	dynamic-prompting	0.232	0.166	0.140	0.119	0.933	0.961

Table 1: The full results of our experiments on DailyDialogue with DialoGPT.

Data	Model	BLEU1	BLEU2	BLEU3	BLEU4	Novelty	Diversity
10%	fine-tuning	0.147	0.069	0.049	0.039	0.910	0.852
	soft-prompting	0.158	0.080	0.060	0.048	0.910	0.903
	dynamic-prompting	0.166	0.082	0.060	0.047	0.939	0.917
20%	fine-tuning	0.170	0.088	0.066	0.054	0.903	0.856
	soft-prompting	0.170	0.098	0.077	0.065	0.881	0.878
	dynamic-prompting	0.188	0.111	0.088	0.074	0.930	0.932
30%	fine-tuning	0.172	0.102	0.082	0.070	0.855	0.837
	soft-prompting	0.188	0.119	0.099	0.084	0.906	0.916
	dynamic-prompting	0.207	0.133	0.111	0.095	0.923	0.934
50%	fine-tuning	0.181	0.120	0.102	0.089	0.806	0.799
	soft-prompting	0.201	0.127	0.103	0.087	0.873	0.902
	dynamic-prompting	0.233	0.164	0.141	0.124	0.924	0.947
70%	fine-tuning	0.209	0.134	0.111	0.095	0.883	0.900
	soft-prompting	0.204	0.128	0.103	0.086	0.897	0.917
	dynamic-prompting	0.234	0.168	0.143	0.124	0.912	0.946
100%	fine-tuning	0.201	0.144	0.125	0.111	0.742	0.801
	soft-prompting	0.211	0.133	0.106	0.087	0.845	0.874
	dynamic-prompting	0.258	0.190	0.164	0.142	0.913	0.944

Table 2: The full results of our experiments on DailyDialogue with GPT-2.

D Appendix D: Examples

Query	Dynamic-prompting	Fine-tuning	Soft-prompting
I think it unwise for either of us to be inflexible. How about meeting each other halfway?	Do I know how to meet these employees?	Then, what's your countr-offer?	That's verywise for either of us. How about meeting another person in person?
It's a success all right. The kids keep trying to catch me.	You better attach some good ones too.	I'm getting more and more special.	I'm glad you're here.
You can audition music on line, and you can audition it off line after downloading it, so you needn't pay for it.	But when I am getting done I will make a big fan list.	Is that a lot of people's music here?	Actually, I have been thinking about taking the notes from the book.
What date would you like to fly? Saturday, July 25. How many people will be traveling?	You'd better get a lot of tickets first.	A few people are coming to get me.	Two flight attendants will be coming with you.
It's more violent than TV news! What time is it anyway?	I don't know, but you are still beating around the bush. You aren't going to be able to stop anybody who looks like you.	Oh, man! I didn't notice it.	It's after ten o'clock, after the candles, so it's hard for me to make a decision.

Table 3: Example responses generated by the models under consideration. Red indicates reused training data response.