Let Me Check the Examples: Enhancing Demonstration Learning via Explicit Imitation

Sirui Wang[1,2]*, **Kaiwen Wei**[2]*†, **Hongzhi Zhang**[2], **Yuntao Li**[2], **Wei Wu**[2]
[1]Department of Automation, Tsinghua University, China
[2]Meituan Inc., Beijing, China
{wangsirui,weikaiwen,zhanghongzhi03}@meituan.com
{liyuntao04,wuwei130}@meituan.com

Abstract

Demonstration learning aims to guide the prompt prediction by providing answered demonstrations in the few shot settings. Despite achieving promising results, existing work only concatenates the answered examples as demonstrations to the prompt template (including the raw context) without any additional operation, neglecting the prompt-demonstration dependencies. Besides, prior research found that randomly replacing the labels of demonstrations *marginally* hurts performance, illustrating that the model could not properly learn the knowledge brought by the demonstrations. Inspired by the human learning process, in this paper, we introduce Imitation DEMOnstration learning (Imitation-Demo) to strengthen demonstration learning via explicitly imitating human review behaviour, which includes: (1) contrastive learning mechanism to concentrate on similar demonstrations.(2) demonstration-label re-prediction method to consolidate known knowledge. Experiment results show that our proposed method achieves state-of-the-art performance on 5 out of 14 classification corpus. Further studies also prove that Imitation-Demo strengthens the associations between the prompt and demonstrations, which could provide the basis for exploring how demonstration learning works.

1 Introduction

Prompt-based learning typically works by modifying the input into cloze-style prompt templates and using the masked language models (MLMs) to complete the unfilled information in probabilistic. It has achieved promising performance in various NLP tasks (Schick and Schütze, 2021; Lester et al., 2021; Hu et al., 2021), especially in low-resource settings (Scao and Rush, 2021). A promising prompt engineering category is *demonstration learning* (Gao et al., 2021; Liu et al., 2021a), which seeks to provide a few answered samples as demonstrations to assist prompt prediction. As shown in Fig. 1 (a), the demonstration learning method concatenates the answered demonstrations per category to the prompt, and seeks to classify the $[MASK]$ token as $great$, indicating a $positive$ prediction result based on a label-to-word mapping.

The intuition of demonstration learning is that samples with similar expressions or content can provide repetitive patterns (Liu et al., 2021a). However, Min et al. (2022) point out that replacing gold demonstration labels with random labels *marginally* hurts performance. This finding is counter-intuitive and illustrates that the model could not comprehensively refer to the knowledge brought by the demonstrations in an implicit way. We attribute this problem to that existing methods simply concatenate the answered demonstrations to the prompt template without any additional operation, ignoring the dependencies between prompt and demonstrations.

To overcome this limitation, we rethink how human beings learn from demonstrations. Intuitively, when faced with a new challenging question, they typically (1) look for the most similar example to the question first, and then (2) reply to the question according to the answering steps of the retrieved example. Humans tend to strengthen the learning process through review strategies, i.e., finding a better solution to select similar examples and re-answering the questions of examples to consolidate known knowledge. Inspired by this, likewise, the interactions between the prompt and demonstrations could also be reinforced by imitating the human reviewing process for demonstration learning.

In this paper, we propose a simple-yet-effective version of demonstration learning, named **Imitation DEMO**nstration Learning (Imitation-Demo) to explicitly strengthen the two sub-steps of demonstration learning via human-like review. Specifi-

*Equal contribution.
†Corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 1080–1088

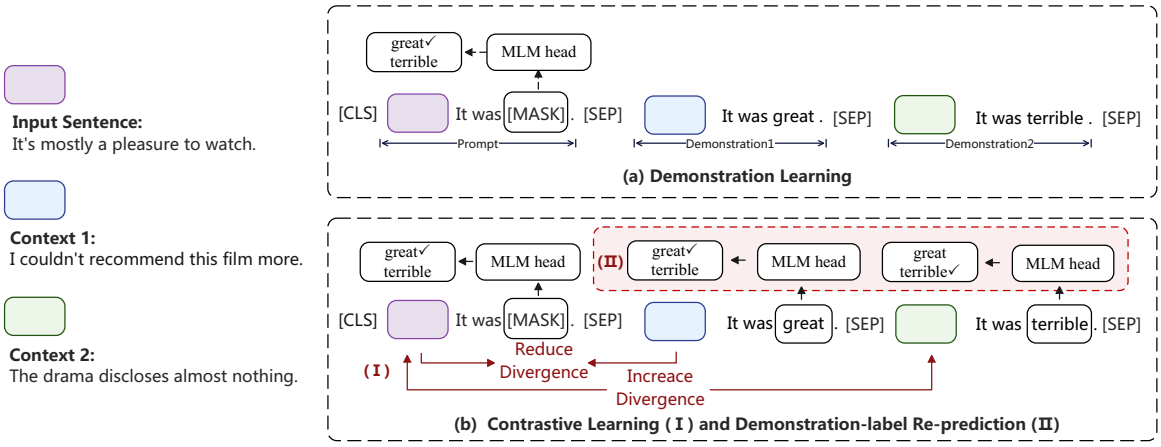

Figure 1: The overview of the proposed Imitation-Demo: (a) Conventional demonstration learning simply concatenate the demonstrations to the prompt. (b) Imitation-Demo reinforces the dependencies between prompt and demonstrations via contrastive learning (I) and demonstration-label re-prediction (II). For brevity, all sentences and contexts from demonstrations are represented by coloured boxes (illustrated in the left part). Best view in colours.

cally, to accurately locate similar samples, we introduce a contrastive learning mechanism (Chen et al., 2020; Robinson et al., 2021) to reorganize demonstrations by reducing the divergences of demonstration contexts among the same category while increasing those divergences between different categories. Besides, to solidify known knowledge, we leverage a demonstration-label re-prediction method to emphasize the positions of the answers in demonstrations. Even without introducing new parameters or any prediction computation, our proposed method achieves state-of-the-art performance on 5 out of 14 classification corpus. Compared to the strong baseline LM-BFF (Gao et al., 2021), Imitation-Demo achieves 1.11 points averaged improvement on the 14 datasets. Further study also shows that Imitation-Demo strengthens the association between prompt and demonstrations, which could provide the basis for exploring how demonstration learning works.

2 Methodology

Demonstration Learning. As illustrated in Fig. 1 (a), The prompt template x^{prompt} consists of input sentence x^{sent} and template x^{temp} containing mask token, i.e., $x^{prompt} = [x^{sent}, x^{temp}]$. Firstly, we leverage the pre-trained SBERT (Reimers and Gurevych, 2019) to retrieve the demonstrations (including context $x^{(k)}$ and label $y^{(k)}$) for the k-th category that has maximum semantic similarity to the raw prompt context. Then, the retrieved demonstrations are concatenated to the input prompt. After that, we convert the concatenated input sentence x^{in} to hidden vectors $\mathbf{h}^{in}$ via the RoBERTa model (Liu et al., 2019). The model is optimized by cross-entropy loss, and the goal of demonstration learning is to predict y^{mask} at the $[MASK]$ position from the hidden state of mask $\mathbf{h}^{mask}$ via MLM head. The whole process could be formulated as[1]:

$$
\begin{aligned}
x^{in} &= [x^{prompt}, (x^{(1)}, y^{(1)}), ..., (x^{(K)}, y^{(K)})] \\
\mathbf{h}^{in} &= \text{RoBERTa}(x^{in}) \\
\mathcal{L}_{mask} &= \text{CE}(\mathbf{h}^{mask}, \hat{Y}^{mask}) \\
p\left(y^{mask} \mid x_{\text{in}}\right) &= \text{MLM}(\mathbf{h}^{mask})
\end{aligned}
\tag{1}
$$

where $[.., .., ..]$ denotes concatenating diverse parts with sentence separator $[SEP]$. K is the number of categories. CE is short for cross-entropy loss, and $\hat{Y}^{mask}$ is the ground-truth labels from the predefined label-to-word mapping.

Demonstration Reorganization via Contrastive Learning. In demonstration learning, it is crucial to decide from which known demonstrations to select the repetitive patterns. Therefore, we introduce a contrastive learning mechanism to imitate human review behaviour by reorganizing the demonstrations based on their contexts. As shown in Fig. 1 (b)(I), we treat the demonstration contexts with identical categories to the input prompt as positive samples, and the others are regarded as negative ones. By pulling in positive samples and pulling out negative samples, the model could select the most relevant sample among the given

[1]Due to the space restriction, we only briefly describe the general process of demonstration learning, please refer to Gao et al. (2021) for more details.

demonstrations more precisely. In the experiment, we apply mean-pooling operations on the hidden states of positive, negative demonstration contexts $\mathbf{h}^+$, $\mathbf{h}^-$, and input sentence $\mathbf{h}^{in}$, obtaining the sentence representations $\mathbf{s}^+$, $\mathbf{s}^-$, and $\mathbf{s}^{in}$. Inspired by Robinson et al. (2021) in computer vision, we introduce HCL loss to ensure intra-class compactness while increasing inter-class distances:

$$\mathcal{L}_{context} = E\left[-\log \frac{e^{s^{in}\cdot s^+}}{e^{s^{in}\cdot s^+} + \sum_{i=1}^{N} e^{s^{in}\cdot s^-}}\right] \quad (2)$$

where $\cdot$ is the dot product operation, N is the number of negative contexts in the task, and $E\,[..]$ denotes calculating the mean value.

Demonstration-label Re-prediction. We further utilize a demonstration-label re-prediction method to mimic human review behaviour by recovering the labels from all the given demonstration contexts. Specifically, the target of our model is not only to identify the category of $[MASK]$ token, but also to classify the tokens located in demonstration label positions. Take the binary classification task in Fig. 1 (b)(II) as an example, more than predicting the class of the mask token, the model also requires to predict y^{great} and y^{terri} (i.e., $great$ and $terrible$) based on the hidden states $\mathbf{h}^{great}$ and $\mathbf{h}^{terri}$ at corresponding label positions.

During training, the cross-entropy loss is utilized to calculate $\mathcal{L}_{great}$ and $\mathcal{L}_{terri}$ for different demonstration labels, then we sum them up to obtain the demonstration-label re-prediction loss $\mathcal{L}_{label}$:

$$\begin{aligned} \mathcal{L}_{great} &= \text{CE}(\mathbf{h}^{great}, \hat{Y}^{great}) \\ \mathcal{L}_{terri} &= \text{CE}(\mathbf{h}^{terri}, \hat{Y}^{terri}) \\ \mathcal{L}_{label} &= \mathcal{L}_{great} + \mathcal{L}_{terri} \end{aligned} \quad (3)$$

where $\hat{Y}^{great}$ and $\hat{Y}^{terri}$ are the ground-truth labels at diverse demonstration label positions.

Similar contrastive learning and demonstration-label re-prediction operations can also be performed for the multi-category classification tasks. The overall loss of Imitation-Demo is defined as follows:

$$\mathcal{L} = \mathcal{L}_{mask} + \alpha\mathcal{L}_{label} + \beta\mathcal{L}_{context} \quad (4)$$

where α, β are weight coefficients to control the importance of different components.

3 Experiments

Experiments Settings. Following the settings in Gao et al. (2021), we evaluate on 14 classification datasets. For SNLI (Bowman et al., 2015), SST-2 (Socher et al., 2013), CoLA (Warstadt et al., 2019), MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), RTE (Dagan et al., 2005; Giampiccolo et al., 2007; Bentivogli et al., 2009), MRPC (Dolan and Brockett, 2005), QQP[2] and SST-B (Cer et al., 2017), we use the original development sets for testing. For MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), MPQA (Wiebe et al., 2005) and Subj (Pang and Lee, 2004), we randomly sample 2,000 examples as the testing set. For SST-5 (Socher et al., 2013) and TREC (Voorhees and Tice, 2000), we use the official test sets. F1 score (F1) are adopted as the evaluation metric of MRPC and QQP, and the other datasets utilize accuracy (acc) as the evaluation criteria.

	MRPC	SNLI	SST-2
Imitation-Demo	80.8 (3.2)	80.0 (3.3)	93.1 (0.5)
LM-BFF*	79.7 (3.2)	77.8 (0.6)	92.1 (1.5)
Imitation-Demo*	74.4 (9.2)	76.0 (5.2)	91.0 (1.3)

Table 1: Results when using demonstrations with random labels. * denotes trained with random labels.

Parameters Setting We implement all the baselines and our frameworks using PyTorch (Paszke et al., 2019). The pre-trained *RoBERTa-large* model and *roberta-large-nli-stsb-mean-tokens* SBERT (Reimers and Gurevych, 2019) from huggingface[3] are applied in the experiments. We get 16 samples per class during training for all models. In order to control the smoothness of the exponential functions when calculation contrastive learning loss, we divide every mean-pooling results with temperature T. Grid search mechanisim are utilized to select optimal hyper-parameter combinations on each split. Finally we select the the coefficients α and β as 1 and 5, respectively. The temperature T is set as 5 and the batch size is 16. The other hyper-parameters and the prompt templates are identical to the default settings in LM-BFF (Gao et al., 2021) for fair comparison. We report the average performance of models trained on 5 different randomly sampled training and dev splits, the random seeds are fixed as 13, 32, 42 ,87 , 100, respectively.

Compared Methods. (1) **Majority**, which select the majority class of the dataset; (2) **Prompt-based zero-shot**: which use prompt tunning in zero-shot situations; (3) **"GPT-3" in-context learn-**

[2]https://www.quora.com/q/quoradata/

[3]https://github.com/huggingface/transformers

	SST-2 (acc)	SST-5 (acc)	MR (acc)	CR (acc)	MPQA (acc)	Subj (acc)	TREC (acc)
Majority	50.9	23.1	50.0	50.0	50.0	50.0	18.8
Prompt-based zero-shot	83.6	35.0	80.8	79.5	67.6	51.4	32.0
"GPT-3" in-context learning	84.8 (1.3)	30.6 (0.9)	80.5 (1.7)	87.4 (0.8)	63.8 (2.1)	53.6 (1.0)	26.2 (2.4)
Fine-tuning	81.4 (3.8)	43.9 (2.0)	76.9 (5.9)	75.8 (3.2)	72.0 (3.8)	90.8 (1.8)	88.8 (2.1)
P-tuning	92.2 (0.4)	-	86.7 (1.2)	91.8 (1.1)	-	90.3 (2.2)	86.3 (4.5)
DART	**93.5** (0.5)	-	88.2 (1.0)	91.8 (0.5)	-	90.7 (1.4)	87.1 (3.8)
Li's	92.8 (0.6)	50.7 (2.9)	**89.4** (0.8)	90.5 (2.2)	83.2 (1.4)	92.1 (0.7)	87.2 (3.8)
Demo-tuning (LM-BFF)	93.2 (0.4)	50.1 (0.4)	87.9 (0.6)	91.5 (0.6)	85.9 (1.5)	92.3 (0.6)	**90.7** (4.5)
LM-BFF + SupCon	94.2 (0.7)	**54.0** (0.8)	89.6 (0.8)	91.0 (1.4)	86.9 (1.1)	92.4 (0.6)	89.8 (1.8)
EFL ♡	91.1 (1.5)	41.8 (1.6)	85.7 (3.7)	87.7 (5.4)	75.8 (4.8)	91.7 (1.8)	88.1 (2.3)
LM-BFF ♡	92.2 (1.4)	51.2 (1.6)	88.2 (0.9)	91.8 (1.5)	85.5 (4.2)	90.9 (1.9)	87.6 (4.8)
Imitation-Demo (ours)	93.1 (0.5)	52.3 (0.6)	89.1 (1.0)	**91.8** (0.7)	**87.7** (1.2)	**92.4** (1.1)	89.1 (3.2)
Prompt-based Fine-tuning (man) ♡	92.6 (0.5)	47.4 (2.5)	87.0 (1.2)	90.3 (1.0)	84.7 (2.2)	91.2 (1.1)	84.8 (5.1)
+ demonstrations♡	92.2 (1.4)	51.2 (1.6)	88.2 (0.9)	91.8 (1.5)	85.5 (4.2)	90.9 (1.9)	87.6 (4.8)
+ demonstration-label re-prediction	92.8 (0.7)	51.4 (1.0)	89.2 (1.0)	92.2 (1.2)	87.5 (1.0)	92.1 (1.6)	89.9 (3.1)
+ contrastive learning	93.1 (0.5)	52.3 (0.6)	89.1 (1.0)	91.8 (0.7)	87.7 (1.2)	92.4 (1.1)	89.1 (3.2)
	MNLI (acc)	MNLI-mm (acc)	SNLI (acc)	QNLI (acc)	RTE (acc)	MRPC (F1)	QQP (F1)
Majority	32.7	33.0	33.8	49.5	52.7	52.7	0.0
Prompt-based zero-shot	50.8	51.7	49.5	50.8	51.3	61.9	49.7
"GPT-3" in-context learning	52.0 (0.7)	53.4 (0.6)	47.1 (0.6)	53.8 (0.4)	60.4 (1.4)	45.7 (6.0)	36.1 (5.2)
Fine-tuning	45.8 (6.4)	47.8 (6.8)	48.4 (4.8)	60.2 (6.5)	54.4 (3.9)	76.6 (2.5)	60.7 (4.3)
P-tuning	61.5 (2.1)	-	72.3 (3.0)	64.3(2.8)	-	76.2 (2.3)	65.6 (3.0)
DART	67.5 (2.6)	-	75.8 (1.6)	66.7 (3.7)	-	78.3 (4.5)	67.8 (3.2)
Li's	69.2 (4.0)	71.0 (3.5)	79.3 (3.2)	69.0 (4.5)	**74.2** (3.1)	73.2 (7.5)	68.2 (3.4)
Demo-tuning (LM-BFF)	71.0 (2.0)	72.8 (1.5)	78.7 (1.9)	**73.1** (1.8)	70.0 (3.4)	78.4 (2.3)	70.2 (1.7)
LM-BFF + SupCon	**72.4** (2.0)	**74.2** (1.9)	79.6 (2.6)	71.1 (6.8)	71.8 (1.1)	77.8 (4.6)	**74.0** (2.5)
EFL ♡	65.8 (3.7)	68.5 (2.8)	78.2 (1.3)	67.6 (5.5)	68.9 (1.5)	77.4 (6.3)	67.0 (2.9)
LM-BFF ♡	69.6 (2.9)	71.3 (2.6)	78.0 (3.6)	68.8 (5.4)	68.7 (2.3)	77.3 (6.0)	68.7 (4.7)
Imitation-Demo (ours)	71.4 (0.9)	72.0 (2.0)	**80.0** (3.3)	70.5 (3.3)	71.5 (1.5)	**80.8** (3.2)	70.9 (1.5)
Prompt-based Fine-tuning (man) ♡	68.3 (2.3)	70.5 (1.9)	77.2 (3.7)	64.5 (4.3)	69.1 (3.6)	74.5 (5.3)	65.5 (5.3)
+ demonstrations♡	69.6 (2.9)	71.3 (2.6)	78.0 (3.6)	68.8 (5.4)	68.7 (2.3)	77.3 (6.0)	68.7 (4.7)
+ demonstration-label re-prediction	71.3 (0.9)	72.5 (1.4)	79.6 (3.2)	70.3 (4.1)	70.8 (3.4)	77.0 (2.6)	68.8 (2.6)
+ contrastive learning	71.4 (0.9)	72.0 (2.0)	80.0 (3.3)	70.5 (3.3)	71.5 (1.5)	80.8 (3.2)	70.9 (1.5)

Table 2: Overall results on RoBERTa-large with 16 samples per class. We report the mean (variance) of models trained on 5 different randomly sampled training and dev splits. Prompt-based Fine-tuning (man) indicates trained with manually designed templates. ♡ denotes we re-implement the EFL and LM-BFF models for fair comparisons.

ing, which use the in-context learning proposed in RoBERTa with no parameter updating; (4) **Fine-tuning**; (5) **P-tuning** (Liu et al., 2021b), which employ trainable continuous prompt embeddings; (6) **DART** (Zhang et al., 2021), which differentially optimize the prompt template and the target label during the backpropagation process; (7) **Li's** (Li et al., 2022), which reformulate a classification or a regression task as a token-replaced detection problem utilizing pre-trained model Electra (Clark et al., 2020); (8) **Demo-tuning (LM-BFF)** (Liang et al., 2022), which select "mask token" output feature as the input for contrastive learning to get a good representation of "virtual demonstration". We select the LM-BFF as the basic backbone model for fair comparisons. (9) **LM-BFF + SupCon** (Jian et al., 2022), which propose a supervised contrastive framework that clusters inputs from the same class under different augmented "views" and repel the ones from different classes. The LM-BFF is selected as the basic model. (10) **EFL** (Wang et al., 2021), which reformulate potential NLP task into an entailment one. (11) **LM-BFF** (Gao et al., 2021), which manually design templates and augment prompt tuning with demonstrations.

Main Results. From the experiment results illustrated in Table 2, we can conclude that: (1) The methods leveraging demonstrations (e.g. LM-BFF and Imitation-Demo) generally achieve productive results, proving the superiority of demonstration learning mechanism. (2) Compared to those methods that utilize continuous prompt embeddings or reformulate the task formats to boost experiment results, Imitation-Demo achieves state-of-the-art results on 5 out of 14 datasets in the original mask-prediction way without introducing additional pa-

	QQP	MNLI-mm	MNLI
LM-BFF	1.11	1.02	1.01
Imitation-Demo	1.16	1.04	1.05

Table 3: Averaged RoBERTa attention results pointing from demonstrations to prompt. The values are normalized by default RoBERTa pre-training weights.

rameters or any prediction computation. The performance gain indicates that Imitation-Demo could effectively promote experiment results by reinforcing the connections between the prompt and demonstrations. (3) Ablation experiment results in the lower part of Table 2 illustrate the effectiveness of the proposed demonstration reorganization and demonstrations-label re-prediction methods.

Analysis. Extensive experiments are conducted to show that our human-like imitation mechanisms enhance the connection between prompt and demonstration. Firstly, when trained with random demonstration labels, as shown in Table 1, we observe that Imitation-Demo has a greater drop rate than LM-BFF, indicating $[MASK]$ is dependent more on the semantic information from demonstrations. This finding could explain why there is little performance degradation when using random demonstration labels in Min et al. (2022) to some extent. Moreover, following Wei et al. (2021), we further conduct an experiment to show the review process with attention weights of the RoBERTa backbone. We average the total 384 attention heads of Roberta-large pointing from demonstrations to prompt, then normalize the values by default RoBERTa pre-trained weights. From the results in Table 3, we observe Imitation-Demo received larger attention values. The result indicates that our approach could direct the RoBERTa model by modifying the corresponding attention weights and guiding prompt to focus more on the clues brought by demonstrations. Since the models are trained in a few-shot scenario, the weights of models are not tuned heavily, thus we do not observe significant average attention score difference between the proposed Imitation-Demo and baseline method. However, with only 16 samples per class for training, Imitation-Demo can already show higher averaged attention weights compared with baseline method, indicating stronger connections between prompt and demonstrations.

4 Conclusion

In this paper, we propose imitation demonstration learning (Imitation-Demo) to reinforce the correlations between prompt and given demonstrations. Inspired by the human review process, we introduce contrastive learning to locate similar samples and demonstration-label re-prediction mechanisms to solidify known knowledge. Experiments show that our method consistently outperforms other baselines on 5 out of 14 classification datasets in the few-shot settings. We hope this work could inspire the exploration of the working mechanism of demonstration learning and toward better few-shot learning abilities.

Limitations

Although the experiment results have illustrated the effectiveness of the proposed Imitation-Demo method, we have to admit that our work has the following limitations:

1) This article is based on that the readers have some knowledge of prompt-based learning or demonstration learning. Due to the space limitation, we can only briefly describe the basic process of the demonstration learning, which may make the article a bit obscure and difficult to follow.

2) Imitation-Demo does not achieve state-of-the-art on all the datasets, but outperforms other strong baselines on 5 out of 14 datasets. Besides, it consistently surpasses the demonstration learning-based baseline LM-BFF. Since Imitation-Demo is trained without introducing new parameters and explores the working principle of demonstration learning from a certain perspective, we believe the results are acceptable.

References

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. In *TAC*.

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, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics.

Daniel M. Cer, Mona T. Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity multilingual

and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, pages 1–14. Association for Computational Linguistics.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing, IWP@IJCNLP 2005, Jeju Island, Korea, October 2005, 2005*. Asian Federation of Natural Language Processing.

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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3816–3830. Association for Computational Linguistics.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and William B Dolan. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004*, pages 168–177. ACM.

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

Yiren Jian, Chongyang Gao, and Soroush Vosoughi. 2022. Contrastive learning for prompt-based few-shot language learners. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 5577–5587. 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, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3045–3059. Association for Computational Linguistics.

Zicheng Li, Shoushan Li, and Guodong Zhou. 2022. Pre-trained token-replaced detection model as few-shot learner. *CoRR*, abs/2203.03235.

Xiaozhuan Liang, Ningyu Zhang, Siyuan Cheng, Zhenru Zhang, Chuanqi Tan, and Huajun Chen. 2022. Contrastive demonstration tuning for pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 799–811. Association for Computational Linguistics.

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

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. GPT understands, too. *CoRR*, abs/2103.10385.

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. *CoRR*, abs/1907.11692.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *CoRR*, abs/2202.12837.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, 21-26 July, 2004, Barcelona, Spain*, pages 271–278. ACL.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 115–124. The Association for Computer Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive learning with hard negative samples. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Teven Le Scao and Alexander M. Rush. 2021. 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, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2627–2636. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. 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, EACL 2021, Online, April 19 - 23, 2021*, pages 255–269. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1631–1642. ACL.

Ellen M. Voorhees and Dawn M. Tice. 2000. Building a question answering test collection. In *SIGIR 2000: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 24-28, 2000, Athens, Greece*, pages 200–207. ACM.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. 2021. Entailment as few-shot learner. *CoRR*, abs/2104.14690.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Trans. Assoc. Comput. Linguistics*, 7:625–641.

Kaiwen Wei, Xian Sun, Zequn Zhang, Jingyuan Zhang, Zhi Guo, and Li Jin. 2021. Trigger is not sufficient: Exploiting frame-aware knowledge for implicit event argument extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4672–4682. Association for Computational Linguistics.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Lang. Resour. Evaluation*, 39(2-3):165–210.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

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

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitation Section.

☑ A2. Did you discuss any potential risks of your work?
Limitation Section.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Introduction Sections.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 3

☑ B1. Did you cite the creators of artifacts you used?
Section 3

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 3

☒ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

In Section 3.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
In Section 3.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
In Section 3.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
In Section 3.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
In Section 3.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.