CRL-Prompt: Contrastive and Reinforcement Learning for Soft Prompt Tuning of Language Models

Anonymous ACL submission

Abstract

Prompt choice is crucial in adapting language models to downstream tasks, particularly under low-resource conditions. Manual prompt engineering is time-consuming, non-scalable, and brittle, while current auto-prompting techniques are still far from maturity. This paper presents a two-stage method for prompt learning, CRL-Prompt, based on soft prompt initialization followed by contrastive and reinforcement-based refinement. Our method operates entirely over frozen models and is compatible with standard classification tasks. Experimental study demonstrates that our approach achieves consistent improvements in accuracy over baseline prompt tuning strategies, with gains of up to 2.2% while training fewer than 0.25% of model parameters.

1 Introduction

Language models (LMs) have become a cornerstone of modern natural language processing, achieving state-of-the-art results on tasks ranging from sentiment classification (Edwards and Camacho-Collados, 2024; Stigall et al., 2024) to open-ended generation (Maity et al., 2024). A key factor in their adaptability is using *prompts*, input sequences that condition the model's behavior and outputs. It is known that prompt choice can significantly influence performance in few-shot and zero-shot settings, where labeled data is scarce or unavailable (Brown et al., 2020). As a result, prompt engineering has emerged as a critical mechanism for controlling and adapting LMs in parameter-efficient ways (Chen et al., 2024; Marvin et al., 2023; Peng et al., 2024).

Despite this importance, most prompt engineering today remains manual and heuristic in practice (Sahoo et al., 2024). Indeed, effective prompts often require domain knowledge, iterative experimentation, and extensive trial-and-error. The known practices do not scale and yield fragile solutions that generalize poorly across tasks. This has led to growing interest in automated methods for prompt learning (Chang et al., 2024; Spiess et al., 2025; Xiao et al., 2025). Existing research has approached this problem from three major angles (Shin et al., 2020; Li et al., 2023; Zhuge et al., 2024). The first is *discrete prompt search*, where token-level prompt candidates are generated and scored using surrogate metrics, as seen in methods like TextGrad (Yuksekgonul et al., 2024) and Automatic Prompt Engineer (APE) (Zhou et al., 2022). Second, *continuous soft prompt tuning* directly learns embeddings, virtual prompt vectors, that are optimized via gradient descent on a frozen language model; notable examples include P-Tuning v1 (Liu et al., 2021a) and v2 (Liu et al., 2021b) and MixtureSoft (Qin and Eisner, 2021). Finally, *Reinforcement Learning (RL)–based refinement* methods, such as ConsPrompt (Weng et al., 2024), adapt prompts by optimizing for task-specific rewards. While these techniques have demonstrated promise, they also exhibit limitations: discrete methods can be computationally expensive and brittle; continuous tuning may overfit in low-resource regimes due to proxy losses; and RL often requires complex reward shaping and unstable training dynamics.

This work addresses these shortcomings by introducing a two-stage CRL-Prompt framework for automated soft prompt learning. In the first stage, we leverage P-Tuning v2 (Liu et al., 2021b) to initialize trainable key/value vectors injected into all transformer layers. In the second stage, we refine the prompts using a combination of contrastive regularization (Chen et al., 2020; Weng et al., 2024; Yu et al., 2020) and reinforcement feedback-based policy optimization guided by task-level accuracy (Li et al., 2023; Zhuge et al., 2024). The former term is designed to enhance the robustness of learned prompts by improving representation geometry. The latter term aligns optimization more closely with downstream task goals. Our method is fully

Algorithm 1 Proposed CRL-Prompt approach

```
Require: Frozen LM f_θ, initial prompt P_0, labeled data D, reward subset D_RL, number of steps T, reward-update interval N, coefficients β, γ, noise variance σ^2
Phase 1: Prompt tuning via cross-entropy
for each batch (x_i, y_i) from D do
    Compute L_CE(P, x_i, y_i)
    Update P using gradient of L_CE
end for
Phase 2: Mixed optimization loop
for t = 1 to T do
    for each batch (x_i, y_i) from D do
        Compute L_CE(P, x_i, y_i)
        Compute L_contrast(P, x_i, y_i)
        if t mod N = 0 then
            Sample perturbed prompt: P' ~ N(P, σ^2 I)
            Evaluate f_θ(P', x) on D_RL to obtain accuracy r
            L_RL ← −r log π(P'; θ)
        else
            L_RL ← 0
        end if
        L_total ← L_CE + β L_contrast + γ L_RL
        Update P using gradient of L_total
    end for
end for
return final prompt P
```

compatible with frozen LMs and operates without modifying their parameters. The experimental results demonstrate that our framework consistently outperforms state-of-the-art baselines (Shin et al., 2020; Yuksekgonul et al., 2024; Qin and Eisner, 2021), achieving gains of up to 2.2% in accuracy while training less than 0.25% of model parameters. The source code will be publicly released[1].

2 Proposed Approach

Let f_θ be a frozen LM with parameters θ, and let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ be a text labeled dataset for a classification task, i.e., $y_i \in \{1, ..., C\}$, where C is the number of classes. We aim to find prompt parameters P (e.g., soft embeddings or prefix vectors) that guide the LM toward accurate predictions, while keeping its weights θ fixed. In particular, we optimize P to maximize classification accuracy on a held-out validation set $\mathcal{D}_{\text{val}}$:

$$\max_{P} \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{(x,y)\in\mathcal{D}_{\text{val}}} \mathbf{1}\left[f_\theta(P,x) = y\right], \quad (1)$$

where $\mathbf{1}[\cdot]$ is the indicator function, and $f_\theta(P,x)$ is the predicted class label for input text x under prompt P.

To solve this problem, we propose a two-stage automated prompt optimization framework (Algorithm 1). The first phase applies an arbitrary technique, e.g., P-Tuning v2 (Liu et al., 2021b), to initialize trainable soft prompts. They consist of virtual key/value vectors injected at each transformer layer of a frozen LM. Only these prompt embeddings are updated during training, keeping the model weights unchanged. The training objective in this phase is standard cross-entropy loss on a small labeled dataset:

$$L_{\text{CE}}(P, x_i, y_i) = -\log\left[softmax\left(f_\theta(P, x_i)\right)_{y_i}\right], \quad (2)$$

where $f_\theta(P,x) \in \mathbb{R}^C$ are the logits produced by the frozen LM under prompt P for input x_i.

Once a stable initialization is learned, we move to the second phase and use contrastive regularization and RL-based optimization to enhance the soft prompts. While standard soft prompt tuning minimizes a cross-entropy loss over labeled examples, it does not explicitly encourage the prompt to separate semantically similar inputs from different classes. This can lead to overfitting or instability in few-shot settings, particularly when initialization is noisy or training data is limited. To address this, we introduce a contrastive loss term that improves the representational geometry of the learned prompt space. Given a batch of examples $\{(x_i, y_i)\}$, negative prompt variants are generated for each batch by applying dropout or noise to the current prompt parameters. The InfoNCE-style loss (Chen et al., 2020) is used to encourage embeddings of correct predictions to cluster together, while pushing apart those of incorrect ones:

$$\mathcal{L}_{\text{contrast}}(P, x_i, y_i) =$$

$$= -\log \frac{e^{\frac{\text{sim}(h_i, h_i^+)}{\tau}}}{e^{\frac{\text{sim}(h_i, h_i^+)}{\tau}} + \sum_{j=1}^{k} e^{\frac{\text{sim}(h_i, h_j^-)}{\tau}}}, \quad (3)$$

where h_i is the CLS embedding of input x_i with prompt P, h_i^+ is a positive example (original or clean version), h_j^- are negative examples obtained

[1]https://anonymous.4open.science/r/CRL-prompt-4FBA

via dropout or permutation, sim is a similarity measure, and τ is a temperature hyperparameter. This regularization makes the prompt more robust to small perturbations and improves generalization in low-data regimes.

Periodically (every N steps), we run the RL optimization. The soft prompt is periodically perturbed with Gaussian noise, and a sampled prompt $P' \sim \mathcal{N}(P, \sigma^2 I)$ is used to evaluate reward, downstream Accuracy of the model on a held-out subset $\mathcal{D}_{\mathrm{RL}}$:

$$r \;=\; \mathrm{Acc}\Big\{ \arg\max f_\theta(P', x) \;\Big|\; (x, y) \in \mathcal{D}_{\mathrm{RL}} \Big\}. \tag{4}$$

The RL gradient is used to fine-tune the prompt to maximize this reward:

$$L_{\mathrm{RL}}(P; P', \mathcal{D}_{\mathrm{RL}}) \;=\; -r\;\log\big[\pi(P'; \theta)\big], \tag{5}$$

where $\pi(P';\theta) \propto e^{-\frac{\|P'-P\|^2}{2\sigma^2}}$.

The final loss at the second step combines cross-entropy, contrastive, and policy-gradient terms with scalar weights:

$$L_{\mathrm{total}} \;= L_{\mathrm{CE}} \;+\; \beta\, L_{\mathrm{contrast}} \;+\; \gamma\, L_{\mathrm{RL}}\,. \tag{6}$$

Our Algorithm 1 draws on prior work in prompt tuning, contrastive learning, and reinforcement feedback to form a unified and efficient framework. Unlike frameworks that rely on prompt embeddings or classifier-based scoring, we work entirely within the training loop of a frozen LM. This makes our method efficient and deployable in real-world few-shot scenarios with limited data and compute. Unlike discrete methods, our CRL-Prompt operates over continuous prompts; unlike pure soft tuning, it optimizes prompts directly for the downstream task metric; and unlike full RL pipelines, it maintains low compute cost by combining contrastive loss with lightweight REINFORCE (Williams, 1992) updates. For example, compared to ConsPrompt (Weng et al., 2024), our framework does not require batch-level sampling or re-ranking and is compatible with standard PEFT libraries.

Speaking of limitations, our algorithm requires a validation split for reward estimation and introduces additional training costs due to periodic REINFORCE updates. It is also sensitive to hyperparameter tuning (e.g., reward weights, contrastive temperature). Currently, the framework is limited to classification tasks and assumes the availability of labeled data.

3 Experimental Results

To validate the proposed hybrid prompt engineering framework (Section 2), we conduct experiments on three standard text classification benchmarks for English language: 1) **AG News** (Zhang et al., 2015) – 4-way news topic classification, 120K training and 7.6K testing examples; 2) **TREC** (Li and Roth, 2002) – 6-way question type classification, 5.5K training and 500 testing examples; and 3) **SST–2** (Stanford Sentiment Treebank) (Socher et al., 2013) – binary sentiment classification, 67K training and 1.8K testing examples. In addition, we consider a more complicated **EmpatheticDialogues** dataset (Rashkin et al., 2019), which contains 25000 empathetic dialogues labeled by 32 emotion classes. For all datasets, we use conventional train/test splits provided by their authors. Appendix B contains additional details about baselines and hyperparameters.

Table 1 summarizes mean accuracy on test sets after 10 runs. We ran all experiments on a single Nvidia A100 GPU. The total training time for each method was restricted by 2 hours.

On the AG News dataset, our method reaches 94.0% accuracy with RoBERTa-base and 94.6% with Falcon-RW-1B. These results improve upon the best-performing baseline (Prompt v2) by 0.8 and 1.4 percentage points, respectively. The gains are significant given Prompt v2's strong performance.

On the TREC question classification task, our method achieves 96.0% with RoBERTa and 96.2% with Falcon, significantly outperforming Prompt v2 by 2.2% and 3.4%. This gap is the largest among the datasets, which we attribute to TREC's small size and fine-grained nature. In low-resource settings, the robustness introduced by contrastive regularization and alignment with end-task rewards proves especially beneficial.

For SST-2, a binary sentiment task, our approach yields 94.0% with RoBERTa and 94.3% with Falcon. Although absolute gains over Prompt v2 and MixtureSoft are slightly smaller (1.8–2.0%), they are consistent across architectures. The results also show that MixtureSoft, while effective for RoBERTa, does not generalize to larger models like Falcon, highlighting a strength of our approach.

For our most complex dataset, EmphateticDialogues, the proposed CRL-Prompt is again the most accurate technique with accuracy of more than 47% and 40% with RoBERTa and Falcon, re-

	AG News		TREC		SST-2		EmphateticDialogues	
Method	**RoBERTa-base**	**falcon-rw-1b**	**RoBERTa-base**	**falcon-rw-1b**	**RoBERTa-base**	**falcon-rw-1b**	**RoBERTa-base**	**falcon-rw-1b**
HandCraft	0.470±0.00	0.450±0.000	0.500±0.000	0.390±0.000	0.730±0.000	0.790±0.000	0.272±0.000	0.203±0.000
TextGrad	0.700 ±0.004	0.880±0.003	0.550±0.003	0.395±0.002	0.734 ±0.005	0.810±0.004	0.363±0.006	0.306±0.005
APE	0.550±0.003	0.520±0.003	0.546 ±0.003	0.410±0.004	0.845±0.002	0.827±0.002	0.372±0.005	0.318±0.004
MixtureSoft	0.888±0.004	–	0.656±0.005	–	0.922±0.001	–	–	–
ConsPrompt	0.839±0.005	–	0.691±0.003	–	0.892±0.002	–	–	–
Prompt v1	0.927±0.003	0.929±0.002	0.922±0.004	0.672±0.007	0.898±0.001	0.948±0.001	0.437±0.005	0.385±0.007
Prompt v2	0.932±0.001	0.932±0.001	0.938±0.002	0.928±0.001	0.922±0.002	0.903±0.003	0.454±0.004	0.391±0.006
Our CRL-Prompt	**0.940**±0.002	**0.946**±0.003	**0.960** ±0.002	**0.962**±0.001	**0.940** ±0.003	**0.943** ±0.001	**0.474**±0.007	**0.402**±0.005

Table 1: Main results: accuracy on the test set.

Method Variant	**AG News**	**TREC**	**SST-2**
P-Tuning v2 only ($\beta = 0, \gamma = 0$)	0.932	0.938	0.922
+ Contrastive only ($\beta > 0, \gamma = 0$)	0.936	0.953	0.931
+ RL only ($\beta = 0, \gamma > 0$)	0.935	0.948	0.927
Full: Contrastive + RL ($\beta > 0, \gamma > 0$)	**0.940**	**0.960**	**0.940**

Table 2: Ablation results: test accuracy for different loss variants (RoBERTa-base).

spectively. It is worth noting that the former metric is higher than the results of specialized techniques on this dataset: 36.57% of KEMP (Li et al., 2022) and 36.84% of CEM (Sabour et al., 2022). Moreover, we achieve state-of-the-art results compared with handcrafted prompts for GPT-4 (44.2%) and its application in the multi-agent InsideOut framework (Mozikov et al., 2024) (45.1%).

Thus, across all datasets, our approach proves effective and stable. It consistently outperforms discrete search (APE, TextGrad), pure soft tuning (Prompt v1/v2), and reinforcement-only methods (ConsPrompt).

Moreover, the proposed approach achieves high parameter efficiency by updating only the soft prompt parameters while keeping the backbone LM frozen throughout training. Following standard practice, we use 20 learnable virtual tokens per transformer layer. For instance, RoBERTa-base consists of 12 layers with a hidden size of 768, which results in $20 \times 12 \times 768 = 184{,}320$ trainable parameters. Given the full model size of approximately 125 million parameters, this accounts for less than 0.15% of the total parameters. Similarly, for Falcon-RW-1B, the number of updated parameters $32 \times 20 \times 4544 = 2{,}908{,}160$ remains approximately equal to 0.25% out of the total number (1 billion) of weights in the model. This lightweight design makes our method well-suited for settings with limited computational resources or constraints on model modification.

To better understand the contribution of each component in our hybrid framework, we perform ablation experiments by selectively disabling parts of the loss function. Table 2 summarizes the impact of each component for the RoBERTa model. These results confirm that combining gradient-based soft prompt tuning with contrastive and reinforcement refinements leads to consistent and significant accuracy improvements across tasks and model architectures. The two-stage design is practical: the initial P-Tuning phase provides a strong starting point for optimization, while contrastive regularization enhances generalization by improving the representational geometry of the learned prompts. Although applied infrequently, reinforcement-based updates align the optimization process with the true end-task objective (classification accuracy), yielding additional performance gains.

4 Conclusion

In this paper, we introduced a novel Algorithm 1 for soft prompt learning. Our approach operates in a parameter-efficient regime without modifying the backbone LM, making it suitable for resource-constrained few-shot scenarios. It is experimentally shown (Table 1) that the proposed strategy consistently outperformed discrete and soft prompt baselines, yielding accuracy gains of up to 2.2%. While robustness to input perturbations is a promising direction, it was not evaluated in this work and remains a subject of future research. Our ablation studies (Table 2) confirm these components' individual and joint benefits. Our method's simplicity, modularity, and effectiveness across datasets and models may make it a practical foundation for scalable LM adaptation in various settings.

Limitations

In this paper, we focus on text classification tasks to simplify the computation of a reward as a validation accuracy (4). In future research, we plan to extend the proposed framework beyond classification to include generative (Zhou et al., 2022) and multi-task settings, where prompt adaptation becomes even more challenging.

Second, due to focus on classification problems, we chose relatively small models (RoBERTa-base, Falcon-1B) that are widely used in practice for such tasks. Moreover, as shown in our experiment with the EmphateticDialogues dataset, our results with the RoBERTa model are even better than hand-crafted prompt design for GPT-4 (Mozikov et al., 2024). Nevertheless, in the case of text generation and similar tasks, it is worth considering more complicated LMs in the future.

Finally, the online computation of accuracy in RL loss (4) may be time-consuming if the validation set is large. One of the promising directions is using off-policy or bandit-based RL algorithms (Li et al., 2023) to reduce the overhead introduced by online reward computation.

Ethical Considerations

It is recognized that the development of language models is accompanied by inherent risks, which require a deliberate examination of the ethical implications. The experimental framework has incorporated pretrained models, such as RoBERTa-base and falcon-rw-1b, and public datasets, including AG News, TREC, SST-2, and EmphateticDialogues. Their respective publishers have carefully processed these models and datasets, addressing potential ethical concerns. Moreover, using text classification algorithms may have potential societal risks, none of which we feel must be specifically highlighted here. However, ethical risks from deployment should be carefully analyzed: overconfidence in CRL-Prompt's high accuracy could lead to unchecked outputs of text classification in high-stakes scenarios (e.g., healthcare).

References

Younes Belkada, Sylvain Gugger, Omar Sanseviero, and 1 others. 2023. Peft: Parameter-efficient fine-tuning. https://github.com/huggingface/peft. Hugging Face.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Kaiyan Chang, Songcheng Xu, Chenglong Wang, Yingfeng Luo, Xiaoqian Liu, Tong Xiao, and Jingbo Zhu. 2024. Efficient prompting methods for large language models: A survey. *arXiv preprint arXiv:2404.01077*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*.

Xiaojun Chen, Ting Liu, Philippe Fournier-Viger, Bowen Zhang, Guodong Long, and Qin Zhang. 2024. A fine-grained self-adapting prompt learning approach for few-shot learning with pre-trained language models. *Knowledge-Based Systems*, 299:111968.

Aleksandra Edwards and Jose Camacho-Collados. 2024. Language models for text classification: Is in-context learning enough? *arXiv preprint arXiv:2403.17661*.

Qintong Li, Piji Li, Zhaochun Ren, Pengjie Ren, and Zhumin Chen. 2022. Knowledge bridging for empathetic dialogue generation. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 36, pages 10993–11001.

Xiang Lisa Li, Ping Yu, Chunting Zhou, and Timo Shen. 2023. Dialogue for prompting: Policy–gradient–based discrete prompt generation (dp2o). *arXiv preprint arXiv:2308.07272*.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING)*.

Xiao Liu, Kaixuan Ji, Yicheng Fu, and Weng Lam. 2021a. P-tuning: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2103.10385*.

Xiao Liu, Kaixuan Ji, Yicheng Fu, and Weng Lam. 2021b. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*.

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. *arXiv preprint arXiv:1907.11692*.

Subhankar Maity, Aniket Deroy, and Sudeshna Sarkar. 2024. Investigating large language models for prompt-based open-ended question generation in the technical domain. *SN Computer Science*, 5(8):1–32.

Ggaliwango Marvin, Nakayiza Hellen, Daudi Jjingo, and Joyce Nakatumba-Nabende. 2023. Prompt engineering in large language models. In *International conference on data intelligence and cognitive informatics*, pages 387–402. Springer.

Mikhail Mozikov, Nikita Severin, Maria Glushanina, Mikhail Baklashkin, Andrey Savchenko, and Ilya Makarov. 2024. InsideOut: Unifying emotional llms to foster empathy. In *ECAI 2024*, pages 4499–4502. IOS Press.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Hamza Alobeidli, Alessandro Cappelli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data only. *Advances in Neural Information Processing Systems*, 36:79155–79172.

Cheng Peng, XI Yang, Kaleb E Smith, Zehao Yu, Aokun Chen, Jiang Bian, and Yonghui Wu. 2024. Model tuning or prompt tuning? a study of large language models for clinical concept and relation extraction. *Journal of Biomedical Informatics*, 153:104630.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. *arXiv preprint arXiv:2104.06599*.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

Sahand Sabour, Chujie Zheng, and Minlie Huang. 2022. CEM: Commonsense-aware empathetic response generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11229–11237.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of EMNLP*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment TreeBank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642.

Claudio Spiess, Mandana Vaziri, Louis Mandel, and Martin Hirzel. 2025. AutoPDL: Automatic prompt optimization for llm agents. *arXiv preprint arXiv:2504.04365*.

William Stigall, Md Abdullah Al Hafiz Khan, Dinesh Attota, Francis Nweke, and Yong Pei. 2024. Large language models performance comparison of emotion and sentiment classification. In *Proceedings of the 2024 ACM Southeast Conference*, pages 60–68.

Jinta Weng, Yifan Deng, Donghao Li, Hao You, Yue Hu, and Heyan Huang. 2024. Consprompt: Exploiting contrastive samples for few-shot prompt learning. *arXiv preprint arXiv:2211.04118*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256.

Zehao Xiao, Shilin Yan, Jack Hong, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiayi Shen, Qi Wang, and Cees GM Snoek. 2025. DynaPrompt: Dynamic test-time prompt tuning. *arXiv preprint arXiv:2501.16404*.

Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. 2020. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. *arXiv preprint arXiv:2010.07835*.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Zhi Huang, Carlos Guestrin, and James Zou. 2024. Textgrad: Automatic "differentiation" via text. *arXiv preprint arXiv:2406.07496*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.

Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jürgen Schmidhuber. 2024. Gptswarm: Language agents as optimizable graphs. *arXiv preprint arXiv:2402.16823*.

A Related Work

Discrete Prompt Search. Early work in prompt engineering explored discrete search over token templates to elicit knowledge from frozen LMs. AutoPrompt (Shin et al., 2020) uses gradient signals to select informative trigger tokens. APE (Zhou et al., 2022) employs a language model to generate and rank natural language instructions. While fully automatic, these methods are limited to template-level search and do not learn prompts to optimize the downstream metrics.

Soft Prompt Tuning. Continuous prompt tuning approaches replace discrete templates with learnable embeddings. P-Tuning (Liu et al., 2021a) and its extension P-Tuning v2 (Liu et al., 2021b) inject virtual key/value vectors into frozen transformer layers, achieving strong performance with fewer parameters. MixtureSoft (Qin and Eisner, 2021) learns multiple prompt variants and averages their outputs. However, such methods rely on cross-entropy as a proxy loss, which does not always align with end-task accuracy.

Reinforcement and Contrastive Methods. Recent work incorporates RL to refine prompt parameters using task-level rewards. DP2O (Li et al., 2023) and GPTSwarm (Zhuge et al., 2024) generate prompt policies or agent graphs via policy gradients. ConsPrompt (Weng et al., 2024) introduces contrastive loss to improve few-shot generalization. These approaches improve robustness but often involve complex architectures or require substantial reward engineering.

B Experimental Setup

B.1 Baselines

All experiments use the following LMs: RoBERTa-base (Liu et al., 2019) with 125M parameters and falcon-rw-1b (Penedo et al., 2023) with 1B parameters. We compare our method to both "soft" and "discrete" prompt-engineering baselines using accuracy as the primary evaluation metric: APE (Zhou et al., 2022), ConsPrompt (Weng et al., 2024), MixtureSoft (Qin and Eisner, 2021), Prompt v1 (Liu et al., 2021a), Prompt v2 (Liu et al., 2021b) and TextGrad (Yuksekgonul et al., 2024). ConsPrompt and MixtureSoft are only available for RoBERTa, because they use maskedLM mode unsupported by falcon.

In addition, we use the following *HandCraft* (manual) prompts, where `{text}` is replaced by the input example, and the model's top-scoring output token was mapped to the corresponding label:

- **AG News:** "`Read the following news: {text}. What is the category of news (World, Sport, Business, or Science)? Answer:`"
- **TREC:** "`Question: {text}. Class of question (Entity, Abbreviation, Description, Human, Location, Number):`"
- **SST–2:** "`Review: {text}. Sentiment of review (positive or negative):`"

B.2 Hyperparameters and Implementation Details

All methods were implemented using the HuggingFace Transformers and PEFT libraries (Belkada et al., 2023). For our approach, we extended the standard training pipeline with a custom Trainer to support the two-phase optimization scheme described in Algorithm 1. In the first phase, we initialize the soft prompt with 20 virtual tokens per layer, following standard practice in prompt tuning literature. We optimize this prompt using cross-entropy loss with a fixed learning rate of 1×10^{-3} and a batch size 32. This setup proved sufficient to yield stable convergence in the initial prompt embedding.

In the second phase, the model is refined via contrastive regularization and RL-style updates. We perform a small-scale grid search over learning rates $\{1 \times 10^{-3},\ 2 \times 10^{-3},\ 5 \times 10^{-4},\ 1 \times 10^{-4}\}$ to ensure stable performance during joint optimization. In all experiments, we use contrastive dropout at a rate of 10% and set the temperature parameter in the InfoNCE loss to $\tau = 0.1$.

To reduce computational overhead, we compute RL-based reward signals every $N = 100$ steps using a 10% hold-out subset of the training data. Although this introduces additional cost, it helps align prompt updates with the true downstream metric (classification accuracy). The loss components (6) are weighted as follows: $\beta = 0.3$ for contrastive loss, and $\gamma = 2 \times 10^{-5}$ for reinforcement loss. We found that performance was sensitive to the choice of γ: overly large values destabilized training, while too-small values rendered the reward signal ineffective. Overall, these settings represent a trade-off between stability, efficiency, and expressiveness, and they generalize well across datasets and model architectures in our experiments.

C Use of scientific artifacts and AI assistants

AG News dataset (`https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset`) was provided by the academic community for research purposes. TREC dataset (`https://www.kaggle.com/datasets/thedevastator/the-trec-question-classification-dataset-a-longi`) is available under CC0: Public Domain license. SST–2 (`https://www.kaggle.com/datasets/atulanandjha/stanford-sentiment-treebank-v2-sst2`) was also released under CC0: Public Domain license. Finally, EmpatheticDialogues (`https://www.kaggle.com/datasets/atharvjairath/empathetic-dialogues-facebook-ai`) was provided under a CC BY-NC-SA 4.0 license.

RoBERTa-base (https://huggingface.co/FacebookAI/roberta-base) is available under the MIT License, while falcon-rw-1b (https://huggingface.co/tiiuae/falcon-rw-1b) is distributed under the Apache License 2.0. We used all the artifacts as intended by their creators. No personal information or offensive content is contained in the considered datasets.

The original text of this paper was spell- and grammar-checked and slightly smoothed out using Grammarly.