ARE BERT FAMILIES ZERO-SHOT LEARNERS? A STUDY ON THEIR POTENTIAL AND LIMITATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Starting from the resurgence of deep learning, language models (LMs) have never been so popular. Through simply increasing model scale and data size, large LMs pre-trained with self-supervision objectives demonstrate awe-inspiring results on both task performance and generalization. At the early stage, supervised fine-tuning is indispensable in adapting pre-trained language models (PLMs) to downstream tasks. Later on, the sustained growth of model capacity and data size, as well as newly presented pre-training techniques, make the PLMs perform well under the few-shot setting, especially in the recent paradigm of prompt-based learning. After witnessing the success of PLMs for few-shot tasks, we propose to further study the potential and limitations of PLMs for the zero-shot setting. We utilize 3 models from the most popular BERT family to launch the empirical study on 20 different datasets. We are surprised to find that a simple Multi-Null Prompting (without manually/automatically created prompts) strategy can yield very promising results on a few widely-used datasets, e.g., $86.59\%(\pm 0.59)$ accuracy on the IMDB dataset, and $86.22\%(\pm 2.71)$ accuracy on the Amazon dataset, which outperforms manually created prompts without engineering in achieving much better and stable performance with the accuracy of $74.06\%(\pm 13.04)$, $75.54\%(\pm 11.77)$ for comparison. However, we also observe some limitations of PLMs under the zero-shot setting, particularly for the language understanding tasks (e.g., GLUE).

1 INTRODUCTION

In recent years, the Natural Language Processing (NLP) community have witnessed the explosive development of pre-trained language models (PLMs) such as GPT series (Radford et al., a;b; Brown et al., 2020), BERT family, represented by (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2019), and encoder-decoder models like BART (Lewis et al., 2020) and T5 (Raffel et al., 2020), and marvelled at their impressive task performance and generalization ability. Through simple post-tuning, the rich knowledge in PLMs can be effectively transferred to various downstream tasks, i.e., the post-tuning process can fill the gap between pre-training objectives and task utilization. However, the scale of recent proposed PLMs can easily spiral up to hundreds of billions and even over one trillion parameters, resulting in a dilemma that adequate post-tuning of PLMs on such a scale is difficult to afford while inadequate tuning will significantly decrease the adaptation performance.

To resolve the above dilemma, various prompt-based methods are introduced (Brown et al., 2020; Schick & Schütze, 2021a;b; Gao et al., 2021; Hambardzumyan et al., 2021) to reduce the gap between pre-training and task utilization by reformulating the downstream tasks to be more like the pre-training process. With a more unified task formulation, the post-tuning efforts are significantly reduced. Take a simple text classification case for example. Given a sentence *"This food is delicious!"* as input, conventional methods based on PLMs require supervised post-tuning to fill the gap between masked language model (MLM) objective (i.e., predicting the masked word of *"This food is [MASK]!"*) and the expected classification output of $0, 1, \ldots, n$. Alternatively, prompt-based learning transforms the input text into the MLM task by appending a prompt, e.g., *"This food is delicious! The sentiment of this sentence is [MASK]"*, to make PLMs perform prediction directly, where the expected classification outputs are extracted from the output of the masked position, e.g., re-scaling the probabilities of a pre-defined label word pair *"positive, negative"*. In doing so, the efforts of post-tuning PLMs and training new modules for downstream tasks are no longer needed but requiring selective prompts and effective strategy to extract task output from the MLM prediction.

Through exploring three different paradigms of prompt designing strategies (*hand-crafted, discrete prompt, continuous prompt*) and various answer engineering (e.g., selecting label words for the classification tasks), prompt-based methods have achieved very impressive and appealing results on the BERT family, especially for the few-shot setting (Schick & Schütze, 2021a;b; Gao et al., 2021; Liu et al., 2021b; Perez et al., 2021). With the success of prompting methods in the few-shot setting, it is natural to explore their effectiveness in a more challenge setting, i.e., the zero-shot scenario.

Our Contributions. We (1) launch a thorough empirical study to test the zero-shot capabilities of representative PLMs from BERT family; (2) propose a few simple yet effective strategies to improve the zero-shot performance and robustness (3) along with coarse-to-fine studies to analyze and understand the effect of each strategy; (4) observe a few essential limitations and give the possible reasons, which might shed light on the future work of zero-shot prompting on the BERT family.

Most Surprising Findings. Though the designs of experiments are simple and straightforward, we still observe a few surprising findings. First, Multi-Null Prompt, a simple modification of the Null Prompt method by concatenating multiple [MASK] tokens, can outperform manual prompts in text classification tasks. More surprisingly, without using [MASK] tokens, inserting multiple random tokens even achieve improved performance. Besides, even breaking the form of natural language, inserting multiple [MASK] tokens at the random position of the text also performs well.

Paper Structure. The rest of this paper is organized as follows: We present some preliminaries about the utilization of PLMs in Section 2, and we test some basic models adapted from the few-shot setting and propose two new strategies for zero-shot setting in Section 3, following with a coarse-to-fine study in Section 4. Section 5 and 6 present the possible limitations of the BERT family.

2 PRELIMINARIES

To study and calibrate the effect of prompt-based methods for the zero-shot setting, we first present two typical paradigms that perform well for supervised post-tuning and the few-shot setting, including fine-tuning methods and prompt-based learning. These methods will be extended to the zero-shot scenario in Section 3 to calibrate the potential of prompting augmented with simple strategies.

2.1 FINE-TUNING OF PLMS

Starting from the occurrence of large-scale PLMs, fine-tuning methods have almost been indispensable and are still strong competitors in recent papers to mitigate the gap between the pre-training objectives and downstream tasks (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2019; Radford et al., a;b; Brown et al., 2020; Raffel et al., 2020). For text classification and natural language understanding tasks, PLMs are mainly utilized to obtain the contextualized representation of text. Each given text input $x = (x_1,x_2,\dots)$ first concatenates a $[CLS]$ token and a $[SEP]$ token at the beginning and the end of x, respectively. Then, PLMs process the input text $([CLS], x, [SEP])$ to obtain the corresponded vector representations $h = (h_{[CLS]},h_1,h_2,\dots,h_{[SEP]})$. In most cases, either $h_{[CLS]}$ or the output of mean pooling over each element of h is utilized as the aggregated representation of input text x, denoted as h_x, which is further processed by task-specific modules to obtain the expected output label y. The task-specific modules are usually implemented as a stack of multiple linear layers (MLPs) along with a softmax classifier to compute the output labels, written by

$$p(y|x) = \texttt{softmax}(W \cdot \texttt{MLP}(h_x)),$$

where W refers to trainable parameters of the classifier, which are trained from scratch.

2.2 PROMPT-BASED METHODS

Instead of performing fine-tuning to adapt PLMs to different downstream tasks, prompt-based learning leverages prompt to reformulate the downstream tasks to be more like the pre-training task, which is usually a language modeling task. Formally, the input text from the downstream task is first mapped to a prompt by a prompting function $f_{prompt}(\cdot)$ which comprises two stages:

- A template is first constructed and applied to combine prompt tokens, input text slot [X], and answer slot [Y]. For instance, *"[X] The sentiment of this sentence is [Y]"* is a feasible template for the sentiment classification task, where [Y] is [MASK] for the BERT family.
- Then, $f_{prompt}(\cdot)$ fills the slot [X] with the input text, e.g., converting the above-mentioned template to *"This food is delicious! The sentiment of this sentence is [Y]"*.

After mapping the input text to prompt, the model predicts the token in the position of answer slot [Y], which resembles predicting possible tokens during the pre-training stage. Different from predicting possible tokens in the whole vocabulary in pre-training, prompt-based learning only calculates the possibility of the pre-set tokens, called label words. Take the aforementioned prompt template for example. The pre-set tokens can be *"positive"* and *"negative"*. Then, a *Verbalizer* (Liu et al., 2021a) maps the highest-scoring label word (e.g., *"positive"* for *"This food is delicious! The sentiment of this sentence is [Y]"*) to its corresponding label indices $\{0, 1, \ldots, n\}$ as the output.

3 On the Potential of Prompting for Zero-Shot Setting

In this section, we first present a few basic models adopted from existing works and their implementation details. We then test the performance of these basic models on four widely utilized datasets under the zero-shot setting. After calibrating the zero-shot performance of existing methods from other settings, we introduce two simple strategies upon the basic models for further exploration.

3.1 Settings

Basic Models. In the fine-tuning of PLMs, task-specific models mainly comprise two essential components, i.e., PLMs for mapping input text to the contextualized representation and the task-specific classifier learned from scratch. In the prompt-based learning, the input text is represented by the output hidden state of PLMs at the position of the answer slot [Y], and the task classifier is part of the PLMs, i.e., the MLM head, which is utilized to compute the probabilities of the pre-set label words. In the zero-shot setting, it is inapplicable to train a task-specific classifier, and thus the MLM head classifier utilized in prompt-based learning is a natural and feasible alternative. To this end, we create three basic models based on different representation aggregation strategies in conventional methods of fine-tuning PLMs, including

- **CLS.** The output hidden state of the [CLS] token, i.e., the first token before the input text, is utilized as the aggregated representation of the input text.
- **SEP.** The output hidden state of the [SEP] token, i.e., the token appended to the end of the input text, is used as the aggregated representation of the input text.
- **Mean Pooling.** The hidden states of all tokens in the input text (without [CLS] and [SEP]) are averaged as the representation of the input text.

The aggregated representations of the input text are then processed by the MLM head classifier to predict task-specific labels. Note that we do not utilize prompt to process the input text in the above three basic models and only borrow the MLM head classifier and the pre-set label words from the prompt-based learning to replace the task-specific output layer that requires supervised training.

As for the prompt-based learning, we also extend three representative few-shot methods from existing literature and implement the following methods in zero-shot prompt-based learning.

- **Manual Prompt (Prior).** We use the handcrafted prompts designed by Schick & Schütze (2021a;b); Hu et al. (2021), which are tuned using large validation sets or selected by validation data (Logan IV et al., 2021). The utilized prompts are given in Appendix A.1.
- **Manual Prompt$^{\#}$.** We also manually design a few intuitive prompts that are not selected carefully using validation sets to study the influence of prompt engineering for the zero-shot setting, where the manually designed prompts are presented in the Appendix A.1.
- **Null Prompt.** Following Logan IV et al. (2021) to simplify prompt engineering, we only insert a [MASK] token at the end of the text, i.e., using the prompt template *"[X][MASK]"*.

We also introduce three models that use unlabeled data or external knowledge from existing works.

Table 1: The zero-shot performance of basic models and our proposed strategies on IMDB, Amazon, AG News, DBPedia corpora, where $*$ refers to utilizing more correlated label words (here 30, 60, 90 correlated label words), and the average results are reported with standard deviation.

Method	IMDB	Amazon	AG News	DBPedia	Average	Human	Unlabeled
CLS	50.20	50.22	27.26	11.19	34.72	✗	✗
SEP	51.76	50.48	44.63	29.69	44.14	✗	✗
Mean Pooling	72.22	63.14	36.84	58.41	57.65	✗	✗
Manual Prompt$^{\#}$	$74.06_{13.04}$	$75.54_{11.77}$	$67.65_{5.77}$	$56.75_{19.23}$	68.50	✓	✗
Manual Prompt (Prior)	$88.98_{4.12}$	$82.29_{11.08}$	$66.42_{9.58}$	$69.25_{15.95}$	76.74	✓	✗
Null Prompt	66.84	82.79	54.64	55.98	65.06	✗	✗
Null Prompt*	$82.47_{1.48}$	$89.36_{0.97}$	$67.87_{1.97}$	$56.75_{3.86}$	74.11	✗	✗
Multi-Null Prompt	78.26	85.05	50.01	69.67	70.75	✗	✗
Multi-Null Prompt*	$86.59_{0.59}$	$86.22_{2.71}$	$68.15_{1.81}$	$67.58_{1.78}$	77.14	✗	✗
NSP-BERT (Sun et al., 2021)	$72.82_{1.13}$	$72.68_{3.93}$	$77.39_{0.63}$	$64.65_{5.31}$	71.89	✓	✗
LOTClass (Meng et al., 2020)	86.50	91.60	86.40	91.10	88.90	✗	✓
LOTClass (w/o self-train)	80.20	85.30	82.20	86.00	83.43	✗	✓
KPT (Hu et al., 2021)	91.50	92.50	83.00	82.50	87.38	✓	✓

- **NSP-BERT.** Sun et al. (2021) reformulate text classification task into text entailment-style tasks and then use the Next Sentence Prediction (NSP) head to predict the result. The templates we used are shown in Appendix A.1.
- **LOTClass.** Meng et al. (2020) use unlabeled data to find the words similar to label names and introduce a self-training approach to induce a classifier.
- **KPT.** Hu et al. (2021) propose a prompt-based method that uses external knowledge to expand the space of label name.

Datasets. We carry out experiments on several widely-acknowledged datasets targeting different aspects of text classification to study the performance of the above-described methods and evaluate newly proposed strategies that are designed for exploring the potential of zero-shot setting. Specifically, we follow Meng et al. (2020) and systematically evaluate these methods on 2 topic classification tasks and 2 sentiment classification tasks, which are *IMDB* (Maas et al., 2011), *Amazon* (McAuley & Leskovec, 2013), *AG News* (Zhang et al., 2015), and *DBPedia* (Lehmann et al., 2015). For each task, we only leverage the test set to directly evaluate model performance without introducing validation or training sets to perform post-tuning or cherry-pick hand-crafted prompts.

Implementation Details. We mainly conduct experiments using *roberta-large* as the backbone pre-trained LM. To confirm the generalization of the proposed method, we also report the results of *bert-base-uncased* and *albert-xxlarge-v2*, which will be discussed in the next section. Our model implementation is based the open toolkit Huggingface Transformers (Wolf et al., 2020) [1]. To load the data of different text classification tasks, we also use the open toolkit Huggingface Datasets[2] (Lhoest et al., 2021). More accurately, we use *imdb*, *amazon_polarity*, *ag_news*, *dbpedia_14*.

3.2 Results and Explorations

Overall Performance of Basic Models. As shown in Table 1, the combination of simple representation aggregation methods and MLM head classifier along with the pre-set label words can only achieve a preliminary performance, represented by the low accuracy of CLS and SEP models (34.72% and 44.14% on average). Through utilizing all tokens in the input text, Mean Pooling performs much better and is stable than the previous two methods, resulting in an accuracy of 57.68% on average. With simplified human efforts for designing prompts, the Manual Prompt$^{\#}$ significantly improves the performance. The cherry-picked manual prompts further increase the power of the Manual Prompt method. Without human efforts for designing prompt, the Null Prompt method (only with an extra [MASK] as the prompt token) yields nearly comparable results with Manual

[1] https://github.com/huggingface/transformers
[2] https://github.com/huggingface/datasets

Table 2: Performance of inserting different number of prompt [MASK]s at a random position.

Number	IMDB	Amazon	AG News	DBPedia	Average
1	59.14	63.02	44.82	41.78	52.19
2	62.24	65.78	48.09	49.83	56.49
3	63.78	67.21	48.47	53.83	58.32
4	65.36	68.08	49.51	56.24	59.80

Prompt$^{\#}$ but outperforms these non-prompting zero-shot methods. Through leveraging extra knowledge and unlabelled data, LOTClass and KPT achieve the best results, which are almost comparable with SOTA few-shot learning methods and even supervised learning models. In short, unlabeled data and extra knowledge are most effective for the zero-shot setting, and human efforts take second place. Prompt-based learning (even the Null Prompt setting) is more effective than tuning-based zero-shot methods. However, neither much human effort nor unlabelled data, nor extra knowledge is ideal for realizing the full potential of PLMs in generalizing to different downstream tasks. Thus, we will focus on the Null Prompt setting in the rest of this section to explore the potential of PLMs.

Recall that, in our challenging setting, we cannot utilize any external information other than label name, which is necessary to tell the model to classify the text from which aspect. To fully use the PLMs knowledge, we propose two simple yet effective strategies at different inference stages, which can be utilized simultaneously. Their effectiveness is studied in the following paragraphs.

Ensemble of Multiple-Null Prompt Masks. We first propose to enhance the capability of the Null Prompt method with multiple prompt [MASK] tokens[3], named Multi Null Prompt, i.e., inserting multiple prompt [MASK] at different positions of the input text (e.g., the end, the head, the middle). The multiple answers from the prompt masks are aggregated by different ensemble methods. The positions of prompt [MASK] and ensemble strategies will be thoroughly studied in Section 4. Table 1 presents the results of inserting one single [MASK] at both the beginning and the end of the input text, i.e., in the form of *"[MASK][X][MASK]"*. We can see that the simple Multi Null Prompt method results in substantial performance gains over the Null Prompt basic model.

Searching for More Label Words. This strategy directly utilizes cosine distance as the metric to search and pick up the words with similar embeddings to the label names. It can be easily seen from Table 1 that with the enhancement of more label words, the Null Prompt method gains 9.05% on average, which is almost comparable with the Manual Prompt (Prior) that needs considerable human efforts. Experimental result on the Multi-Null Prompt model also shows significant improvement, which even performs better than Manual Prompt with non-trivial human efforts.

We can conclude from these experimental results that simple strategies without human efforts or extra knowledge or unlabelled data can also release the potential of PLMs and significantly enhance the zero-shot performance. It is possible to design more general and effective strategies to facilitate the zero-shot utilization of PLMs, even for the BERT family.

4 Coarse-to-Fine Study of Prompting Strategies

In this section, we conduct coarse-to-fine studies to analyze the reasons behind the impressive performance of the simple Multi-Null Prompt method, including the position of prompt [MASK], the number of prompt [MASK], the possible alternative of prompt [MASK], the effect of similar label words, and the influence of different ensemble methods for multiple predicted answers. We also conduct experiments on different backbone PLMs to test the generalization of the proposed method.

4.1 The Position and Number of Prompt [MASK]

We first study the effect of the position and number of the prompt [MASK]. Figure 1 shows that inserting [MASK] tokens at both the start and the end of the text outperforms inserting at the start

[3]Schick & Schütze (2021a) also utilize multiple [MASK] tokens to deal with the situation that the label words are comprised of multiple sub-words, which is quite different from ours.

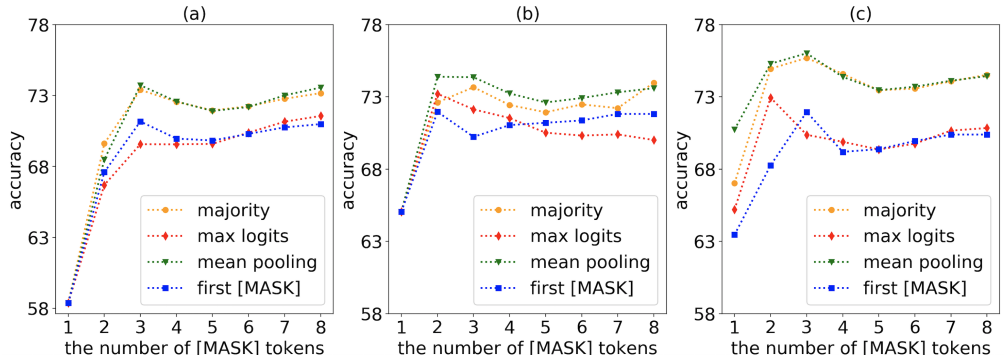

Figure 1: The average performance on text classifications with different ensemble methods and number of [MASK]s. (a) shows the results of inserting different number of [MASK]s at start position, while (b) is at the end. (c) inserts the same number of [MASK]s at both start and end positions.

or end separately. Since we do not utilize manual prompts, we do not need to follow prior prompt-based methods to insert prompts without breaking the form of natural language. Therefore, we insert the [MASK] tokens at the random position of the text. The results are reported in Table 2, which shows that even though breaking the form of natural language, it can still achieve promising performance, and the performance improves as the number of [MASK] tokens increases. We also study the changes in the performance when the number of [MASK] tokens differs. Figure 1 shows that Multi-Null Prompt is robust to the number of [MASK] tokens.

4.2 The Possible Alternative of Prompt [MASK]

Table 3: The results on IMDB, Amazon, AG News, DBPedia when using different answer slots .

Answer Slot [Y]	IMDB	Amazon	AG News	DBPedia	Average
MASK	86.94	87.81	68.80	70.36	78.48
.	66.54	56.48	62.63	68.00	63.41
M	86.52	88.18	67.26	69.95	77.98
star	87.71	87.88	67.55	71.13	78.57
First	87.39	86.70	69.76	73.86	79.43
med	87.05	79.92	66.58	70.87	76.11
Industrial	86.88	87.45	68.14	71.42	78.47
hardest	86.94	88.12	68.83	69.19	78.27
frightening	86.82	88.00	68.51	69.93	78.32

We also explore the question that *Must we use [MASK] token as answer slot [Y]?* The results are reported in Table 3. In this experiment, we insert one token as answer slot [Y] at both the start and end of the text and utilize 50 correlated label words. The results show that even though we do not use [MASK] token as answer slot [Y], Multi-Null Prompt still works and achieves even better performance. This is surprising because all the prior prompt-based methods use [MASK] token as answer slot [Y]. We observe that compared to other tokens, the token '.' achieve the worst performance. We speculate that this phenomenon is caused by the intrinsic attribute of MLM objectives. MLM task not only substitutes tokens with prompt [MASK]s but also with random tokens, which make the BERT family have the ability to recognize the mistaken tokens. Hence, if we insert random tokens, the model can recognize inserted tokens easily and represent them as [MASK] tokens, while the model may treat the token as a reasonable element of the whole text rather than the prompt [MASK] if we insert token like '.'.

4.3 The Effect of Different Ensemble Methods for Multiple Prompt Answers

To utilize multiple prompt answers from different prompt [MASK]s, we implement four different ensemble methods, including 1) **Single Position.** Only one of the prompt [MASK] positions is

Table 4: The performance of different PLMs on IMDB, Amazon, AG News, DBPedia.

Model	Method	IMDB	Amazon	AG News	DBPedia	Average
BERT-base	Null Prompt	51.76	51.79	47.82	38.80	47.54
BERT-base	Multi-Null Prompt	63.37	69.90	55.92	33.03	55.56
RoBERTa-large	Null Prompt	66.84	82.79	54.64	55.98	65.06
RoBERTa-large	Multi-Null Prompt	86.94	87.81	68.80	70.36	78.48
ALBERT-xxlarge	Null Prompt	72.87	63.51	54.01	62.53	63.23
ALBERT-xxlarge	Multi-Null Prompt	65.60	65.87	68.88	66.27	66.66

utilized to obtain the prompt answer, which is then processed to output the task-specific label; 2) **Mean Pooling.** The output logits at all prompt [MASK] positions are averaged as the ensemble output; 3) **Max Logits.** The prompt [MASK] position with the maximum probabilities of the preset label words (i.e., the sum of all the probabilities of each label word) is selected for computing the task-specific label; 4) **Majority.** The majority voting result of different prompt answers is utilized as the final output. As presented in Figure 1, the mean pooling method performs best in aggregating the outputs from multiple prompt [MASK]s, and the majority method takes second place.

4.4 The Effect of Similar Label Words

In this experiment, we study the effect of the number of similar label words, and the results are shown in Figure 2. From the figure, we can conclude that the performance does not change so much when the number of similar label words varies, which proves that searching for more label words by the cosine similarity of embeddings is a simple but effective and robust method to improve the performance.

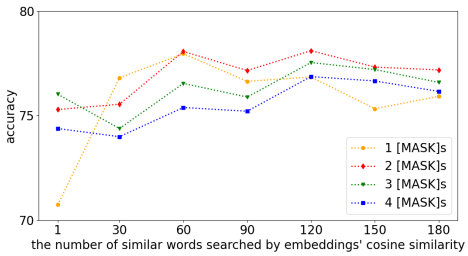

Figure 2: The average performance on text classifications when the number of similar words differing. **1/2/3/4 [MASK]s** means inserting one/two/three/four [MASK] tokens at both the start and end positions.

4.5 Model Variants

To further confirm the effectiveness of Multi-Null Prompt, we also use BERT and ALBERT to conduct experiments on IMDB, Amazon, AG News, DBPedia. The results are reported in Table 4. The significant improvement on different PLMs shows that the benefits of Multi-Null Prompt are brought from MLM tasks and can be applied to BERT families.

5 Possible Limitations

To further study the prompt-based methods in zero-shot settings, we also conduct experiments on NLU benchmarks, GLUE, and SuperGLUE. The results are shown in Table 5 and Table 6[4]. The results show that prompt-based methods fail in the more challenging natural language understanding tasks, though achieving promising results in zero-shot text classification. Therefore, in this section, we analyze possible limitations of prompt-based methods in zero-shot settings by comparing results on different types of tasks and few-shot learning methods and try to give some possible reasons.

5.1 Experiment Results of Different Kinds of Tasks

To explore how prompt-based methods perform on different kinds of tasks, we conduct experiments on various kinds of tasks, including single-sentence tasks (SST-2, CoLA), inference tasks (MNLI, QNLI,RTE), similarity&paraphrase tasks (MRPC, QQP, STS-B, CB), question answering

[4]For SuperGLUE, we only report the results of BoolQ, CB, MultiRC, and WiC as the label words of COPA, WiC and ReCoRD need multiple [MASK] positions, which is conflicted with Multi-Null Prompt. The metrics used for GLUE are reported in Appendix A.1.

Table 5: The experiment results on the validation set of GLUE.

Method	SST-2	CoLA	MNLI	MNLI-mm	MRPC	QNLI	QQP	RTE	STS-B
Majority	50.9	0.0	32.7	33.0	81.2	49.5	0.0	52.7	–
Manual Prompt	70.6	2.2	49.4	50.2	44.2	50.7	46.6	53.8	−20.6
Null Prompt	49.1	−2.9	36.6	36.9	7.4	55.4	16.3	46.9	−11.8
Null Prompt*	$79.1_{4.0}$	$-1.1_{2.0}$	$33.1_{0.4}$	$33.8_{0.5}$	$12.9_{7.0}$	$50.7_{0.1}$	$1.3_{1.0}$	$47.2_{0.6}$	$-16.8_{1.2}$
Multi-Null Prompt	75.0	0.0	39.2	40.5	4.1	54.6	15.8	48.7	−23.7
Multi-Null Prompt*	$70.2_{7.7}$	$6.2_{2.0}$	$38.0_{3.5}$	$38.5_{4.1}$	$19.9_{8.7}$	$52.2_{1.7}$	$25.5_{13.4}$	$53.0_{2.2}$	$-18.6_{1.6}$
Fine-tuning (Few-Shot)	$81.4_{3.8}$	$33.9_{14.3}$	$45.8_{6.4}$	$47.8_{6.8}$	$76.6_{2.5}$	$60.2_{6.5}$	$60.7_{4.3}$	$54.4_{3.9}$	$53.5_{8.5}$
Prompt-tuning (Few-Shot)	$92.7_{0.9}$	$9.3_{7.3}$	$68.3_{2.3}$	$70.5_{1.9}$	$74.5_{5.3}$	$60.2_{6.5}$	$65.5_{5.3}$	$69.1_{3.6}$	$71.0_{7.0}$

tasks (BoolQ, MultiRC), word sense disambiguation tasks (WiC)[5]. Table 5 and Table 6 show that manual prompts can achieve much better performance than majority class in SST-2, MNLI, QQP, comparable performance than majority class in CoLA, QNLI, RTE, BoolQ, CB, MultiRC, WiC, a worse performance than majority class in MRPC. Besides, without the help of prior knowledge in manual prompts, Multi-Null Prompt can achieve comparable results to manual prompts, except for better performance in SST-2, CoLA, and worse performance in MRPC, QQP, BoolQ.

5.2 The Possible Reasons

For inference tasks and similarity&paraphrase tasks, we suspect the difficulty of designing high-quality label words that can represent the meaning of the label well may cause poor performance. Since we conduct experiments in zero-shot settings, the label words are the only source for the model to achieve the task-specific information, which has a considerable impact on the performance. The importance of label words is also emphasized in other different low resource settings, which proves our hypothesis to some extent (Meng et al., 2020; Le Scao & Rush, 2021). Meng et al. (2020) leverage unsupervised data and introduce that if label names can not well explain the meaning of the label, the model can not predict well in zero-shot settings. Le Scao & Rush (2021) focus on few-shot learning, and they find that verbalizer brings more benefits than prompt for small-scale data tasks.

For question answering tasks, where the prompt and label words can express the meaning of the label clearly, the failure of prompt-based methods may be caused by the scaling-up since GPT-3 also fail to achieve promising results at comparable scaling-up. For word sense disambiguation tasks, there may exist various possible reasons. Since GPT-3 also performs like a random choice, Brown et al. (2020)) argues that this may be due to GPT-3's disability of the bidirectionality. However, even though RoBERTa is bidirectional, it also fails to achieve promising results. Model scaling-up may be another reason, but no BERT-like model has comparable scaling-up to GPT-3 to verify this point. Besides, since prompt can help PLM recall the data knowledge during pre-training (Zhong et al., 2021), another possible reason is that there is no data similar to the task in pre-training.

5.3 Zero-shot Prompt-based Methods v.s. Few-shot Learning Methods

Comparing zero-shot prompt-based methods to few-shot learning methods on various tasks, we find that few-shot learning can improve performance considerably. Therefore, although zero-shot prompt-based methods achieve promising performance on some tasks, there is still a big gap between zero-shot prompt-based methods and few-shot learning methods, which shows that the current PLMs still need more supervision than label names to deal with various downstream tasks.

6 Related Work

The few-shot and zero-shot capabilities of GTM series (Radford et al., a;b; Brown et al., 2020) make prompt-based learning draw significant attention in the field of large-size language model pre-training and adaptation. Existing works of prompt-based learning are mainly from the following two angles, i.e., prompt engineering and answer engineering [6].

[5]For these tasks, the manual prompts and label words we used are shown in the Appendix A.1.

[6]More related work and other perspectives can be found in the recent survey (Liu et al., 2021a)

Table 6: The experiment results on the validation set of SuperGLUE.

Method	BoolQ (Acc.)	CB (Acc./F1)	MultiRC (EM/F1a)	WiC (Acc.)
Majority	62.2	0.5 / 22.2	3.2 / 0.0	50.0
GPT-3 Small (Zero-Shot)	49.7	0.0 / 0.0	4.7/ 57.0	0.0
GPT-3 Med (Zero-Shot)	60.3	32.1 / 29.3	9.7 / 59.7	0.0
GPT-3 Large (Zero-Shot)	58.9	8.9 / 11.4	12.3 / 60.4	0.0
GPT-3 175B (Zero-Shot)	60.5	46.4 / 42.8	27.6 / 72.9	0.0
GPT-3 Small (Few-Shot)	43.1	42.9/ 26.1	6.1 / 45.0	49.8
GPT-3 Med (Few-Shot)	60.6	58.9 / 40.4	11.8 / 55.9	55.0
GPT-3 Large (Few-Shot)	62.0	53.6 / 32.6	16.8 / 64.2	53.0
GPT-3 175B (Few-Shot)	77.5	82.1 / 57.2	32.5 / 74.8	55.3
Manual Prompt	63.2	39.3 / 22.3	4.7 / 30.5	50.0
Null Prompt	38.4	33.9 / 29.2	4.1 / 33.6	50.5
Null Prompt*	$38.2_{0.5}$	$41.7_{19.3}$ / $30.2_{9.5}$	$2.8_{2.1}$ / $19.0_{14.2}$	$49.7_{0.3}$
Multi-Null Prompt	38.4	55.4 / 46.0	4.9 / 27.7	50.3
Multi-Null Prompt*	$38.5_{0.9}$	$44.6_{17.6}$/ $31.0_{5.5}$	$3.1_{1.7}$ / $18.8_{9.2}$	$49.7_{0.2}$
PET (Few-Shot) (Schick & Schütze, 2021b)	79.4	85.1 / 59.4	37.9 / 77.3	52.4
iPET (Few-Shot) (Schick & Schütze, 2021b)	80.6	92.9 / 92.4	33.0 / 74.0	52.2

Prompt Engineering Prompt engineering aims to construct a better prompt function to improve the performance of the downstream tasks. To this end, Petroni et al. (2019); Brown et al. (2020); Schick & Schütze (2021a;b) create prompt manually to handle various tasks, including knowledge probing, question answering, translation, and text classification. However, the handcrafted prompts need lots of human effort and may fail to be optimal (Jiang et al., 2020). Many work focus on achieving prompts automatically, which can be divided into *discrete* (Davison et al., 2019; Jiang et al., 2020; Haviv et al., 2021; Shin et al., 2020; Gao et al., 2021) and *continuous* (Zhong et al., 2021; Qin & Eisner, 2021; Li & Liang, 2021; Lester et al., 2021; Hambardzumyan et al., 2021; Liu et al., 2021b; Zhang et al., 2021), to avoid these problems. In recent works, Logan IV et al. (2021) proposed *null prompts*, which contain no task-specific prompt, and achieve comparable performance than handcrafted prompts. One parallel work with us is Sanh et al. (2021), which utilizes multi-task training with multiple prompts to test the generalization ability on zero-shot task learning. Unlike previous works that require labeled data to select prompts or post-tune the model parameters, we launch an empirical study to learn the potential and limitations of prompt-based methods under the real zero-shot setting, i.e., without labeled data for prompt selection and tuning.

Answer Engineering Answer engineering selects label words and constructs a verbalizer to make the model perform better. Meng et al. (2020) use unlabeled data to search candidate label words. Jiang et al. (2020) paraphrase label words to broaden its space. Schick et al. (2020) use small amounts of training data to compute a word's suitability as label words iteratively. Shin et al. (2020) learn a logistic classifier to select label word. Gao et al. (2021) first use PLMs to select the candidate label words and then extract the final label words based on their zero-shot performance. Hambardzumyan et al. (2021); Zhang et al. (2021) learn a continuous label words. Hu et al. (2021) use external knowledge to expand the label word space of the verbalizer. In our work, we focus on making full of the ability of PLMs without using any extra knowledge and labeled/unlabeled data.

However, all these works need unlabeled data, which may not be available in many realistic scenarios. In this work, we focus on the zero-shot potential of the prompting paradigm. Different from previous work, we do not utilize unlabeled data and only introduce limited costs for label names.

7 Conclusion

This paper thoroughly studies the effectiveness of powerful few-shot learning methods on pre-trained language models from the BERT family in the zero-shot setting. We surprisingly find that simply introducing a few prompt [MASK]s could significantly improve the performance and robustness of the Null Prompt method and even exceed cherry-piked manual prompts, which shows the zero-shot potential of BERT family. We also launched a coarse-to-fine study to learn the influence of multiple components in our proposed method. In the end, we briefly discuss the possible limitations of current zero-shot methods. In the near future, we will explore more under-performed tasks.

REFERENCES

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Joe Davison, Joshua Feldman, and Alexander M Rush. Commonsense knowledge mining from pretrained models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1173–1178, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. In *Association for Computational Linguistics (ACL)*, 2021.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. WARP: Word-level Adversarial ReProgramming. 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)*, pp. 4921–4933, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.381. URL https://aclanthology.org/2021.acl-long.381.

Adi Haviv, Jonathan Berant, and Amir Globerson. Bertese: Learning to speak to bert. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 3618–3623, 2021.

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Juanzi Li, and Maosong Sun. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. *arXiv preprint arXiv:2108.02035*, 2021.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*, 2019.

Teven Le Scao and Alexander Rush. How many data points is a prompt worth? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2627–2636, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.208. URL https://aclanthology.org/2021.naacl-main.208.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195, 2015.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–7880, 2020.

Quentin Lhoest, Albert Villanova del Moral, Patrick von Platen, Thomas Wolf, Yacine Jernite, Abhishek Thakur, Lewis Tunstall, Suraj Patil, Mariama Drame, Julien Chaumond, Julien Plu, Joe Davison, Simon Brandeis, Teven Le Scao, Victor Sanh, Kevin Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Steven Liu, Sylvain Lesage, Lysandre Debut, Théo Matussière, Clément Delangue, and Stas Bekman. huggingface/datasets: 1.11.0, July 2021. URL https://doi.org/10.5281/zenodo.5148649.

Xiang Lisa Li and Percy Liang. 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)*, pp. 4582–4597, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021a.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *arXiv preprint arXiv:2103.10385*, 2021b.

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

Robert L Logan IV, Ivana Balažević, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. Cutting down on prompts and parameters: Simple few-shot learning with language models. *arXiv preprint arXiv:2106.13353*, 2021.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142–150, 2011.

Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pp. 165–172, 2013.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. Text classification using label names only: A language model self-training approach. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9006–9017, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.724. URL https://aclanthology.org/2020.emnlp-main.724.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. True few-shot learning with language models. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=ShnM-rRh4T.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2463–2473, 2019.

Guanghui Qin and Jason Eisner. Learning how to ask: Querying lms with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5203–5212, 2021.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. a.

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

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

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

Timo Schick and Hinrich Schütze. 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*, pp. 255–269, 2021a.

Timo Schick and Hinrich Schütze. 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*, pp. 2339–2352, 2021b.

Timo Schick, Helmut Schmid, and Hinrich Schütze. Automatically identifying words that can serve as labels for few-shot text classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 5569–5578, 2020.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Eliciting knowledge from language models using automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4222–4235, 2020.

Yi Sun, Yu Zheng, Chao Hao, and Hangping Qiu. Nsp-bert: A prompt-based zero-shot learner through an original pre-training task–next sentence prediction. *arXiv preprint arXiv:2109.03564*, 2021.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Wenpeng Yin, Jamaal Hay, and Dan Roth. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3914–3923, 2019.

Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. Differentiable prompt makes pre-trained language models better few-shot learners. *arXiv preprint arXiv:2108.13161*, 2021.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28:649–657, 2015.

Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5017–5033, 2021.

A APPENDIX

A.1 DETAILED EXPERIMENT SETTINGS

Table 7: The label names for IMDB, Amazon, AG News, DBPedia.

Task	Label Names
IMDB	negative,positive
Amazon	negative,positive
AG News	politics,sports,business,technology
DBPedia	company,school,artist,athlete, politics,transportation,building, river,village,animal,plant,album, film,book

Table 8: The templates of Manual Prompt (Prior).

Task	Templates	Accuracy
IMDB	It was [MASK] . [X]	87.32
	[X] All in all, it was [MASK].	85.95
	[X] | In summary, the film is [MASK].	93.68
Amazon	It was [MASK] . [X]	83.02
	[X] All in all, it was [MASK].	70.86
	[X] | In summary, it is [MASK].	92.98
AG News	[MASK] - [X]	58.25
	[MASK] News: [X]	64.04
	[Category: [MASK]] [X]	76.97
DBPedia	[MASK] - [X]	71.12
	[MASK] News: [X]	52.45
	[Category: [MASK]] [X]	84.18

Table 9: The templates of Manual Prompt (w/o Engineering).

Task	Templates	Accuracy
IMDB	[X] Overall, my attitude is [MASK].	82.51
	[X] This text shows [MASK] sentiment .	59.04
	[X] The attitude of this text is [MASK].	80.62
Amazon	[X] Overall, my attitude is [MASK].	76.02
	[X] This text shows [MASK] sentiment .	63.53
	[X] The attitude of this text is [MASK].	87.06
AG News	The next topic is [MASK] .[X]	63.30
	The following is about [MASK] . [X]	65.45
	[X] The topic of this text is [MASK] .	74.20
DBPedia	The next topic is [MASK] .[X]	60.89
	The following is about [MASK] . [X]	73.58
	[X] The topic of this text is [MASK] .	35.79

Table 7 reports the label names for IMDB, Amazon, AG News, DBPedia. Table 8 and Table 9 show the templates for IMDB, Amazon, AG News, DBPedia, which are modified from Schick & Schütze (2021b), and theirs accuracy of Manual Prompt (Prior) and Manual Prompt (w/o Engineering), respectively. Table 10 shows the templates we used for NSP-BERT, which is modified from Yin et al. (2019), and their accuracy. Table 11 shows the shows the templates we used for GLUE and SuperGLUE, which is introduced from Gao et al. (2021) and Schick & Schütze (2021b). For SST-2, MNLI, QNLI, RTE, we report accuracy; for MRPC and QQP, we report F1; for CoLA, we report Matthew's correlation; for STS-B, we report Pearson's correlation.

Table 10: The templates of NSP-BERT.

Task	Templates	Accuracy
IMDB	Overall, my attitude is *label name*.	71.56
	This text shows *label name* sentiment.	73.75
	The attitude of this text is *label name*.	73.16
Amazon	Overall, my attitude is *label name*.	68.17
	This text shows *label name* sentiment.	74.49
	The attitude of this text is *label name*.	75.39
AG News	This text is about *label name* .	77.28
	The topic of this text is *label name*.	76.82
	News: *label name*	78.07
DBPedia	This text is about *label name* .	65.66
	The topic of this text is *label name*.	58.91
	News: *label name*	69.39

Table 11: The templates for GLUE and SuperGLUE.

Task	Template	Label Words
SST-2	$[X]$ It was [MASK].	positive,negative
CoLA	$[X]$ This is [MASK].	correct,incorrect
MNLI	$[X_1]$? [MASK], $[X_2]$.	Yes,Maybe,No
MRPC	$[X_1]$ [MASK], $[X_2]$.	Yes,No
QNLI	$[X_1]$? [MASK], $[X_2]$.	Yes,No
QQP	$[X_1]$ [MASK], $[X_2]$.	Yes,No
RTE	$[X_1]$? [MASK], $[X_2]$.	Yes,No
STS-B	$[X_1]$ [MASK], $[X_2]$.	Yes,No
BoolQ	$[P]$. Question: $[Q]$? Answer: [MASK]	Yes,No
CB	$[H]$? ‖ [MASK], $[P]$	Yes,Maybe,No
MultiRC	$[P]$. Question: $[Q]$? Is it $[A]$? [MASK].	Yes,No
WiC	$[X_1]$ $[X_2]$ Does $[W]$ have the same meaning in both sentences? [MASK]	Yes,No

A.2 Manual Prompt + Multi-Null Prompt Strategy

Table 12: The performance of Manual Prompt (Prior) with multiple mask tokens.

Task	Templates	[MASK] number							
		1	2	3	4	5	6	7	8
IMDB	It was [MASK] . $[X]$	87.32	87.49	88.67	88.50	88.14	87.28	86.61	85.95
	$[X]$ All in all, it was [MASK].	85.95	89.05	91.22	91.63	91.48	91.19	90.98	90.63
	$[X]$ | In summary, the film is [MASK].	93.68	93.87	94.00	93.80	93.32	92.88	92.40	92.15
Amazon	It was [MASK] . $[X]$	83.02	82.81	84.40	84.39	84.10	83.02	81.97	80.87
	$[X]$ All in all, it was [MASK].	70.86	73.85	75.63	74.89	73.94	72.62	71.26	69.53
	$[X]$ | In summary, it is [MASK].	92.98	93.29	93.50	92.73	92.12	91.49	90.56	89.42
AG News	[MASK] - $[X]$	58.25	59.92	58.33	59.03	61.20	61.84	62.64	63.07
	[MASK] News: $[X]$	64.04	63.84	61.68	62.83	64.24	64.72	64.79	64.92
	[Category: [MASK]] $[X]$	76.97	75.43	78.62	79.43	79.72	79.72	79.99	79.95
DBPedia	[MASK] - $[X]$	71.12	59.57	52.49	48.48	46.97	47.57	49.08	50.33
	[MASK] News: $[X]$	52.45	54.26	54.56	54.75	55.98	58.19	60.34	61.92
	[Category: [MASK]] $[X]$	84.18	83.08	82.34	82.31	82.37	81.61	80.89	80.32

Our proposed simple Multi-Null Prompt strategy can be effective for zero-shot text classifications, and the core spirit is to use multiple [MASK] tokens. Therefore, to fully explore the potential of the strategy, we extend the verification to more prompting strategies. That is, we combine the previous effective prompt methods with Multi-Null Prompt on the zero-shot setting to investigate its impact. Specifically, we conduct experiments on our Multi-Null Prompt with (1) Manual Prompt (Prior), and (2) KPT (Hu et al., 2021) method on text classifications. The corresponding results are shown

Table 13: The performance of KPT with multiple mask tokens.

Task	Templates	[MASK] number							
		1	2	3	4	5	6	7	8
IMDB	It was [MASK] . [X]	89.47	89.90	89.43	88.81	88.10	87.36	86.37	85.67
	Just [MASK] ! [X]	88.01	88.58	87.10	87.42	87.39	87.33	87.48	87.60
	[X] All in all, it was [MASK].	88.94	92.02	94.03	93.71	92.97	92.12	90.98	89.99
	[X] In summary, the film was [MASK].	93.60	93.58	93.48	93.04	92.83	92.46	92.05	91.53
Amazon	It was [MASK] . [X]	86.92	86.76	87.80	85.67	83.39	81.31	79.81	78.75
	Just [MASK] ! [X]	91.22	90.78	90.06	89.47	88.99	88.54	88.11	87.66
	[X] All in all, it was [MASK].	75.18	77.23	83.78	81.43	78.02	75.08	72.90	71.11
	[X] In summary, it was [MASK].	92.15	90.62	91.82	90.99	89.69	88.02	86.17	84.27
AG News	A [MASK] News: [X]	71.79	72.28	71.99	72.51	73.03	72.91	72.97	72.74
	[X] This topic is about [MASK].	66.68	77.87	74.46	73.61	73.46	71.98	71.32	71.54
	[Category: [MASK]] [X]	79.09	77.45	76.67	75.79	75.55	75.42	75.68	75.83
	[Topic: [MASK]] [X]	76.86	78.30	76.95	75.51	74.75	74.47	74.33	74.37
DBPedia	[T] [P] [T] is a [MASK].	75.36	76.00	73.88	72.74	71.54	70.27	69.26	68.55
	[T] [P] In this sentence, [T] is a [MASK].	77.15	77.91	80.96	78.96	77.16	76.48	76.20	75.87
	[T] [P] The type of [T] is [MASK].	68.74	75.52	72.01	68.59	66.37	65.45	65.35	65.71
	[T] [P] The category of [T] is [MASK].	72.95	72.59	71.35	69.76	68.77	69.01	70.35	71.45

in Table 12 and Table 13. We can clearly observe that the performance for each task is further significantly improved. For example, in Table 12, the accuracy on Amazon can be 93.50, which is much better than 82.29 of Manual Prompt (Prior) and 86.22 of Multi-Null Prompt in Table 1. The results clearly demonstrate the effectiveness of the combination of our multiple mask strategy and the other prompt strategies.

A.3 Examples of Different Prompt Strategies

To better clearly compare the different prompt strategies used in this work, we put several examples of these strategies in Table 14, which are [CLS], Manual Prompt, Null Prompt, our Multi-Null Prompt, and NSP-BERT methods. The differences can be found in these examples. From these examples, we can also see that our simple Multi-Null Prompt is complementary with others and can be easily combined with other strategies, which are also verified in the above study.

Table 14: Examples of Different Prompt Strategies.

Strategies	Examples
CLS	[CLS] This food is delicious! [SEP]
Manual Prompt	[CLS] This food is delicious! The sentiment of this sentence is [MASK] [SEP]
Null Prompt	[CLS] This food is delicious! [MASK] [SEP]
Multi-Null Prompt	[CLS] [MASK] This food is delicious! [MASK] [SEP]
NSP-BERT	[CLS] This food is delicious! [SEP] I am happy. [EOS]