Prompting for Few-shot Learning

Edward Tian and Kaixuan Huang
Motivation and Related Work

Prompt based fine-tuning

How many data points is a prompt worth? (Le Scao et al., 2021)

Adapting a pre-trained language model

LM-BFF (Gao et al., 2021)

Conclusion, Results and Discussion
What is a good prompt?

GPT3: “A good prompt is one that is general enough to be used for a variety of tasks, but specific enough to be helpful for a particular task”
What makes a good prompt? for an NLP task,

GPT3: “a good prompt is one that is specific and provides enough context for the model to be able to generate a response that is relevant to the task.”
Large Language Models are Few-shot Learners (Brown, et al.)

- GPT-3 huge motivator for prompting
- Earliest work in prompts traces back to GPT-1/2 (Radford et al., 2018, 2019)
- With good prompts, LMs can achieve decent zero-shot performance on tasks from sentiment classification to reading comprehension
Can we make “smaller” LMs also work with few examples?

GPT
Natural language prompts, gigantic model, few in-context examples, no-parameters updated

BERT
110M Parameters
1000x smaller than GPT3
Generic [CLS] Token
Fine-tuned to 2.5 to 400k examples for GLUE tasks
How to adapt a pre-trained Language model?

Head-based fine-tuning

How to adapt a pre-trained Language model?

Prompt-based fine-tuning

# Head-based v.s. Prompt-based Fine-tuning

<table>
<thead>
<tr>
<th></th>
<th>Head-based</th>
<th>Prompt-based</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New parameters?</strong></td>
<td>Yes. hidden_size * num_classes</td>
<td>No</td>
</tr>
<tr>
<td><strong>Few-shot friendly?</strong></td>
<td>😓</td>
<td>😊</td>
</tr>
</tbody>
</table>
Prompt-based Fine-tuning (Classification task)

Input: \( X_1 = \text{No reason to watch.} \)

Step 1. Formulate the downstream task into a (Masked) LM problem using a template:

\[
[\text{CLS}] \ \text{No reason to watch}. \ \text{It was} [\text{MASK}]. \ \text{[SEP]}
\]
Prompt-based Fine-tuning (Classification task)

Input: \( \mathcal{X}_1 = \text{No reason to watch.} \)

**Step 1.** Formulate the downstream task into a (Masked) LM problem using a template:

\[
[\text{CLS}] \text{No reason to watch . It was [MASK]. [SEP]}
\]

**Step 2.** Choose a label word mapping \( \mathcal{M} \), which maps task labels to individual words.

\[
\text{great (label: positive)} \quad \text{terrible (label: negative)✓}
\]

Prompt-based Fine-tuning (Classification Task)

**Step 3.** Fine-tune the LM to fill in the correct label word.

\[
p(y \mid x_{in}) = p([\text{MASK}] = \mathcal{M}(y) \mid x_{\text{prompt}}) = \frac{\exp (w_{\mathcal{M}(y)} \cdot h_{[\text{MASK}]})}{\sum_{y' \in \mathcal{Y}} \exp (w_{\mathcal{M}(y')} \cdot h_{[\text{MASK}]})},
\]

---

Prompt-based Fine-tuning (Regression Task)

Regression: interpolating between two extremes

\[ y = v_l \cdot p(y_l \mid x_{in}) + v_u \cdot p(y_u \mid x_{in}) \]

Predicted \( y = 0.2 \)

Prompt-based Fine-tuning (Regression Task)

Regression: interpolating between two extremes

\[ y = v_l \cdot p(y_l \mid x_{in}) + v_u \cdot p(y_u \mid x_{in}) \]

The LM is fine-tuned to minimize the KL-divergence between the inferred \( p(y_u \mid x_{in}) \) and the observed mixture weight \( \frac{(y-v_l)}{(v_u-v_l)} \)

Q1. How does prompt-based fine-tuning work and why does it outperform head-based fine-tuning (as the method described in BERT) in low-data regimes?

A1. Prompt-based fine-tuning involves:
- a **template** which turns the downstream task into a (masked) language modelling problem, and
- a set of **label words** that map the textual output of the LM to the classification labels.

In this way, we don't need to introduce any new parameters so all the pre-trained parameters can be fine-tuned more sample-efficiently.

It outperforms head-based fine-tuning in low-data regimes since BERT introduces new randomly-initialized parameters (often more than 1k), which are hard to learn well from only a few examples.
Making Pre-trained Language Models Better
Few-shot Learners

Tianyu Gao, Adam Fisch, Danqi Chen
## Datasets

| Category      | Dataset | $|Y|$ | Type          | Labels (classification tasks)                               |
|---------------|---------|-----|---------------|--------------------------------------------------------------|
| single-sentence | SST-2   | 2   | sentiment     | positive, negative                                           |
|               | SST-5   | 5   | sentiment     | v. pos., positive, neutral, negative, v. neg.                |
|               | MR      | 2   | sentiment     | positive, negative                                           |
|               | CR      | 2   | sentiment     | positive, negative                                           |
|               | MPQA    | 2   | opinion polarity | positive, negative                                      |
|               | Subj    | 2   | subjectivity  | subjective, objective                                       |
|               | TREC    | 6   | question cls. | abbr., entity, description, human, loc., num.                |
|               | CoLA    | 2   | acceptability | grammatical, not_grammatical                                 |
| sentence-pair | MNLI    | 3   | NLI           | entailment, neutral, contradiction                           |
|               | SNLI    | 3   | NLI           | entailment, neutral, contradiction                           |
|               | QNLI    | 2   | NLI           | entailment, not_entailment                                   |
|               | RTE     | 2   | NLI           | entailment, not_entailment                                   |
|               | MRPC    | 2   | paraphrase    | equivalent, not_equivalent                                  |
|               | QQP     | 2   | paraphrase    | equivalent, not_equivalent                                  |
|               | STS-B   | $\mathcal{R}$ | sent. similarity | -                                                              |

- Most tasks: # Labels $\leq 3$

# Datasets

| Category         | Dataset | $|Y|$ | Type             | Labels (classification tasks)                                                                 |
|------------------|---------|-----|------------------|------------------------------------------------------------------------------------------------|
| single-sentence  | SST-2   | 2   | sentiment        | positive, negative                                                                             |
|                  | SST-5   | 5   | sentiment        | v. pos., positive, neutral, negative, v. neg.                                                  |
|                  | MR      | 2   | sentiment        | positive, negative                                                                             |
|                  | CR      | 2   | sentiment        | positive, negative                                                                             |
|                  | MPQA    | 2   | opinion polarity | positive, negative                                                                             |
|                  | Subj    | 2   | subjectivity     | subjective, objective                                                                          |
|                  | TREC    | 6   | question cls.    | abbr., entity, description, human, loc., num.                                                   |
|                  | CoLA    | 2   | acceptability    | grammatical, not_grammatical                                                                    |
| sentence-pair    | MNLI    | 3   | NLI              | entailment, neutral, contradiction                                                              |
|                  | SNLI    | 3   | NLI              | entailment, neutral, contradiction                                                              |
|                  | QNLI    | 2   | NLI              | entailment, not_entailment                                                                     |
|                  | RTE     | 2   | NLI              | entailment, not_entailment                                                                     |
|                  | MRPC    | 2   | paraphrase       | equivalent, not_equivalent                                                                     |
|                  | QQP     | 2   | paraphrase       | equivalent, not_equivalent                                                                     |
|                  | STS-B   | $\mathcal{R}$ | sent. similarity | -                                                                                               |

- Most tasks: # Labels $\leq 3$
- SST-5, TREC have 5 or 6 labels

### Datasets

| Category       | Dataset | | Type          | Labels (classification tasks) |
|----------------|---------|------------------|------------------------------|
| Single-sentence| SST-2   | 2                | sentiment                    | positive, negative          |
|                | SST-5   | 5                | sentiment                    | v. pos., positive, neutral, negative, v. neg. |
|                | MR      | 2                | sentiment                    | positive, negative          |
|                | CR      | 2                | sentiment                    | positive, negative          |
|                | MPQA    | 2                | opinion polarity            | positive, negative          |
|                | Subj    | 2                | subjectivity                 | subjective, objective       |
|                | TREC    | 6                | question cls.               | abbr., entity, description, human, loc., num. |
|                | CoLA    | 2                | acceptability                | grammatical, not_grammatical |
| Sentence-pair  | MNLI    | 3                | NLI                          | entailment, neutral, contradiction |
|                | SNLI    | 3                | NLI                          | entailment, neutral, contradiction |
|                | QNLI    | 2                | NLI                          | entailment, not_entailment  |
|                | RTE     | 2                | NLI                          | entailment, not_entailment  |
|                | MRPC    | 2                | paraphrase                   | equivalent, not_equivalent  |
|                | QQP     | 2                | paraphrase                   | equivalent, not_equivalent  |
|                | STS-B   | itespace       | sent. similarity            | -                            |

- Most tasks: # Labels <= 3
- SST-5, TREC have 5 or 6 labels
- STS-B is a regression task

Examples

SST-2: sentiment analysis.
- E.g. S1 = “The movie is ridiculous”. **Label**: negative.
- Manual prompt:

<table>
<thead>
<tr>
<th>Template</th>
<th>Label words</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;S₁&gt; It was [MASK].</td>
<td>great/terrible</td>
</tr>
</tbody>
</table>
Examples

SNLI: Natural Language Inference

- **S1** = “A soccer game with multiple males playing”. **S2** = “Some men are playing sport”. **Label**: Entailment.
- Manual prompt:

<table>
<thead>
<tr>
<th>Template</th>
<th>Label words</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt;S_1&gt; , [M\text{ASK}] , &lt;S_2&gt;)</td>
<td>Yes/Maybe/No</td>
</tr>
</tbody>
</table>
Few-shot Learning & Evaluation Protocol

Q2. How does (Gao et al., 2021) conduct evaluations in few-shot settings?

- Training dataset: \( K=16 \) examples per class.
- Dev dataset: same size as training dataset.

Performance measured across 5 random splits of \{train, dev\} set.
What is a “True” Few-shot Learning setting?

- Perez et al. (2021): “Tuned few-shot learning algorithms should be compared against data-rich supervised learning algorithms that use the same amount of total data \(|D_{\text{train}}| + |D_{\text{val}}|\)”
- Larger dev set leads to better performance.

<table>
<thead>
<tr>
<th>Fine-tuning</th>
<th>SST-2</th>
<th>SNLI</th>
<th>TREC</th>
<th>MRPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>No (D_{\text{dev}})</td>
<td>79.5</td>
<td>49.2</td>
<td>83.9</td>
<td>77.8</td>
</tr>
<tr>
<td>(</td>
<td>D_{\text{dev}}</td>
<td>=</td>
<td>D_{\text{train}}</td>
<td>)</td>
</tr>
<tr>
<td>(</td>
<td>D_{\text{dev}}</td>
<td>= 10</td>
<td>D_{\text{train}}</td>
<td>)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prompt-based FT</th>
<th>SST-2</th>
<th>SNLI</th>
<th>TREC</th>
<th>MRPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>No (D_{\text{dev}})</td>
<td>92.1</td>
<td>75.3</td>
<td>84.8</td>
<td>70.2</td>
</tr>
<tr>
<td>(</td>
<td>D_{\text{dev}}</td>
<td>=</td>
<td>D_{\text{train}}</td>
<td>)</td>
</tr>
<tr>
<td>(</td>
<td>D_{\text{dev}}</td>
<td>= 10</td>
<td>D_{\text{train}}</td>
<td>)</td>
</tr>
</tbody>
</table>


Same setting as PET (Schick and Schütze, 2021a,b)
Q2: Is it still true few-shot learning if we manually tune the prompt?

A2: It is still "true" few-shot learning, because the whole training process, including hyper-parameter/prompt tuning, still only involves a few examples, which is the training dataset plus the development dataset.
How important is a good prompt for few-shot learning?

Label words match the semantic classes → good final accuracy

<table>
<thead>
<tr>
<th>Template</th>
<th>Label words</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SST-2 (positive/negative)</td>
<td>mean (std)</td>
<td></td>
</tr>
<tr>
<td>&lt;S₁&gt; It was [MASK].</td>
<td>great/terrible</td>
<td>92.7 (0.9)</td>
</tr>
<tr>
<td>&lt;S₁&gt; It was [MASK].</td>
<td>good/bad</td>
<td>92.5 (1.0)</td>
</tr>
<tr>
<td>&lt;S₁&gt; It was [MASK].</td>
<td>cat/dog</td>
<td>91.5 (1.4)</td>
</tr>
<tr>
<td>&lt;S₁&gt; It was [MASK].</td>
<td>dog/cat</td>
<td>86.2 (5.4)</td>
</tr>
<tr>
<td>&lt;S₁&gt; It was [MASK].</td>
<td>terrible/great</td>
<td>83.2 (6.9)</td>
</tr>
<tr>
<td>Fine-tuning</td>
<td>-</td>
<td>81.4 (3.8)</td>
</tr>
</tbody>
</table>

Experiments are done with $K=16$ examples per class.

How important is a good prompt for few-shot learning?

A small change in the template can make a huge difference (>10% performance drop)

<table>
<thead>
<tr>
<th>Template</th>
<th>Label words</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNLI (entailment/neutral/contradiction)</td>
<td>mean (std)</td>
<td></td>
</tr>
<tr>
<td>$&lt;S_1&gt;$ ? [MASK] , $&lt;S_2&gt;$</td>
<td>Yes/Maybe/No</td>
<td>77.2 (3.7)</td>
</tr>
<tr>
<td>$&lt;S_1&gt;$ . [MASK] , $&lt;S_2&gt;$</td>
<td>Yes/Maybe/No</td>
<td>76.2 (3.3)</td>
</tr>
<tr>
<td>$&lt;S_1&gt;$ ? [MASK] $&lt;S_2&gt;$</td>
<td>Yes/Maybe/No</td>
<td>74.9 (3.0)</td>
</tr>
<tr>
<td>$&lt;S_1&gt;$ $&lt;S_2&gt;$ [MASK]</td>
<td>Yes/Maybe/No</td>
<td>65.8 (2.4)</td>
</tr>
<tr>
<td>$&lt;S_2&gt;$ ? [MASK] , $&lt;S_1&gt;$</td>
<td>Yes/Maybe/No</td>
<td>62.9 (4.1)</td>
</tr>
<tr>
<td>$&lt;S_1&gt;$ ? [MASK] , $&lt;S_2&gt;$</td>
<td>Maybe/No/Yes</td>
<td>60.6 (4.8)</td>
</tr>
<tr>
<td>Fine-tuning</td>
<td>-</td>
<td>48.4 (4.8)</td>
</tr>
</tbody>
</table>

Experiments are done with $K=16$ examples per class.

Several slides on LMBFF adapted from Tianyu Gao’s conference presentation.
How do we design a good prompt?

**BoolQ**: given a passage $q$ and question $p$, design a prompt for question answering

For **BoolQ**, given a passage $p$ and question $q$:


- $p$. Based on the previous passage, $q$? $<$MASK$>$.

- Based on the following passage, $q$? $<$MASK$>$. $p$

with "yes" or "no" as verbalizers for True and False.
How do we design a good prompt?

**WiC**: given two sentences $S_1$ and $S_2$, and a word $W$, design a prompt to determine whether $W$ was used in the same sense in both sentences.

For WiC, given two sentences $s_1$ and $s_2$ and a word $w$, we classify whether $w$ was used in the same sense.

"$s_1$" / "$s_2$". Similar sense of “$w$”? <MASK>.

$s_1$ $s_2$ Does $w$ have the same meaning in both sentences? <MASK>. 


How do we design good prompts?

- Difficult problem, manually designed in previous works (Schick and Schutze, 2021 a,b)
- Requires domain expertise and trial and error
- Challenge to construct prompt $P$ find a template $T$ and label words $M(y)$ that work in conjunction
- Low number of examples -> overfitting
Recall …

*Slight variations in prompts between terrible/great leads to sizable differences!*

Automatic Prompt Generation

Automatic Label Word Search → Automatic Template Search → Automatic Prompt Search

* In experiments assume access to a few-shot training and development set with 16 samples per class.
For a classification task, for each label, construct a set of top-k words with highest MLM probabilities conditioned on all training examples.

Given the manual template: 

\[<S> \text{It was [MASK].}\]

- \text{label: positive}
  - good
  - great
  - perfect
  - ...

- \text{label: negative}
  - awful
  - bad
  - terrible
  - ...
Automatic Label Search

Enumerate all combinations. Prune by Zero-shot results on training set.
Automatic Label Search

- Construct Candidate Sets
- Enumerate Combinations and Prune
- Re-Rank by Finetuning

**Finetune** all top \( n \) assignments and re-rank to find the best ones using development dataset.

Given the **manual** template: <S> It was [MASK].

- good/bad
- great/bad
- great/terrible
- perfect/terrible

**Fine-tune and evaluate on** \( \mathcal{D}_{dev} \)
- good/bad (85%)
- great/bad (82%)
- great/terrible (91%)
- perfect/terrible (86%)
Intuition

Mask the prompts and ask T5 🚀 to ___ in the blanks
Automatic Template Search

Heuristic

1. Use T5 to generate candidates.
2. Re-rank them based on performance on development set after fine-tuning.
Automatic Template Search

- Mask Tokens for T5
- Autoregressive Decoding
- Repeat until Stopword
- Beam Search

Training examples for label: positive

A fun ride. <X> great <Y>
A pleasure to watch. <X> great <Y>
...

Training examples for label: negative

No reason to watch. <X> terrible <Y>
This junk. <X> terrible <Y>
...

positive: great, negative: terrible
Label mapping $\mathcal{M}(\mathcal{Y})$
Automatic Template Search

Mask Tokens for T5

Autoregressive Decoding

Repeat until Stopword

Beam Search

At each position in auto-regressive decoding, we condition the model on all training examples.

$$\sum_{(x_{in}, y) \in D_{train}} \log P_{T5}(t_j | t_1, ..., t_{j-1}, T_k(x_{in}, y))$$
Automatic Template Search

At each position in auto-regressive decoding, we condition the model on all training examples.

$$\sum_{(x_{in}, y) \in D_{train}} \log P_{T5}(t_j \mid t_1, \ldots, t_{j-1}, T_g(x_{in}, y))$$
Automatic Template Search

Apply **beam search** with large width ~100 to generate many templates to evaluate.
Demonstrations

GPT3 In-context Learning: Randomly Samples Examples and fills them in context.

[CLS] No reason to watch. It was [MASK]. [SEP]

Prompt-based fine-tuning
Demonstrations

Improved: Selective Sampling, ie. for this example sample from the positive class 😎

Prompt-based fine-tuning with demonstrations
Demonstrations

And we can also sample one from a negative training instance

Prompt-based fine-tuning with demonstrations
Intuition

Selective apply demonstrations that are semantically close to the input for optimal results
Demonstrations Sampling

Heuristic

1. Measure cosine similarity between all training examples and input.
2. Use pre-trained sentence encoder BERT to measure similarities
3. Only use top 50% of examples as demonstration candidates
Demonstrations Example

Input
No reason to watch.

Similarity

Examples for label: positive
- 0.83 The movie is really great.
- 0.21 Food is delicious.
- ...

Examples for label: negative
- 0.92 I don't like watching the show.
- 0.37 I don't like the food here.
- ...
Recall: Experiment Setup

16 Experiments, 8 Single-Sentence and 7 Sentence-pair tasks

For each experiment, paper used 16 samples per class for training and development sets

Sample 5 fewshot sets for each dataset and averaged the results to address variance

```python
python run.py 
   --task_name SST-2 
   --data_dir data/k-shot/SST-2/16-42 
   --overwrite_output_dir 
   --do_train 
   --do_eval 
   --do_predict 
   --evaluate_during_training 
   --model_name_or_path roberta-large 
   --few_shot_type prompt-demo 
   --num_k 16 
   --max_steps 1000 
   --eval_steps 100 
   --per_device_train_batch_size 2 
   --learning_rate 1e-5 
   --num_train_epochs 0 
   --output_dir result/tmp 
   --seed 42 
   --template "*cls*sent_0* It was*mask*.*sep*" 
   --mapping "{'0': 'terrible', '1': 'great'}" 
   --num_sample 16 
```

Figure: Example of running an output example source from github.com/princeton-nlp/LM-BFF
Results (single prompts)

<table>
<thead>
<tr>
<th></th>
<th>Fine-tuning</th>
<th>Prompt-based fine-tuning</th>
<th>+ Automatic templates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Sentence</td>
<td>70.4</td>
<td>73.4</td>
<td>74.4</td>
</tr>
<tr>
<td>Sentence Pair</td>
<td>55.9</td>
<td>70.1</td>
<td>72.0</td>
</tr>
</tbody>
</table>

Average accuracy (%)

Tables on LMBFF results adapted from Tianyu Gao’s conference presentation.
Results (single prompts)

- **Single Sentence**
  - Fine-tuning: 70.4%
  - Prompt-based fine-tuning: 73.4%
  - Prompt-based fine-tuning + Automatic templates: 74.4%
  - Prompt-based fine-tuning + Demonstrations: 76.3%

- **Sentence Pair**
  - Fine-tuning: 55.9%
  - Prompt-based fine-tuning: 70.1%
  - Prompt-based fine-tuning + Automatic templates: 72.0%
  - Prompt-based fine-tuning + Demonstrations: 72.6%
Results (single prompts)

![Bar chart showing accuracy improvements for different tasks and methods.](chart.png)

- Fine-tuning
- "GPT-3" in-context
- Prompt-based fine-tuning
- LM-BFF

Accuracy (%)

<table>
<thead>
<tr>
<th>Task</th>
<th>Fine-tuning</th>
<th>&quot;GPT-3&quot; in-context</th>
<th>Prompt-based fine-tuning</th>
<th>LM-BFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>SST-2</td>
<td>+0.3%</td>
<td>+2.1%</td>
<td>+0.7%</td>
<td>+4.6%</td>
</tr>
<tr>
<td>SST-5</td>
<td></td>
<td></td>
<td>+0.7%</td>
<td>+4.6%</td>
</tr>
<tr>
<td>MR</td>
<td></td>
<td></td>
<td>+1.8%</td>
<td></td>
</tr>
<tr>
<td>CR</td>
<td></td>
<td></td>
<td>+0.2%</td>
<td></td>
</tr>
<tr>
<td>MPQA</td>
<td></td>
<td></td>
<td>+12.5%</td>
<td></td>
</tr>
<tr>
<td>Subj</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TREC</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CoLA (Matt.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results (single prompts)

Accuracy (%)

- Fine-tuning
- "GPT-3" in-context
- Prompt-based fine-tuning
- LM-BFF

For each task:
- MNLI
- MNLI-mm
- SNLI
- QNLI
- RTE
- MRPC (F1)
- QQP (F1)
- STS-B (Pear.)
Results (ensemble)

<table>
<thead>
<tr>
<th></th>
<th>Single manual prompt</th>
<th>PET's prompts ensemble</th>
<th>LM-BFF ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNLI</td>
<td>68.3</td>
<td>71.9</td>
<td>74.0 (4 prompts)</td>
</tr>
<tr>
<td>RTE</td>
<td>69.1 (5 prompts)</td>
<td>69.2</td>
<td>71.9 (5 prompts)</td>
</tr>
</tbody>
</table>

Accuracy (%)
Results (ensemble)

- **MNLI**
  - 68.3 (4 prompts)
  - 71.9 (4 prompts)
  - 74.0 (4 prompts)
  - 75.4 (20 prompts)

- **RTE**
  - 69.1 (5 prompts)
  - 69.2 (5 prompts)
  - 71.9 (5 prompts)
  - 72.3 (20 prompts)

Legend:
- Single manual prompt
- PET’s prompts ensemble
- LM-BFF ensemble
- LM-BFF (20 templates)
Ablation Study: Automatic Prompt Search

### SST-2 (positive/negative)

- $T(x_{in}) = \langle S_1 \rangle \text{ It was } [\text{MASK}].$
- #1. irresistible/pathetic
- #2. wonderful/bad
- #3. delicious/bad

### SNLI (entailment/neutral/contradiction)

- $T(x_{in}) = \langle S_1 \rangle \text{? } [\text{MASK}] \text{, } \langle S_2 \rangle$
- #1. Alright/Watch/Except
- #2. Hi/Watch/Worse
- #3. Regardless/Fortunately/Unless
Ablation Study: Demonstrations

- Prompt-based fine-tuning
- demo w/ uniform sampling

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SST-2</td>
<td></td>
</tr>
<tr>
<td>SNLI</td>
<td></td>
</tr>
<tr>
<td>TREC</td>
<td></td>
</tr>
<tr>
<td>MRPC</td>
<td></td>
</tr>
</tbody>
</table>

Accuracy (%)
Ablation Study: Demonstrations

- Prompt-based fine-tuning
- demo w/ uniform sampling
- demo w/ selective sampling

Accuracy (%)

- SST-2
- SNLI
- TREC
- MRPC

70 80 90 100
Key Findings

- LMBFF Introduced **automatic search** prompt based fine tuning and a selective way for **incorporating demonstrations**

- Provided few-shot evaluations on 15 tasks. LMBFF **dramatically outperforms** standard fine tuning

- Limitations include large variance and automatic search reliance on manual label words
The benefits of prompts are prominent when $K$ is small.

How Many Data Points is a Prompt Worth?

Teven Le Scao, Alexander M. Rush
Setting

- Compare head-based v.s. Prompt-based fine-tuning
- Model: RoBERTa-large
- Manually-designed prompts

Datasets

7 datasets from SuperGLUE + MNLI.

- Entailment tasks: CB, MNLI, RTE
- Multiple-Choice Question Answering: BoolQ, MultiRC
- Common-sense Reasoning: WSC, COPA, WiC
Prompt-based vs head-based

“Data advantage” @ acc 0.75

Source: How Many Data Points is a Prompt Worth?, Le Scao & Rush, 2021
Prompt-based vs head-based

Source: How Many Data Points is a Prompt Worth?, Le Scao & Rush, 2021
Prompt-based vs head-based

Averaged data advantage over different accuracy levels:

<table>
<thead>
<tr>
<th>MNLI</th>
<th>BoolQ</th>
<th>CB</th>
<th>COPA</th>
<th>MultiRC*</th>
<th>RTE</th>
<th>WiC</th>
<th>WSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>3506 ± 536</td>
<td>752 ± 46</td>
<td>90 ± 2</td>
<td>288 ± 242</td>
<td>384 ± 378</td>
<td>282 ± 34</td>
<td>-424 ± 74</td>
<td>281 ± 137</td>
</tr>
</tbody>
</table>

Source: How Many Data Points is a Prompt Worth?, Le Scao & Rush, 2021
How important is a good prompt?

\[ P = \text{good template} + \text{good label words} \]
\[ N = \text{good template} + \text{non-sensical label words.} \]
\[ \text{[e.g. Mike -> “Positive”, John -> “Negative”]} \]

\[ H = \text{head-based} \]

---

**Source:** How Many Data Points is a Prompt Worth?, Le Scao & Rush, 2021
How important is a good prompt in few-shot setting?

N catches up with P when training points are more than ~300

Source: How Many Data Points is a Prompt Worth?, Le Scao & Rush, 2021
Additional Method for Making Prompts

- Round-trip translation (Jiang et al., 2020)
- Replacement of phrases from thesaurus (Yuan et al, 2021)
- Neural prompt re-writer (Haviv et al, 2021)

Ethical Considerations

“LMs appear to follow yet do not actually follow users’ instructions has important implications, especially considering the increasing commercial use of LMs. While traditional fine-tuned models also pose challenges in interpretability, with prompt-based models, an illusion of instruction following can be more pernicious than having no instructions at all”

Source: Do Prompt-Based Models Really Understand the Meaning of Their Prompts? Webson, et al.
Credits and Special Thanks!

Professor Chen
Alexander Wettig
Tianyu Gao
Q3: We already know that finding a good prompt is so important. Sometimes, it is also challenging to find prompts that are natural and fit in pre-trained distributions. For example, <S1> ? [MASK] , <S2>, the chance that “Maybe” can fill in [MASK] is very low (this is the prompt used for NLI tasks in Gao et al., 2021). Do you have any ideas about how to improve this and find better prompts?
Additional Prompt Engineering Methods (discrete / hard prompts)

**Prompt Mining**
Jiang et al. (2020) uses a mining-based method to automatically find templates given a set of training inputs x and y. Scrapes a large text corpus (e.g. Wikipedia) for strings containing x and y, and finds middle words or dependency paths between the inputs and outputs.

**Prompt Paraphrasing**
Takes an existing prompt, paraphrases into other prompts, and uses the prompt that achieves the best result. Prompt paraphrasing can be done with multiple methods including round-trip translation (Jiang et al., 2020); replacement of phrases from thesaurus (Yuan et al, 2021) and a neural prompt re-writer (Haviv et al, 2021)

**Prompt Generation:**
Gao et al. (2021) introduces pre-trained T5 seq to seq to fill in missing spans and generate template tokens. Ben-David et al. (2021) builds upon this method in introducing a domain adaptation algorithm that trains T5 to generate unique domain relevant features.