On the Investigation of Evolutionary Multi-Objective Optimization for Discrete Prompt Search

Anonymous ACL submission

Abstract

Discrete prompt search (DPS) aims to automatically find high-performing prompts that yield top accuracy in interactions with a pretrained language model. In the context of few-shot learning, evaluations of candidate prompts can only be done via a limited number of labelled examples. The search is often formulated as an optimization problem where prediction accuracy, F1 score, or cross-entropy loss is used as the objective function. While resulting prompts achieve top performance, they are mostly unreadable and uninterpretable, i.e., unlike natural languages. In this paper, we formulate DPS as a true multi-objective optimization (MOO) problem considering simultaneously both prompt performance and readability as separate objectives. We show that there exist certain degrees of conflict between the objectives, making the search for human-readable and highly-accurate prompts a challenging problem. We then propose the Multi-objective Evolutionary Algorithm for Predictive Probability guided Prompting (MoEAP3) to address the problem. Our MoEAP3 returns not a single final prompt as in conventional methods but a whole front of multiple candidate prompts, each representing an efficient trade-off between the objectives. Decision makers can straightforwardly investigate this front and intuitively select the prompt that yields the desired trade-off. Experimental results exhibit the superiority of MoEAP3 over state-of-the-art baselines.

1 Introduction

Pretrained language models (PLMs) can be fine-tuned with sufficiently-large training datasets to properly address many downstream tasks in natural language processing (Liu et al., 2019; Radford et al.; Raffel et al., 2020). In the scenarios of few-shot or zero-shot data, PLMs also yield competitive results via prompt-based learning (Gao et al., 2021), prompt tuning (Lester et al., 2021) and in-context learning (Brown et al., 2020). When the gradients of PLMs are accessible, prompts can be efficiently optimized with gradient descent algorithms because much fewer parameters need to be tuned in prompt tuning compared to conventional PLM fine-tuning (Li and Liang, 2021; Lester et al., 2021; Liu et al., 2023).

In practice, parameters and gradients of PLMs are not always available for prompting tuning via first-order gradient-based methods, e.g., GPT-3 (Brown et al., 2020) can only be accessed via OpenAI API at the current time. Some alternatives using evolutionary algorithms (EAs) are proposed for prompt tuning when PLM gradients are not available, e.g., black-box tuning (BBT) (Sun et al., 2022) employs the covariance matrix adaptation evolution strategy (CMA-ES) (Hansen et al., 2003) to optimize continuous prompts that can be appended to input texts for querying a PLM. However, such soft-prompt tuning approaches still require that the output of the PLM's embedding layer is accessible. In the true black-box scenario, **discrete prompt search (DPS)** is the feasible approach.

DPS methods can be divided into two groups: **(1) editing-based** (Zhang et al., 2023; Xu et al., 2022; Prasad et al., 2023): the search algorithm replaces, adds, or deletes some words of the manual prompts or instructions to boost their performance in a specific task. However, these methods require human effort to design *a priori* proper initial prompts (Wang et al., 2022b; Mishra et al., 2022), making their applications to specific tasks in low-resource languages difficult. **(2) sampling-based** (Deng et al., 2022; Zhou et al., 2023; Zhao et al., 2023; Shi et al., 2023; Diao et al., 2023): the search algorithm iteratively samples prompt tokens based on the vocabulary of the PLM, evaluates candidate prompts with few-shot data of a specific task, and then adapt these prompts with newly sampled tokens. Sampling-based DPS methods require much less human effort in initialization. Genetic Algorithm for Predictive Probability guided

Prompting (GAP3) (Zhao et al., 2023) is an exemplary sampling-based method, where the optimized prompts yield competitive results with full-model fine-tuning in certain tasks. Despite their effectiveness, these searched prompts are often unintelligible, incoherent, and mostly gibberish in terms of natural languages. Such a phenomenon is due to their high degree of freedom during the search process and the fact that prompt performance is the sole optimization objective.

In this paper, we re-formulate DPS as a bi-objective problem where both prompt performance and human-readability are equally handled as separate optimization objectives. We replace the single-objective genetic algorithm (GA) in GAP3 with a widely-used multi-objective evolutionary algorithm (MOEA), namely non-dominated sorting genetic algorithm II (NSGA-II) (Deb et al., 2002). Following GAP3, we name our method, Multi-objective Evolutionary Algorithm for Predictive Probability guided Prompting (MoEAP3). Employing MoEAP3 for solving the multi-objective DPS problems exhibits interesting findings as follows:

- There might exist certain trade-offs between prompt performance and intelligibility such that searching for highly-accurate and readable prompts is non-trivial, at least regarding current prompt representations. This conflicting nature of DPS requires the two objectives are kept separately because aggregating them with improper weights would result in poor generalization performance.

- Solving DPS as a multi-objective optimization problems returns a solution set of multiple prompts, where each candidate corresponds to a compromise between the objectives. Practitioners can investigate the solution set and simply select the prompt exhibiting the desirable trade-off *a posteriori* instead of having to determine some fixed weights *a priori*.

- There are prompts in the returned solution set of MoEAP3 that locate in interesting positions so-called "knee" solutions, which should be considered by the decision makers.

2 Related works

2.1 Readability of optimized prompts

The perplexity (Meister and Cotterell, 2021) is one popular metric for evaluating language models. The larger the perplexity value of a text, the less likely it can be observed given the probability distribution for perplexity computation. Therefore, the perplexity of a prompt can be estimated via a casual language model, e.g., GPT2 (Radford et al.). However, the perplexity metric exhibits certain disadvantages in scoring prompt readability for EA-based search methods: **(1)** because perplexity is sensitive to prompt lengths (Wang et al., 2022a), it may be inaccurate for prompt evaluations at early generations when candidate prompts have very few tokens (empty strings in the first generation); **(2)** because prompts containing repeated phrases have low perplexity (Wang et al., 2022a), trivial prompts may survive through many generations. In this paper, we consider an alternative to the perplexity in evaluating prompt readability, i.e., the **Fluency** metric (Krishna et al., 2020), which can be defined as the predictive probability of a classifier trained on a linguistic acceptance dataset. We here fine-tune the classifier DeBERTa-V3 (He et al., 2022) on the CoLA dataset (Warstadt et al., 2019) containing sentences labelled with "grammatical" or "ungrammatical" from linguistic literature. For efficient inference, we apply `load_in_4bits` quantization (Dettmers et al., 2022) to evaluate Fluency.

2.2 Multi-objective optimization (MOO)

MOO is an optimization methodology in which multiple objectives (or criteria) are taken into account. MOO has been utilized in many NLP research works, e.g., addressing multiple tasks in text classification (Chai et al., 2023), optimizing recall and precision in unknown intent detection (Prem et al., 2021), or optimizing coherence, accuracy, and regularization for word sense disambiguation and entity linking (Weissenborn et al., 2015), etc. For the discrete prompt search task, aiming to find human-readable prompts, FLUENTPROMPT (Shi et al., 2023) optimizes two objectives task labelling loss and prompt fluency loss by considering their weighted sum as a single optimization objective. The problem formulation FLUENTPROMPT can both be addressed with conventional single-objective optimization algorithms.

In this paper, we focus on the cases of *true* multi-objective formulations, where all the involving objectives are kept separately and are optimized simultaneously in an equal manner. If these objectives are competing with each other, there exists no feasible solutions that ideally optimize all objectives at the same time. Instead, true MOO aims to find the Pareto set of multiple candidate solutions, which can all be considered optimal in the sense that each of them represents an optimal trade-off regarding the objectives. The weighted-sum approach as in (Shi et al., 2023) can also be used, but the optimization must be run multiple times with different weight settings because each set of weights only yields a single trade-off solution. Due to the evolution mechanism for a population of multiple individuals, multi-objective evolutionary algorithms (MOEAs) are intrinsically well-suited to approximate such a Pareto solution set in a single optimization run. We here employ the widely-used MOEA Non-dominated Sorting Genetic Algorithm (NSGA-II) (Deb et al., 2002) to construct our MoEAP3 approach for discrete prompt search as solving an MOO problem.

3 Background: GA for Predictive Probability guided Prompting (GAP3)

Notation: Let $(x, y) \in D$ denote a sample (`input`, `label`) in a dataset D. A prompt $\mathcal{P}$ is merged with a sample (x, y) following the prompt template $\mathcal{T}$ (see Table 4 in Appendix) to make the final input $\mathcal{T}(x, y)$. We use a specific prompt template for each task. For example, the template [$\mathcal{P}$1] [X] [$\mathcal{P}$2] [Y] for SST-2 means that the input sentence x is placed between two prompts [$\mathcal{P}$1] and [$\mathcal{P}$2], and the label y is put at the end..

Population: GAP3 (Zhao et al., 2023) maintains a population $\mathcal{P}$ consists of N individuals $\mathcal{P} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_N\}$, where each individual $\mathcal{I}_i$ contains a prompt [$\mathcal{P}$1] or a pair of prompts [$\mathcal{P}$1] and [$\mathcal{P}$2]. The prompts in all individuals are initialized as empty strings.

Fitness score: The objective function value is often used to score the fitness of individuals in the evolving population. Even though we aim to find top-performing prompts, accuracy or F1 score should not be (solely) employed as the fitness function because there exist many prompts having the same values for these metrics, making it difficult to select truly better candidates in the evolution process. BBT (Sun et al., 2022) thus employs cross entropy or hinge loss as the minimization function instead of the negative accuracy. GAP3 (Zhao et al., 2023) uses accuracy or F1 score as the main fitness function to be optimized with a genetic algorithm, and for tie breaking in selections, a secondary fitness score is computed as follows:

$$F(\mathcal{T}) = \frac{1}{|D|} \sum_{(x,y)\in D} \delta_{y,\hat{y}} P(x, y),$$
$$P(x, y) = \frac{P(y^m = y|\mathcal{T}(x, y^m))}{\sum_{y'\in\mathcal{Y}} P(y^m = y'|\mathcal{T}(x, y^m))}, \quad (1)$$
$$\hat{y} = \arg\max_{y'\in\mathcal{Y}} P(y^m = y'|\mathcal{T}(\mathrm{x}, y^m)),$$

where $P(\cdot|\cdot)$ denotes the conditional probability given by the language model, $\mathcal{Y}$ is the set of task labels, and $\delta_{y,\hat{y}}$ is the Kronecker delta function for true label y and predicted label $\hat{y}$ so that only correctly predicted examples are taken into account. In our multi-objective formulation, we do not directly optimize for accuracy nor F1 score, but we use the above re-normalized predictive probability distribution in Equation 1 as one objective function (the other is the Fluency metric). We also use the two following variation operators of GAP3 (Zhao et al., 2023) to generate new candidate prompts.

Crossover: Crossover is performed over random pairs of individuals to create offspring (i.e., new individuals). During crossover, each [$\mathcal{P}_i$] of the two individuals are swapped with probability ρ_c.

Mutation: Each individual can also be mutated with probability ρ_m to create a new candidate. GAP3 (Zhao et al., 2023) implements two types of mutation: *insert* and *replace*. The *insert* mutation adds a `<mask>` token into a random position in a prompt. The *replace* mutation changes a random existing token into a `<mask>` token. This `<mask>` token is then filled with a token sampled based on the probability distribution of the PLM. Let t^m be a masked token in a prompt, $\mathcal{V}$ is a vocabulary of the PLM, t^m is determined by the following equation:

$$t^m = \arg\max_{t\in\mathcal{V}} \sum_{x,y\in D} \log P(y^m = y|\mathcal{T}^m_{i\leftarrow t}(x, y^m)) \quad (2)$$

where y^m is the label y masked, and $\mathcal{T}^m_{i\leftarrow t}$ is prompt template whose <mask> token is replaced with token t at the i index in mutation.

4 Proposed approach

4.1 Multi-objective formulation

In this paper, we formulate discrete prompt search as a bi-objective optimization problem:

$$\mathcal{T}^* \leftarrow \arg\max_{\mathcal{T}\,\in\,\Omega} f(\mathcal{T}) = (f_1(\mathcal{I}), f_2(\mathcal{I})), \quad (3)$$

where $\mathcal{T}$ denote a candidate prompt in a prompt search space Ω, the first objective $f_1(\mathcal{T}) = F(\mathcal{T})$

is the fitness score in Equation 1, the second objective $f_2(\mathcal{T})$ is a readability measure. We adapt the Fluency metric in (Krishna et al., 2020) to score the readability of candidate prompts (see Section 2.1).

A candidate prompt $\mathcal{T}_a$ Pareto **dominates** another prompt $\mathcal{T}_b$ (denoted as $\mathcal{I}_a \succ \mathcal{I}_b$) if $\mathcal{T}_a$ is not worse than $\mathcal{T}_b$ in both objectives and $\mathcal{T}_a$ is strictly better than $\mathcal{T}_a$ in at least one objective, i.e., $(\forall i \in \{1, 2\} : f_i(\mathcal{I}_a) \geq f_i(\mathcal{I}_b)) \wedge (\exists i \in \{1, 2\} : f_i(\mathcal{I}_a) > f_i(\mathcal{I}_b))$. If objectives f_1 and f_2 conflict with each other, there exists no utopian solution $\mathcal{T}^*$ that maximizes both objective simultaneously. Instead, the optimum of this bi-objective problem is the **Pareto set** P_S of prompts that are not dominated by any other prompts in the search space:

$$P_S = \{\mathcal{I}_a \mid \neg\exists\, \mathcal{I}_b \in \Omega : \mathcal{I}_b \succ \mathcal{I}_a\} \tag{4}$$

The objective value vectors of these Pareto-optimal prompts in P_S yield the so-called **Pareto front** P_F in the bi-objective space:

$$P_F = \{f(\mathcal{I}_i) = (f_1(\mathcal{I}_i), f_2(\mathcal{I}_i)) \mid \mathcal{I}_i \in P_S\} \tag{5}$$

It is costly and unnecessary to obtain the entire P_S. Instead, it suffices to achieve an **approximation set** $\mathcal{A}$ of non-dominated prompts forming a corresponding **approximation front** $f(\mathcal{A})$ that well approximates the Pareto front P_F.

4.2 Multi-objective evolutionary optimization

MoEAP3 utilizes NSGA-II (Deb et al., 2002) as the search method to obtain a good approximation set $\mathcal{A}$ of discrete prompts. The implementation of MoEAP3 is illustrated in Figure 1. Similar to GAP3, MoEAP3 evolves a population $\mathcal{P}$ of N candidate prompts (following a certain template for each task) in a generational manner. The population can be initialized with empty prompts, where tokens can be filled in via the mutation operator.

In each generation, candidate prompts are evaluated for their objective values (i.e., fitness score f_1 and fluency score f_2). A binary tournament selection procedure is carried out to select better prompts into a selection set $\mathcal{S}$: each time, two prompts are randomly sampled from the $\mathcal{P}$, and the better one is selected into $\mathcal{S}$. Selected prompts are then used as parent individuals to create offspring individuals $\mathcal{O}$ (i.e., new candidate prompts) via crossover and mutation. $\mathcal{P}$ and $\mathcal{O}$ are merged into a selection pool $\mathcal{P} \cup \mathcal{O}$, from which a so-called *non-dominated sorting* procedure takes place to partition both old and new individuals into non-domination ranks. Rank 1 ($F1$) contains prompts that are not dominated by any other prompts in $\mathcal{P} \cup \mathcal{O}$. Prompts in rank i (Fi), where $i > 1$, are also non-dominated if prompts from lower ranks are disregarded (see Figure 4). Afterward, N candidates with the lowest ranks from $\mathcal{P} \cup \mathcal{O}$ are selected to be the new population in the next generation. During any selection, a prompt in rank i is considered better than a prompt in rank j if $i < j$. When candidates from the same ranks need to compete with each other, the *crowding distance* metric, that measures the distance between a candidate and its two nearest neighbors in the objective space, is used to favor prompts that lie far away from the others. The algorithm terminates when certain criteria are met (e.g., the computing budget is over or the maximum number of generations is reached). The set of non-dominated prompts in the final population are considered the approximation set returned by MoEAP3.

5 Experiments

5.1 Datasets

We conduct experiments on both single-sentence and sentence-pair classification tasks, covering (1) **Sentiment analysis**: SST-2 (Socher et al., 2013), MR (Pang and Lee, 2005); (2) **Topic classification**: AG's News (Zhang et al., 2015); (3) **Natual language inference**: SNLI (Bowman et al., 2015), RTE (Wang et al., 2019); (4) **Paraphrase**: MRPC. (Dolan and Brockett, 2005). Dataset statistics and label words are listed in Table 4 in Appendix.

5.2 Baselines

We experiment with several existing black-box methods for prompt search as follows. **(1) Manual prompt**: human-designed prompts to formulate classification problems into fill-in-the-blank problems as in (Sun et al., 2022). **(2) Instruction**: handcrafted task description prompts for generalizing PLMs to a variety of unseen tasks (Wang et al., 2022b). **(3) BBT** uses an EA namely CMA-ES to optimize soft prompts for a frozen PLM (Sun et al., 2022). **(4) In context learning (ICL)** uses many samples concatenated for model input. ICL achieves remarkable performance in many tasks (Brown et al., 2020). **(5) BDPL** employs a policy gradient algorithm for gradient estimation to update the categorical distribution that is used to sample prompt tokens (Diao et al., 2023). **(6) ClaPS**

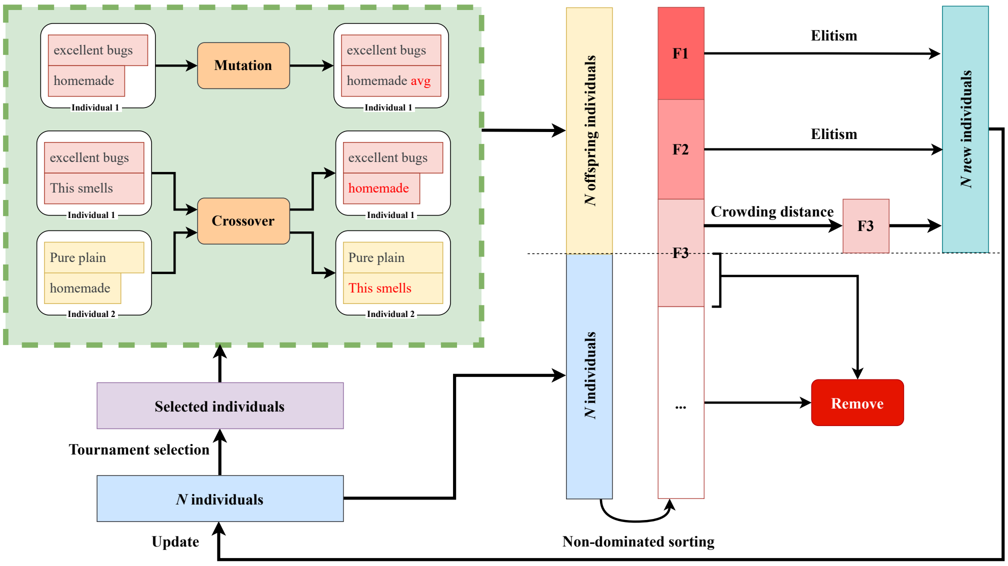

Figure 1: Illustration of of MoEAP3 for discrete prompt search.

clusters and prunes the search space (i.e., vocabulary), and then efficiently finds proper prompts via an EA (Zhou et al., 2023). **(7) GAP3**: our main EA-based baseline in this work. **(8) GAP3-2**: we experiment with a GAP3 variant that optimizes the sum of the fitness score and the fluency score. GAP3-2 is thus similar to the weighted sum approach of FLUENTPROMPT (Shi et al., 2023), but we here weight both kinds of scores equally. Other details of these baselines are in Appendix A.

5.3 Implementation details

Settings: We use `roberta-large` (Liu et al., 2019) and `flan-t5-base` (Chung et al., 2022) as the PLMs. We use GAP3 source code and the NSGA-II implementation of `pymoo` (Blank and Deb, 2020) to create MoEAP3. We set population size $N = 64$, and the number of generations $M = 50$, crossover and mutation probabilities $\rho_c = 0.5$ and $\rho_m = 0.75$, respectively. We use k-shot with $k = 16$ for all experiments, meaning that the training set contains k random samples for each label. The test sets for AG's news, MRPC, MR, and SNLI, are the original test datasets; we use the original development datasets for testing in the cases of SST-2 and RTE. All experiments are conducted on two Tesla T4 GPUs of Kaggle with three different random seeds following the practice in (Sun et al., 2022; Zhao et al., 2023; Diao et al., 2023). The number of samples in training and development sets are equal, following the true few-shot learning methodology as in (Perez et al., 2021). An ablation study regarding backbone PLMs and the k-shot value is provided in Appendix B.

5.4 Few-shot learning results

With the PLM `roberta-large`, Table 1 shows that MoEAP3 demonstrates superior performance across six datasets except for MRPC. MoEAP3 outperforms GAP3 by approximately 1.11%, 0.53%, 1.31%, 1.20%, and 0.24% on SST-2, AG's News, MR, SNLI, and RTE, respectively. However, the F1 score of MoEAP3 is lower than both GAP3 and GAP3-2 in on MRPC. While manual and instruction prompts have mediocre performance for other datasets, they yield the best results for MRPC.

With the PLM `flan-t5-base`, Table 2 shows that instruction prompts and BBT are competitive baselines for black-box prompt tuning. Because Flan-T5 models are fine-tuned by multi-task instruction datasets, instruction prompts tend to achieve high performance in MR and MRPC tasks. MoEAP3 still outperforms all baselines on SST-2, AG's News, and SNLI without using any manual initialization nor human-designed instructions. On average, MoEAP3 performs better with `flant-t5-base` than with `roberta-large`. This demonstrates the effectiveness of MoEAP3 with instruction-tuning models.

6 Pareto front analyses

6.1 Prompt readability analysis

Table 3 lists some example prompts with high fluency scores obtained by MoEAP3 on SST-2 and SNLI datasets. For SST-2, the word "Really" occurs in many prompts, showing that the generated

Methods	SST-2 (Acc)	AG's News (Acc)	MR (Acc)	SNLI (Acc)	RTE (Acc)	MRPC (F1)	Average
Manual	79.70	76.96	72.51	31.09	51.62	78.73	65.10
Instruction	76.95	56.61	75.80	37.43	53.79	**80.24**	63.47
ICL	$84.77_{1.48}$	$57.35_{3.51}$	$80.49_{1.92}$	$47.88_{1.60}$	$57.28_{2.73}$	$46.19_{3.74}$	62.33
BBT	$88.00_{0.98}$	$82.84_{0.80}$	$85.74_{1.18}$	$40.23_{3.85}$	$49.12_{4.10}$	$64.00_{10.79}$	68.32
BDPL	$86.68_{1.97}$	$70.26_{1.38}$	$82.80_{3.05}$	$31.82_{0.61}$	$53.29_{1.74}$	$54.67_{9.20}$	63.25
ClaPS-Ge	$83.87_{1.14}$	$84.16_{0.80}$	$82.85_{0.68}$	$41.23_{2.19}$	$51.34_{3.10}$	$55.48_{7.43}$	66.49
ClaPS-P	$85.00_{2.76}$	$83.57_{1.26}$	$84.75_{1.14}$	$40.57_{1.42}$	$49.53_{2.67}$	$50.86_{8.04}$	65.71
ClaPS-Gr	88.30	79.45	82.83	41.90	49.46	63.30	67.54
GAP3-2	$85.61_{3.36}$	$71.65_{3.87}$	$82.65_{4.46}$	$40.37_{1.35}$	$52.63_{3.00}$	$72.10_{3.00}$	67.50
GAP3	$89.30_{1.32}$	$83.82_{1.17}$	$86.93_{0.36}$	$49.88_{1.38}$	$58.60_{2.22}$	$69.80_{3.96}$	73.06
MoEAP3	$\mathbf{90.41_{0.86}}$	$\mathbf{84.35_{1.27}}$	$\mathbf{88.24_{1.34}}$	$\mathbf{51.08_{1.61}}$	$\mathbf{58.84_{3.44}}$	$68.17_{4.91}$	**73.52**

Table 1: Experimental results with `roberta-large` as the backbone PLM. We report the mean and standard deviation of each method over three random seeds. The best results are highlighted in **bold** for each task.

Methods	SST-2 (Acc)	AG's News (Acc)	MR (Acc)	SNLI (Acc)	RTE (Acc)	MRPC (F1)	Average
Manual	83.03	64.86	78.52	56.99	64.98	41.64	65.00
Instruction	90.25	62.11	**87.71**	63.03	75.45	**82.35**	76.82
BBT	$89.07_{0.43}$	$81.77_{1.47}$	$84.99_{0.97}$	$74.38_{2.74}$	$71.36_{0.91}$	$71.99_{3.70}$	78.93
ClaPS-Ge	$87.04_{0.23}$	$75.95_{1.26}$	$86.08_{0.46}$	$64.77_{2.36}$	$\mathbf{81.11_{1.10}}$	$64.19_{3.74}$	76.53
ClaPS-P	$87.84_{2.48}$	$76.56_{1.57}$	$83.96_{2.07}$	$67.07_{4.12}$	$79.18_{1.99}$	$64.73_{3.40}$	76.56
ClaPS-Gr	90.37	77.18	82.83	63.47	79.42	64.93	76.37
GAP3-2	$82.76_{5.70}$	$70.86_{10.25}$	$81.68_{2.00}$	$63.58_{6.04}$	$63.54_{6.57}$	$68.01_{3.41}$	71.72
GAP3	$85.93_{7.10}$	$84.36_{1.05}$	$82.37_{3.11}$	$69.93_{1.76}$	$63.54_{1.08}$	$70.58_{10.79}$	76.12
MoEAP3	$\mathbf{91.55_{0.59}}$	$\mathbf{84.76_{0.90}}$	$84.99_{1.65}$	$\mathbf{74.62_{1.71}}$	$74.12_{1.37}$	$75.30_{4.34}$	**80.89**

Table 2: Experimental results with `flant-t5-base` as the backbone PLM. We report the mean and standard deviation of each method over three random seeds. The best results are highlighted in **bold** for each task.

tokens are affected by the downstream task (e.g., SST-2), the label words (e.g., "good" and "bad" for SST-2), and the pretraining data of PLMs (e.g., RoBERTa). For example, if we let `roberta-large` fill the mask token in the input "Really <mask>", the mask token is then filled with the token "good". Some tokens exhibit a relationship with the downstream tasks, e.g., "reviewers", "Yep", and "impressions" are relevant words to the domain of movie review. However, for SNLI, the generated prompts seem to be less natural than those for SST-2 while still yielding high fluency scores. While the current Fluency metric can measure comprehensibility to certain extents, further research should be conducted to develop better readability models.

Prompt	**Fluency**	**Acc**
643 reviewers [X] Really [Y]	95.17	80.00
Yep [X] Really [Y]	95.07	78.21
NB [X] This smells unbelievably [Y]	95.15	86.35
Reviewer impressions [X] Really [Y]	94.08	84.44
Prompt	**Fluency**	**Acc**
Yep [X1] [Y] Chrys [X2]	93.50	41.17
Alright [X1] [Y] Hmm [X2]	95.71	38.07
Okay [X1] [Y] !!!! Success [X2]	92.84	45.67
Ibid [X1] [Y] Hmm [X2]	95.90	33.86

Table 3: Example resulting prompts with Fluency scores higher than 90.00 on SST-2 dataset (top) and SNLI (bottom). [X1], [X2], and [X] denote the input sentences, and [Y] denotes the corresponding label.

6.2 Prompt performance versus readability

Figure 2 exhibits the relationship among the performance of prompts, their lengths and readability. In general, shorter prompts tend to have higher fluency scores, and vice-versa. On the other hand, long prompts tend to be more consistent in yielding high performance. Finding long, readable, and highly-accurate prompts are challenging since there seem to exist certain degrees of trade-off among these objectives. Aiming to optimizing solely one objective likely compromises the other one. Aggregating all objectives together into an optimization function as in FLUENTPROMPT (Shi et al., 2023) results in a single trade-off solution for each set of weights. However, there exist no intuitive methods to properly determine weight values that yield the desirable trade-offs. By addressing discrete prompt search as a true MOO problem via approximating the Pareto front, our MoEAP3 is able to obtain a whole approximation set of diverse prompts with the same costs of one GAP3 run.

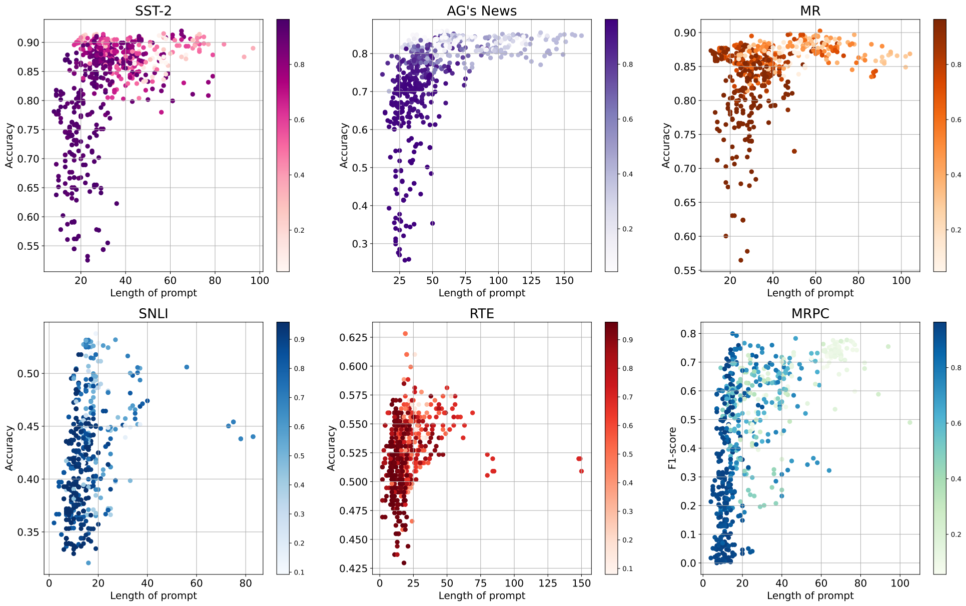

Figure 2: The relationship among the resulting prompts' performance, their lengths and readability in six datasets. Darker-colored dots are prompts with higher fluency scores. We plot the resulting prompts in the final populations of 10 independent MoEAP3 runs.

Figure 3 shows the merged approximation fronts formed by the final populations of 10 runs of MoEAPs and GAP3 (all dominated prompts are omitted). Solely optimizing for performance, resulting prompts of GAP3 score highly in terms of accuracy but poorly in terms of readability. Especially in MR and MRPC tasks, the resulting prompts of MoEAP3 entirely dominate those of GAP3. While GAP3 and other (single-objective) baseline search methods return a single resulting prompt each time, MoEAP3 returns an approximation set containing multiple diverse prompts from which users can choose their desired trade-off.

6.3 MoEAP3 and GAP3 prompt comparisons

We compare some exemplary prompts obtained by MoEAP3 and GAP3 with top performance on test datasets (i.e., accuracy or F1 score) of single-sentence and sentence-pair tasks in Figures 6 and 7, respectively. We also use `bloom-580m` (Workshop et al., 2022) to compute the perplexity (Log-PPL) (Meister and Cotterell, 2021). Bloom is a multilingual model that can tokenize non-Latin words (e.g., Korean, Japanese, Chinese) and is thus suitable for perplexity computation. Overall, top-performing prompts found by GAP3 is longer than those of MoEAP3. For example, in RTE, the prompt ['Yeah ��' , 'ILY Rather!!'] of MoEAP3 achieves an F1 score of 62.85 while the prompts with similar scores found by GAP3 are much longer, more complex, and less readable. The prompts of MoEAP3 contain much fewer tokens, which would incur less operation costs when using these prompts for querying language models. During the single-objective optimization process of GAP3, because performance is the sole fitness function, truncation selection would bias the search toward high-performing prompts, which typically contain many tokens and have low fluency scores. Short prompts are thus unlikely to survive for many generations due to the selection pressure. The true multi-objective optimization process of MoEAP3, on the other hand, retains these short prompts in its evolving population despite their low performance due to their high fluency scores. Population diversity is thus maintained, thereby fostering both kinds of prompts with high performance or high fluency. Short prompts that survive the Pareto dominance-based selection have better chances to improve their performance via crossover and mutation operators in subsequent generations.

6.4 Choosing prompts at knee positions

The return of an MoEAP3 run is an approximation set of many candidate prompts, from which decision makers need to select one (or several) final prompt(s) to query the language model. **Extreme** prompts are typically not favorable choices because one objective is severely compromised to optimize the other. Prompts with the highest fluency often

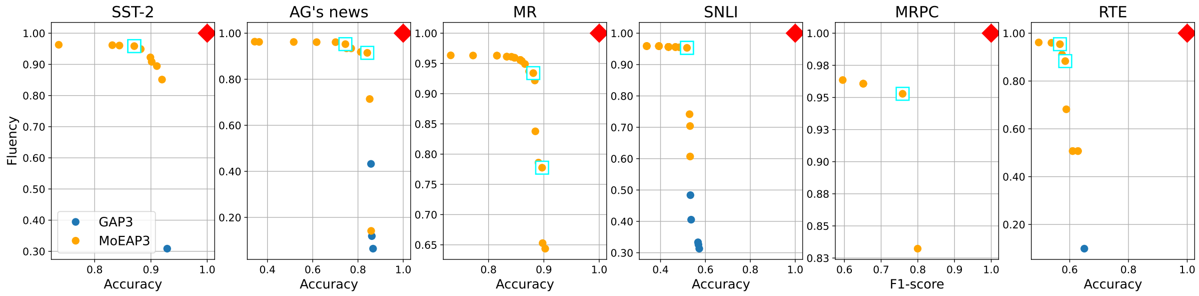

Figure 3: Non-dominated approximation fronts formed by prompts in final populations evaluated on test datasets. Red diamonds denote utopia solutions and cyan squares denote **knee** prompts. We perform 10 independent runs with different random seeds for each task.

score poorly in terms of accuracy of F1. Prompts with the highest performance scores are often unintelligible, and these prompts can be easily obtained with single-objective methods like GAP3 (Zhao et al., 2023).

If there are no particular biases/weights toward certain objectives, **knee** solutions could be promising choices (Branke et al., 2004). In the DPS context, knee solutions are the prompts where a small improvement in readability leads to a large deterioration in performance, or vice versa. Therefore, they are critical solutions that should be considered by decision makers. Figure 4 shows an example of extreme and knee solutions. In order to identify these knees on a bi-objective approximation front, for each non-dominated solution, we compute the angle between the current solution and its two nearest neighbors in the same non-domination rank. If the angle is larger than 210 degree, the prompt is considered a knee solution. We identify knee prompts on the approximation fronts returned by MoEAP3 on six datasets in Figure 3. They represent critical trade-offs between prompt performance and readability.

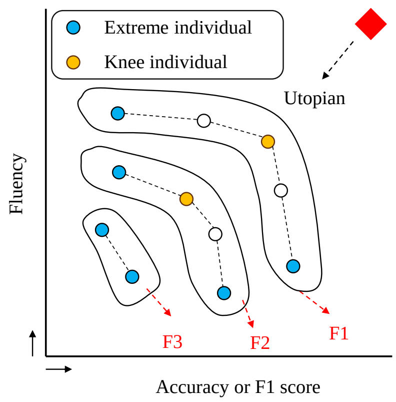

Figure 4: Example of individuals partitioned into non-domination ranks in multi-objective optimization

7 Conclusion

In this paper, we demonstrate that discrete prompt search (DPS) should be formulated as a multi-objective optimization problem to take into account both prompt performance and human-readability. We then propose MoEAP3, which is a gradient-free evolutionary method that can efficiently address the multi-objective DPS in the context of few-shot learning. MoEAP3, while handling both objectives simultaneously, does not suffer from performance drop and even achieves superior results compared to many state-of-the-art baselines. Moreover, the investigation of solution sets of trade-off prompts returned by MoEAP3 is more insightful and intuitive for decision makers.

Limitations

The fluency metric, which is computed based on a classifier fine-tuned on the CoLA dataset, exhibits certain limitations in evaluating the linguistic acceptability of generated prompts. Besides, other readability metrics relevant to text fluency, e.g., repeated phase score, are not considered in this paper. Furthermore, it is challenging to extend MoEAP3 to the cases of more than three objectives because Pareto dominance-based selection would become ineffective as the number of objectives increases. There are indicator-based and reference-based MOEAs which are more suitable to extend MoEAP3 to many-objective scenarios.

Ethics Statement

In this paper, the authors introduce MoEAP3 that samples language models to generate prompts. During mutation, new tokens in the language model vocabulary are inserted into prompts, which may contain offensive, slang, or hate words.

References

David Arthur and Sergei Vassilvitskii. 2007. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, page 1027–1035, USA. Society for Industrial and Applied Mathematics.

J. Blank and K. Deb. 2020. pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509.

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

Jürgen Branke, Kalyanmoy Deb, Henning Dierolf, and Matthias Osswald. 2004. Finding knees in multi-objective optimization. In *Parallel Problem Solving from Nature - PPSN VIII*, pages 722–731, Berlin, Heidelberg. Springer Berlin Heidelberg.

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.

Heyan Chai, Jinhao Cui, Ye Wang, Min Zhang, Binxing Fang, and Qing Liao. 2023. Improving gradient trade-offs between tasks in multi-task text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2565–2579, Toronto, Canada. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric Xing, and Zhiting Hu. 2022. RLPrompt: Optimizing discrete text prompts with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3369–3391, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2022. 8-bit optimizers via block-wise quantization. *9th International Conference on Learning Representations, ICLR*.

Shizhe Diao, Zhichao Huang, Ruijia Xu, Xuechun Li, LIN Yong, Xiao Zhou, and Tong Zhang. 2023. Black-box prompt learning for pre-trained language models. *Transactions on Machine Learning Research*.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. 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 3816–3830, Online. Association for Computational Linguistics.

Nikolaus Hansen, Sibylle D. Müller, and Petros Koumoutsakos. 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evol. Comput.*, 11(1):1–18.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2022. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762, Online. Association for Computational Linguistics.

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.

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.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2023. Gpt understands, too. *AI Open*.

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*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Clara Meister and Ryan Cotterell. 2021. Language model evaluation beyond perplexity. 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 5328–5339, Online. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. In *Advances in Neural Information Processing Systems*, volume 34, pages 11054–11070. Curran Associates, Inc.

Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. 2023. GrIPS: Gradient-free, edit-based instruction search for prompting large language models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3845–3864, Dubrovnik, Croatia. Association for Computational Linguistics.

Prerna Prem, Zishan Ahmad, Asif Ekbal, Shubhashis Sengupta, Sakshi C. Jain, and Roshni Ramnani. 2021. Unknown intent detection using multi-objective optimization on deep learning classifiers. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1130–1137, Held Online. INCOMA Ltd.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Weijia Shi, Xiaochuang Han, Hila Gonen, Ari Holtzman, Yulia Tsvetkov, and Luke Zettlemoyer. 2023. Toward human readable prompt tuning: Kubrick's the shining is a good movie, and a good prompt too? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10994–11005, Singapore. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. 2022. Black-box tuning for language-model-as-a-service. In *Proceedings of ICML*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

Yequan Wang, Jiawen Deng, Aixin Sun, and Xuying Meng. 2022a. Perplexity from plm is unreliable for evaluating text quality. *arXiv preprint arXiv:2210.05892*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022b. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Dirk Weissenborn, Leonhard Hennig, Feiyu Xu, and Hans Uszkoreit. 2015. Multi-objective optimization for the joint disambiguation of nouns and named entities. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics* *and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 596–605, Beijing, China. Association for Computational Linguistics.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Wang Yanggang, Haiyu Li, and Zhilin Yang. 2022. GPS: Genetic prompt search for efficient few-shot learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8162–8171, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tianjun Zhang, Xuezhi Wang, Denny Zhou, Dale Schuurmans, and Joseph E. Gonzalez. 2023. TEMPERA: Test-time prompt editing via reinforcement learning. In *The Eleventh International Conference on Learning Representations*.

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

Jiangjiang Zhao, Zhuoran Wang, and Fangchun Yang. 2023. Genetic prompt search via exploiting language model probabilities. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 5296–5305. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Han Zhou, Xingchen Wan, Ivan Vulić, and Anna Korhonen. 2023. Survival of the most influential prompts: Efficient black-box prompt search via clustering and pruning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13064–13077, Singapore. Association for Computational Linguistics.

A Baseline Implementation Details

We describe the implementation of our baselines:

Manual prompt: We collect handcrafted prompts and their label words from (Sun et al., 2022) as the first baseline.

Instruction: task descriptions manually designed to instruct the language model. We use instructions and their label words from (Mishra et al., 2022; Wang et al., 2022b). The specific instructions are shown in Table 6.

In-context learning (ICL): we sample k random samples per class in the training dataset, and then concatenate them as the model input. We implement ICL following (Gao et al., 2021)

BBT: We re-produce (Sun et al., 2022), we set 50 prompt tokens, the intrinsic dimension is 500, and the population size is 20. The number of API calls is 8,000 to optimize the cross entropy loss function. The templates and label words for BBT are from (Sun et al., 2022).

(BDPL: We use AdamW (Loshchilov and Hutter, 2019) to optimize discrete prompt within 30 epochs, the learning rate is 20^{-4} (for SST-2, AG's News, MR, SNLI) and 10^{-4} (for RTE, MRPC), the prompt length is 50. For SST-2, MRPC, SNLI, and RTE, we apply the templates and label words from (Diao et al., 2023); we use templates and label words of (Sun et al., 2022) for AG's News and MR.

ClaPS: we run ClaPS including 2 phases: search space pruning and training phase (Zhou et al., 2023). In the search space pruning phase, we use the embedding layer of `roberta-large` and encoder embedding layer of `flan-t5-base` to extract the embeddings of PLM vocabulary. Then we apply K-Means++ (Arthur and Vassilvitskii, 2007) to collect 2,000 centroids with the closest word in the embedding space. In the training phase, we implement different optimization algorithms of ClaPS: genetics (ClaPS-GE), particle swarm optimization (ClaPS-P), and greedy search (ClaPS-Gr). We optimize a population of 128 individuals with prompt length is 5 within 30 epochs, we select 64 individuals for mutation and crossover at each epoch.

GAP3: We run GAP3 following (Zhao et al., 2023), the crossover and mutation probability are 0.5 and 0.7 respectively. The population has 64 individuals and the number of generations is 50.

B Ablation study

In this ablation study, we carry out experiments on SST-2 and MRPC datasets. We perform independent runs with different random seeds for each setting.

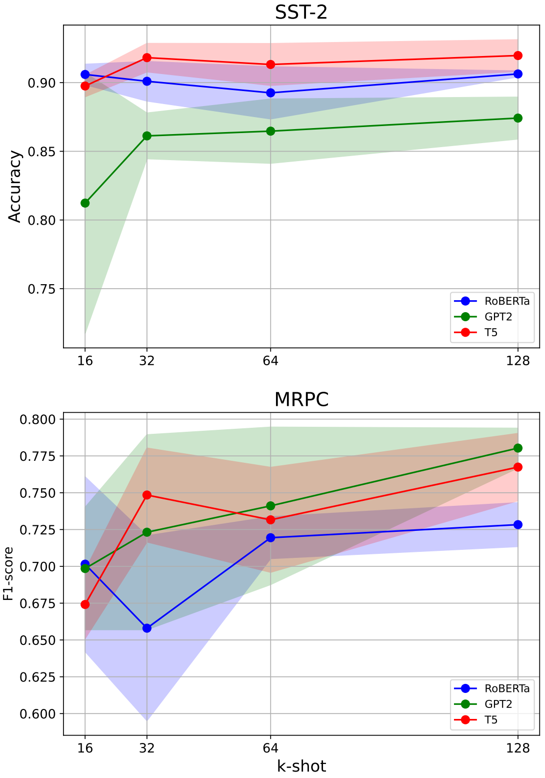

Figure 5: Ablation study on backbone models and k-shot on MoEAP3. We plot mean and standard deviation of the results over three random seeds.

B.1 Different backbone models

We conduct experiments with different backbone transformer models: RoBERTa (an encoder model), GPT-2 (a decoder model), and T5 (an encoder-decoder model). Figure 5 show that GPT-2 has a higher variance over three random seeds compared to RoBERTa and T5. All models yield the highest performance at the largest 128-shot data. T5 exhibits the lowest variance, demonstrating its potential to be a well-suited few-shot learner.

B.2 k-shot

We increase $k = \{16, 32, 64, 128\}$ to show the effect of different k settings to the performance of MoEAP3. Figure 5 shows that, overall, the performance of models is consistent with the size of training datasets. The performance of models with the largest training data outperforms smaller training data settings. In the small 16-shot data, GPT-2 model has the highest variance on SST-2, and the variance decreases over larger training data settings.

Task	**# Label**	**Training size**	**Testing size**	**Template**	**Label words**
SST-2	2	32	0.8K	[$\mathcal{P}$1] [X] [$\mathcal{P}$2] [Y]	good, bad
MR	2	32	1.07K	[$\mathcal{P}$1] [X] [$\mathcal{P}$2] [Y]	good, bad
AG’s News	4	64	7.6K	[X] [$\mathcal{P}$1] [Y]	world, sports business, technology
MRPC	2	32	1.7K	[$\mathcal{P}$1] [X1] [Y] [$\mathcal{P}$2] [X2]	No, Yes
RTE	2	32	0.3K	[$\mathcal{P}$1] [X1] [Y] [$\mathcal{P}$2] [X2]	No, Yes
SNLI	3	48	9.8K	[$\mathcal{P}$1] [X1] [Y] [$\mathcal{P}$2] [X2]	No, Yes, Maybe

Table 4: The dataset statistic, template, verbalizer. # Label denotes the number of labels. $\mathcal{P}$1 and $\mathcal{P}$2 are initialized as empty string, MoEAP3 optimize these prompts to maximize two objective functions in section 4.

Methods	Discrete prompt	Human-free prompt	Readability	Population-based	Multi-objective
Manual Prompt	✓	✗	✓	✗	✗
Instructions	✓	✗	✓	✗	✗
ICL	✓	✗	✓	✗	✗
BBT (v2)	✗	✗	✗	✓	✗
RLPrompt	✓	✓	✓	✗	✗
TEMPERA	✓	✗	✓	✗	✗
GPS	✓	✗	✓	✓	✓
ClaPS	✓	✗	✓	✓	✗
GAP3	✓	✓	✗	✓	✗
Ours	✓	✓	✓	✓	✓

Table 5: The comparison of our proposed with previous methods.

Task	Prompt
SST-2	In this task, you are given sentences from movie reviews. The task is to classify a sentence as "great" if the sentiment of the sentence is positive or as "terrible" if the sentiment of the sentence is negative.
AG’s News	In this task, you are given a news article. Your task is to classify the article to one out of the four topics "World", "Sports", "Business", "Tech" if the article"s main topic is relevant to the world, sports, business, and technology, correspondingly. If you are not sure about the topic, choose the closest option.
MR	In this task, you are given sentences from movie reviews. The task is to classify a sentence as "great" if the sentiment of the sentence is positive or as "terrible" if the sentiment of the sentence is negative.
SNLI	In this task, you’re given a pair of sentences, sentence 1 and sentence 2. Your job is to choose whether the two sentences clearly agree (entailment)/disagree (contradiction) with each other, or if this cannot be determined (neutral). Your answer must be in ’Yes’, ’No’, and ’Maybe’ respectively.
MRPC	You are given two sentences (Sentence1 and Sentence2). Answer "Yes" if these sentences are a paraphrase of one another, otherwise answer "No".
RTE	In this task, you’re given two sentences. Indicate if the first sentence clearly entails the second sentence (i.e., one can conclude the 2nd sentence by reading the 1st one). Indicate your answer with ’Yes’ if the first sentence entails the second sentence, otherwise answer with ’Maybe’.

Table 6: The Instructions for our experiments with both `roberta-large` and `flan-t5-base`.

Method	Log-PPL	Fluency	Perf	Prompt
SST-2				
GAP3	8.94	12.7	91.28	��ア ル シ ャ Hawkins ゴ Ble 方 Authors ��aghhhhosen ークシャ Xen ーク [X] techno ヨ Contin dag dag characterization hereís Just unequivocally extremely [Y]
	9.38	12.10	91.17	��アル irements ゴ Ble 方 Authors ��aghhhhosen ークシャ Xen ーク�[X] gue ����Contin dag kid characterization hereís Just unequivocally extremely [Y]
	9.45	4.59	90.00	uponrailuouslyofferray player purchaser Blu Modray flat disc ク [X] :#tten veryodi VERY [Y]
MoEAP3	7.35	21.05	91.28	wbrama [X] ★★REALLY really [Y]
	7.28	89.45	91.06	our rating [X] ★★Really REALLY [Y]
	7.55	85.12	91.97	My honest IGN rating Wonderful [X] ———— Really really REALLY [Y]
MR				
GAP3	9.17	9.43	87.42	mediocre performer 光 medi NUMiscocreIde mediocreperformance detailsivenessAMIWinner [X] Movie �unflVery denomination deterioration performer Very very [Y]
	10.02	28.75	87.24	hidGGGG hig hig ビ higbbGGGGGGGG Professional SHOWunky degradingEEEEPM [X] ....<unk> feelsplain plain fundament ord downright REALLYreally [Y]
	10.14	16.81	88.18	Post worst94 catastrophicorge wil rac Seasarc の �criticised �george displENTjohnjonghelcot QuoteessionsRichailing ···[X] feels reallyreally,reallyreallyreally [Y]
MoEAP3	7.56	91.69	87.43	�Tinder [X] ★★Really ridiculously [Y]
	6.50	93.40	88.09	★★Broken glass [X] ★★Totally [Y]
	7.42	48.99	88.37	Rating nuns �[X] Veryveryvery veryvery very [Y]
AG's News				
GAP3	10.82	2.50	83.98	[X] <unk> NZ tricks<unk><unk> digest �noon GoldmanIVE Tele 470avers PatriotsialsollarSIMNiamus reliant McF Ratt �gallelsen commentary insight contemporaryclassified [Y]
	8.59	8.02	83.78	[X] �Cameraipe Changing times impacting Throw evolvingomed Contin tipping Challenges unequ footsteps Challenges facing mat Related 裏覚醒 Related allchukfreedom [Y]
	11.80	6.48	84.49	[X] Colts attributablearedevilJeff') Den Continue ド guiIcon<unk>VERTISEMENT————- Modernst Changing day news unregulated concerning [Y]
MoEAP3	9.41	3.31	84.24	[X] ERY �CAP �COL MUSTOUS READ MORE How insane politics() poisons [Y]
	8.42	1.67	83.88	[X] Dragonbound today ���: hot topics in [Y]
	9.79	5.70	83.61	[X] �Understanding dirty creep hitherto unseenlocking conflicts VS mainstream [Y]

Figure 6: Example prompts with top performance on test datasets of single-sentence classification tasks.

Method	Log-PPL	Fluency	Perf	Prompt
SNLI				
GAP3	9.30	23.32	48.55	Therefore IPA Prim Pole�backstory Consider �olarUltimate Possible PesCase C:" assailant Hath [X1] [Y] .... Why,oulder [X2]
	9.37	1.25	52.05	Therefore IPAwhose Primistically Pole Something Neck Consider �olarUltimate Possible Worst externalToEVAOnlyomsday Hugenario assailant Hath [X1] [Y] !!!! Alt Why,oulder [X2]
	9.41	1.45	56.04	Meaning censorasar �zanne Borg�GiovanniMathline Dum�lin</s>OL Ont Eliot Quote IPA quotation quotation Locke," Nep [X1] [Y] !!!! [X2]
MoEAP3	9.73	94.63	43.34	Yeah [X1] [Y] Immediately [X2]
	7.46	84.15	47.40	Yep........ Valid Answer [X1] [Y] !!!! Definitely [X2]
	8.41	52.74	55.07	Yep)] [X1] [Y] !!!! Hur [X2]
RTE				
GAP3	10.70	24.54	58.12	Yeah 1850 elltta Restorationerieaina [X1] [Y] ieri...tem �............ positively SER 374 �OC!!!!!!!! dred ty!!!!!!!! [X2]
	11.17	16.39	61.02	hensurities Clar billionaire representedARY Editororterudge Morning ounce]) PublisherLotPresYork [X1] [Y] !!!!!!!! denial FN0000000000000000!!!!!!!! Ther indeed (),[X2]
	12.02	13.92	57.04	Resp Answers �Candidate e (>oyd Rebirth ________ neg Ku JudicialpaUFpaUFPU �Vale [X1] [Y] !!!!!!!! consequentlyises arg き oss Rather!!!! [X2]
MoEAP3	7.83	37.66	58.84	YepYep [X1] [Y] ises!!!! ow [X2]
	8.95	50.71	62.85	Yeah ��[X1] [Y] ILY Rather!! [X2]
	7.83	62.07	57.40	York Yep Yep LR [X1] [Y] Hmm [X2]
MRPC				
GAP3	10.20	13.18	72.27	Yah NeOTHER Cum//////////////// quem mills Favor!!!!! [X1] [Y] Ne-|!!! disputesooting Allow petitionsrals Fr Allow Forum Policyaciesction Adsusted[X2]
	9.84	8.27	76.62	Yeprals Choose colour disclaimverning048 questionsacket selectedicle Yep.) x19 [X1] [Y] Pastates Wrticket �dirhhh �Occasionally pa Eh Unsure Err Err [X2]
	10.90	24.84	79.21	�verified Kinnikuman restgeon Mace� • • letalheaders��[X1] [Y]x18nette .... Spring .... OWx18 ())!!! Indeed Mace イ [X2]
MoEAP3	8.41	95.28	75.86	Say Leilan x18 [X1] [Y]x18 Nayx18 [X2]
	9.14	83.23	79.96	Peslizu [X1] [Y]izu Hmm [X2]
	9.77	9.53	76.90	Alberto deducted +—FS))) 148avour174 Tav [X1] [Y] !) ;) �Correct opin Toll Firstly [X2]

Figure 7: Example prompts with top performance on test datasets of sentence-pair classification tasks.