Understanding Demonstration-based Learning from a Causal Perspective

Ruiyi Zhang
Adobe Research
ruizhang@adobe.com

Tong Yu
Adobe Research
tyu@adobe.com

Abstract

Demonstration-based learning has shown impressive performance in exploiting pretrained language models under few-shot learning settings. It is interesting to see that demonstrations, even those composed of random tokens, can still improve performance. In this paper, we build a Structural Causal Model (SCM) to understand demonstration-based learning from causal perspectives and interpret random demonstrations as interventions on the demonstration variable within the causal model. We investigate the causal effects and find that the concurrence of specific words in the demonstration will induce bias, while randomly sampled tokens in the demonstration do not. Based on this finding, we further propose simple ways to construct random demonstrations, which even outperform hand-crafted, meaningful demonstrations on public sequence labeling benchmarks[1].

1 Introduction

Large pretrained language models (PLMs) have recently shown great progress (Devlin et al., 2019; Liu et al., 2019a; Lewis et al., 2020; Xie et al., 2020; Huang et al., 2021). These models, such as GPT-4 (Peng et al., 2023), PALM (Anil et al., 2023), and Llama (Touvron et al., 2023), have shown human-level capability with only a few illustrative examples (Lake et al., 2015). Specifically, demonstration-based learning has been introduced to augment the input with demonstrations, *i.e.*, the input and expected output pairs. Brown et al. (2020) simply picked up to a small number of sampled instances and directly concatenated them with the input to perform *in-context learning*. Lee et al. (2022) concatenated the input with task demonstrations to create augmented input and fed them into PLMs to obtain improved token representations to do sequence labeling in a classifier-based fine-tuning way.

However, how and why such demonstrations help still remains unclear, and there has been a growing amount of work investigating the mechanisms of demonstration-based learning. Min et al. (2022) investigated in-context learning with demonstrations under zero-shot settings and found that input with random labels can still produce performance comparable to that of correct labels. Zhang et al. (2022a) replaced every token in the demonstration with random ones and still surprisingly observed good few-shot learners even when the demonstration is meaningless. These observations conflict with some existing hypotheses (Gao et al., 2021; Lee et al., 2022) that models are learning meaningful knowledge from demonstrations.

To better understand demonstration-based learning, we take a deeper dive into the random construction of demonstrations. Specifically, we first build a Structural Causal Model (SCM) to understand demonstration-based learning from a *Causal Perspective*. A causal view is developed to explore the spurious correlations between demonstrations and few-shot training samples. Based on the intervention on the demonstration variable in the SCM, we design multiple simple and effective ways to construct random demonstrations. These methods are evaluated on structured prediction tasks with carefully designed experiment setups. Empirical results show that carefully designed random demonstrations can outperform meaningful demonstrations under the few-shot learning setting. This finding suggests that meaningless demonstrations can still provide valid information for PLMs. Moreover, random demonstrations allow the learning algorithm to identify important features and patterns in the data more effectively than homogeneous handcrafted demonstrations.

2 Background

In this section, we introduce the background of sequence labeling and demonstration-based learning.

[1] Code available at: github.com/zhangry868/RandDemo

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 1465–1475
July 9-14, 2023 ©2023 Association for Computational Linguistics

Sentence:	The Algerian War of Independence marked the end of French colonial rule in North Africa .	
Labels:	O B-MISC I-MISC I-MISC I-MISC O O O O B-ORG O O O B-LOC I-LOC O	
	Biased: French -> [ORG]	Desired: French -> [MISC]
Standard:	[SEP] The unnamed suspect left the British colony after being detained and then freed by the Independent Commission Against Corruption (ICAC) , the radio said . *Independent Commission Against Corruption is ORG* . [SEP] [...]	
Random:	[SEP] Lebanon First Ed ##up CBOE suspect CB Chicago K Chicago Board Options Exchange ##ty Paul Gascoigne CBOE Monday Les into vintage I ##tion Ferdinand ##ca Op [SEP] [...]	

Table 1: An example from the CoNLL03 dataset with different demonstrations. The NER model takes both the sentence and a demonstration as its inputs. The top two rows show examples of the NER model inputs and outputs with standard demonstrations. A biased prediction for 'French' is caused by the demonstration bias. The bottom three lines show three different demonstrations: Standard and Random demonstrations. The notation '[SEP][...]' indicates that there are demonstrations for other classes, which have been omitted due to limited space.

Sequence Labeling Given an input sentence $\mathbf{x} = [x_1, x_2, \cdots, x_n]$ composed of n tokens, the sequence labeling task is to predict a tag $y_i \in Y \cup \{O\}$ for each token x_i, where Y is a predefined set of tags, and O denotes outside a tagged span. In the few-shot setting, we only have K-shot support set $\mathcal{S}$ for training which contains K examples for each tag type. This setting usually refers to K-shot learning. Modern sequence labeling models are usually composed of an encoder and a classification head. The encoders are PLMs such as BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019b), which provides contextualized representations for each token $\mathbf{h} = [h_1, h_2, \cdots, h_n]$ given the natural language sequence $\mathbf{x} = [x_1, x_2, \cdots, x_n]$. The classification head takes these contextualized representations and predicts the label l_i for each token x_i. The model is optimized with the standard cross-entropy loss.

Demonstration-based Learning Given some demonstration $\tilde{\mathbf{x}}$, we concatenate the original input $\mathbf{x}$ with its demonstration $\tilde{\mathbf{x}}$ as $[\mathbf{x}; \tilde{\mathbf{x}}]$. We then feed the demonstration-augmented input $[\mathbf{x}; \tilde{\mathbf{x}}]$ into the encoder, and get the contextualized representation $[\mathbf{h}; \tilde{\mathbf{h}}]$. The classification head takes $\mathbf{h}$ as the input and estimate the corresponding token's label l_i in the original natural-language sequence. Please note that we use identical demonstrations during training and testing (Lee et al., 2022).

Demonstration Construction To construct demonstrations, we first sample an entity $e^{(c)}$ for each label type $t^{(c)}$, and its context $s^{(c)}$ from support set $\mathcal{S}$. Then we convert them into a natural language sequence $d^{(c)} = T(s^{(c)}, e^{(c)}, t^{(c)})$, where T is the template operator and previous works (Lee et al., 2022) focus on finding more effective templates. With these sequences $[d^{(c_i)}]_{i=1}^{|Y|}$ with different tags c_i, a demonstration $\tilde{\mathbf{x}}$ is built by concatenating them together: $\tilde{\mathbf{x}} = d^{(c_1)} \oplus d^{(c_2)} \oplus \cdots \oplus d^{(c_{|Y|})}$, where $\oplus$ is the concatenation operator. An effective template, such as the one used in Lee et al. (2022), is "$s^{(c)}$. $e^{(c)}$ is $t^{(c)}$.". Here, we refer the "$e^{(c)}$ is $t^{(c)}$." part in the template as labeling part of the demonstration.

3 Demonstration-based Learning from a Causal Perspective

In this section, we give a specific example to show the potential bias and understand demonstration-based learning from a causal perspective. Specifically, we first introduce a Structural Causal Model (SCM) (Pearl et al., 2000) to describe the mechanism and identify the induced bias. Then, we perform demonstration variable intervention and propose multiple simple and effective random demonstration templates inspired by our causal model.

We observe that the frequent co-occurrence of tokens in the classical demonstrations generate harmful superficial patterns which is misleading to the model and leads to biased predictions (Zhang et al., 2022a; Min et al., 2022). A specific example with different demonstrations is provided in Table 1, where the entity to predict is French. Following previous work (Zhang et al., 2022a), the observed demonstrations (*i.e.*, standard demonstration) provides some biased information: the concurrency of British and ICAC, which is an organization (ORG), may lead to biased predictions: French is labeled as an Organization while its desired prediction is other classes (MISC). Intuitively, the co-occurrence of two specific words in the demonstration may induce bias, while randomly sampled tokens in the demonstration do not. This specific example suggests why random demonstrations may sometimes perform better than standard ones.

3.1 Causal Model

To study the causal relationship between the NER model and its training data, and explain the role of the demonstration, we introduce a SCM to describe

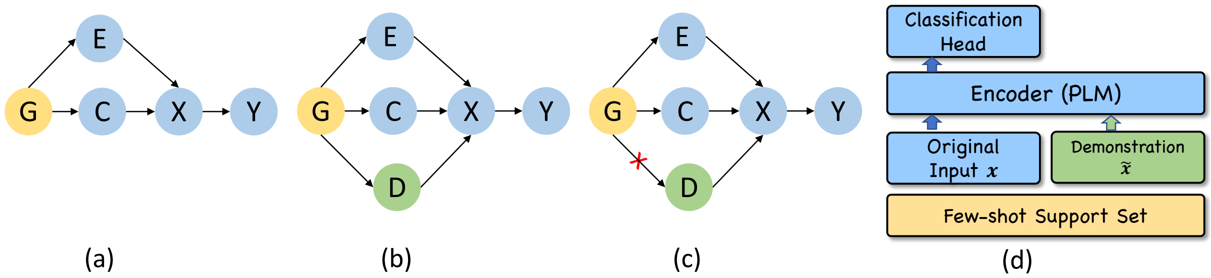

Figure 1: Causal views of NER. (a) shows a traditional NER model (Zeng et al., 2020), (b) shows the demonstration-based NER model under the causal view. With demonstration D, the backdoor path $G \rightarrow D \rightarrow X$ exists, which further introduces the bias. (c) shows the demonstration-based NER model with debiasing techniques, and the red cross means intervention. (d) is model architecture overview between classical and demonstration-based learning.

the inference step in NER models. Figure 1 shows the SCM of NER models. There are mainly 6 variables in NER models: 1) *Demonstration Tokens* D, the tokens which form the demonstration; 2) *Context Tokens* C, the tokens that are related to the context; 3) *Entity Tokens* E, the tokens which are entities; 4) *Input Example* X, which is composed of C and E in the traditional model and composed of C, E and D in the demonstration-based models; 5) *Unobserved confounders* G, a confounding variable (not a concrete token) that influences the generation of C, E and D; 6) *Evaluation result* Y, the evaluation result (the F1 score) of the NER models. Under the causal view, the key difference between the traditional NER model and the demonstration-based NER model is that, the demonstration-based NER model has an additional node D. With the introduction of the demonstration D, a backdoor path $G \rightarrow D \rightarrow X$ exists, which further introduces the bias.

Inspired by our SCM model (Figure 1b), we develop sampling techniques to generates new counterfactual examples by the interventions on the existing observational examples to alleviate this bias. The benefits of interventions on E and C have been studied in (Zeng et al., 2020). In this paper, we focus on understanding the role of demonstrations in NER models under the causal view. We understand the co-occurrence of tokens and harmful superficial patterns from the causal perspective and focus on using interventions on the demonstration variable to create new counterfactual demonstrations.

3.2 Controllable Random Demonstrations

In this section, we first provide a running example to better understand the induced bias from human-crafted demonstrations and then present different ways of intervention on the demonstration tokens. The intervention is implemented via controllable random demonstrations to create new counterfactual examples, as replacing standard demonstrations with random tokens can remove induce bias and still make the model a good few-shot learner (Zhang et al., 2022a).

In Lee et al. (2022), an effective template T is "$s^{(c)}$. $e^{(c)}$ is $t^{(c)}$, and an example demonstration $d^{(c)}$ can be "[SEP] Obama returns to White House. Obama is PER.". Intuitively, the model understands the demonstrations and then better performs inference. However, random demonstrations can still bring performance improvement (Zhang et al., 2022a). The random template is as simple as "$[s_i]_{i=1}^{L}$", where $s_i \in p$, and p is a token distribution. Random demonstrations are composed of L tokens randomly sampled from p.

Demonstration Intervention We use the intervention on the demonstration tokens to create new counterfactual examples, to alleviate the biases. If we do not carefully design D, the backdoor path will exist and the model performance is degraded. Our causal framework enables us to think about the problem from a causal perspective and guides us how to properly design D. We denote uniform distribution composed of vocabulary words of the PLMs as $p_{\mathcal{V}}$. Given the token distribution $p_{\mathcal{V}}$, for any word $w_i \in p_{\mathcal{V}}$, we have $p_{\mathcal{V}}(w_i) = \frac{1}{|\mathcal{V}|}$. Then we have a plain way to construct random demonstrations.

An important observation is that not all counterfactual examples are correct or useful. Hence, the intervention can be better implemented by replacing the uniform distribution with a non-uniform distribution, *i.e.*, by adding or removing words and changing specific words' probabilities. Some mechanism is needed to identify good counterfactual demonstrations, to avoid introducing noise. An intuitive solution is that we consider tokens from the support set are more helpful as PLMs are fine-

Mode	NER						Chunking		
	CoNLL03			OntoNotes 5.0			CoNLL00		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
No Demo.	$28.71_{\pm 10.31}$	$39.96_{\pm 11.25}$	$22.68_{\pm 9.09}$	$37.37_{\pm 7.58}$	$33.80_{\pm 6.79}$	$41.92_{\pm 8.85}$	$63.17_{\pm 4.22}$	$59.28_{\pm 5.05}$	$67.72_{\pm 3.51}$
Standard	$45.86_{\pm 6.08}$	$47.38_{\pm 5.93}$	$44.75_{\pm 7.07}$	$40.21_{\pm 7.65}$	$32.51_{\pm 6.87}$	$52.82_{\pm 8.28}$	$70.55_{\pm 3.08}$	$66.53_{\pm 4.40}$	$75.21_{\pm 2.11}$
Random	$41.33_{\pm 7.36}$	$45.41_{\pm 7.37}$	$38.22_{\pm 7.65}$	$39.71_{\pm 7.56}$	$32.28_{\pm 6.56}$	$51.63_{\pm 8.75}$	$69.28_{\pm 2.78}$	$64.75_{\pm 3.85}$	$74.57_{\pm 1.66}$
Rand-S	$45.55_{\pm 8.02}$	$46.84_{\pm 7.71}$	$44.60_{\pm 8.62}$	$41.60_{\pm 7.05}$	$33.96_{\pm 6.29}$	$53.75_{\pm 7.80}$	$70.63_{\pm 3.01}$	$66.24_{\pm 4.29}$	$75.75_{\pm 1.70}$
Rand-W	$45.93_{\pm 7.57}$	$47.79_{\pm 7.42}$	$44.50\ _{\pm 8.13}$	$45.49_{\pm 3.77}$	$37.82_{\pm 3.64}$	$57.18_{\pm 4.17}$	$72.15_{\pm 3.16}$	$68.00_{\pm 4.42}$	$76.94_{\pm 1.67}$
Rand-E	$47.32_{\pm 7.42}$	$48.96_{\pm 7.02}$	$46.02_{\pm 8.11}$	$46.06_{\pm 3.84}$	$38.32_{\pm 3.65}$	$57.81_{\pm 4.31}$	$74.02_{\pm 2.93}$	$70.37_{\pm 4.23}$	$78.18_{\pm 1.75}$

Table 2: Main results for traditional token classification method (**No Demo.**) and demonstration-based learning with different modes of demonstrations under 5-shot scenario.

tuned on the support set. We expect to see a better downstream predictor when the demonstrations are constructed randomly from a intervened token distribution.

The difference between random demonstrations lies in the vocabulary and its associated probability distributions. We perform the interventions by controlling the vocabulary and changing the probability of random tokens. We encourage entity words (*e.g.*, ICAC, British) to appear more frequently compared to the others (*e.g.*, is). Based on the previous theoretical justification, we consider the following variants of constructing random demonstrations[2] construction methods as counterfactual alternatives of the standard demonstrations[3]:

- **Random**: random context with tokens uniformly sampled from PLMs vocabulary $\mathcal{V}$.
- **Rand-S**: random context with tokens uniformly sampled from unique words (*i.e.*, vocabulary) of support set, denoted as $\mathcal{S}$.
- **Rand-W** [4]: random context with tokens sampled from $\mathcal{S}$, and entity tokens in support set, denoted as $\mathcal{W}$; tokens from $\mathcal{W}$ have four times higher probability compared with those from $\mathcal{S}$.
- **Rand-E**: similar to Rand-W, but replace entity tokens with entities composed of coherent tokens in support set, denoted as $\mathcal{U}$.

4 Experimental Results

4.1 Experiment Setup

Datasets We conduct experiments on two sequence labeling tasks: (*i*) named entity recognition (NER) on dataset **CoNLL03** (Tjong Kim Sang and De Meulder, 2003), and **OntoNotes 5.0** (Weischedel et al., 2013); and (*ii*) chunking on dataset **CoNLL00** (Tjong Kim Sang and Buchholz, 2000). Following previous works Ma et al. (2021); Zhang et al. (2022a), we omit the 7 value types in OntoNotes and only consider the 6 most frequent types in CoNLL00. For few-shot data sampling, we follow the greedy sampling strategy proposed by Yang and Katiyar (2020) to sample K shots for each type in an increasing order with respect to their frequencies, the detailed algorithm can be found. For each dataset, we sample 5 different K-shot support sets and report mean and standard deviation of metrics. For each K-shot support set, we run the experiments with 3 random seeds.

Main Results We show the results for demonstration-based learning with different modes of demonstrations as well as classical sequence labeling with no demonstration in Table 2. The results show that demonstration-based method can consistently improve model performance. In demonstration-based methods, the Random approach shows the worst performance and Rand-S shows comparable results with the standard demonstrations, and the conclusion is consistent with previous works (Zhang et al., 2022a). Interestingly, if we modify the token sampling distributions and sample more entity or entity-related words as Rand-W and Rand-E, our model shows even better performance than standard meaningful demonstrations. The difference between Rand-W and Rand-E lies in whether there are complete entities, and the results show that adding complete entities instead of random entity words can lead to better performance. At the same time, it shows adding random tokens related to the support set can reduce the fine-tuned bias, which verifies our hypothesis in Section 3.1. Intuitively, the benefits of demonstration-based methods come from tokens of support sets $\mathcal{S}$ instead of meaningful demonstrations, as the standard demonstration sampled from the support set also shows good performance.

[2] Random: `[SEP] {random context}`

[3] Standard: `[SEP] {context} {entity} is {tag}`.

[4] Empirical results show sampling only from $\mathcal{W}$ leads to poor performance.

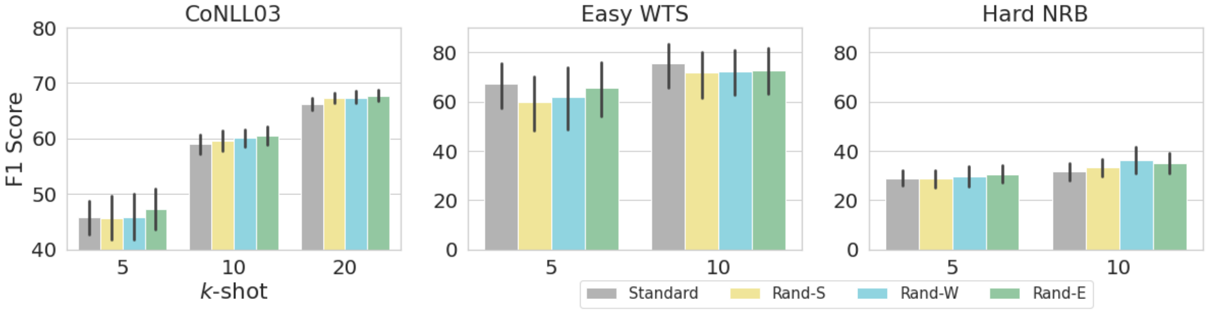

Figure 2: Results with different support set size on CoNLL03, NRB and WTS datasets.

Mode	CoNLL03	OntoNotes5.0	CoNLL00
No Demo.	$45.70_{\pm 8.13}$	$51.62_{\pm 2.76}$	$72.80_{\pm 3.53}$
Standard	$45.73_{\pm 7.29}$	$54.76_{\pm 2.36}$	$75.90_{\pm 1.95}$
Rand-S	$46.86_{\pm 6.50}$	$54.35_{\pm 2.67}$	$72.23_{\pm 3.42}$
Rand-W	$52.11_{\pm 6.15}$	$54.48_{\pm 2.35}$	$73.84_{\pm 2.19}$
Rand-E	$52.87_{\pm 7.64}$	$55.94_{\pm 2.38}$	$75.30_{\pm 3.06}$

Table 3: Main results (F1 scores) of RoBERTa-Large for traditional token classification with different modes of demonstrations under 5-shot scenario.

4.2 Analysis

Ablation Studies We further investigate whether the performance gain of demonstration-based learning changes over the size of support set. We present results of different modes of demonstrations under $K = 5, 10, 20$ shots in Figure 2. With more training examples in the support set, the relative performance gap between Rand-E and Standard remains, but it becomes smaller. This indicates that carefully designed random demonstrations show a consistent performance improvement upon standard demonstration. We also observe that the variance within each group becomes smaller as more data becomes available. Among random demonstrations, Rand-E consistently shows better performance than Rand-W and Rand-S, which verifies our hypothesis based on the SCM.

Additionally, we investigate the effect of using different base models and replace BERT with RoBERTa. The observed results for RoBERTa in Table 3 are consistent with those of BERT, demonstrating that Rand-E exhibits superior performance across different model architectures.

Name Regularity Bias Name Regularity Bias (Ghaddar et al., 2021; Lin et al., 2020) in NER occurs when a model relies on a signal from the entity name to make predictions and disregards evidence from the local context. Ghaddar et al. (2021) carefully designed a testbed utilizing Wikipedia disambiguation pages to diagnose the Name Regularity Bias of NER models. Details about the NRB dataset are provided in the appendix.

We use both the NRB and WTS (as control sets) datasets to evaluate the model trained with different modes of demonstrations on CoNLL03. The results show a smaller gap for random demonstrations, suggesting that random demonstration-based learning can better leverage context information instead of the name regularity patterns.

5 Conclusions

In this paper, we present a casual view to understand demonstration-based learning. Based on the structural causal model we constructed, we investigate the causal effects and discover that the concurrence of specific words in the demonstration can induce bias. To address this issue, we perform interventions by constructing random demonstrations. Our empirical results indicate that carefully designed random demonstrations consistently outperform meaningful demonstrations on public sequence labeling benchmarks.

6 Limitations

All our experiments are done on the sequence labeling task, and they can be further evaluated on sentence classification tasks with classifier-based fine-tuning since the [CLS] token used for classification represents the whole sentence. We provide a causal opinion on demonstration-based learning and a simple but not systematic method to alleviate the induced bias. Our demonstration-based learning builds upon previous works (Lee et al., 2022; Zhang et al., 2022a), where BERT or RoBERTa are used instead of Large Language Models, such as InstructGPT (Ouyang et al., 2022), PaLM (Chowdhery et al., 2022), and OPT (Zhang et al., 2022b). Furthermore, our conclusions are drawn from few-shot learning settings and cannot be directly applied to zero-shot inference.

References

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Abbas Ghaddar, Philippe Langlais, Ahmad Rashid, and Mehdi Rezagholizadeh. 2021. Context-aware Adversarial Training for Name Regularity Bias in Named Entity Recognition. *Transactions of the Association for Computational Linguistics*, 9:586–604.

Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. 2021. Few-shot named entity recognition: An empirical baseline study. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10408–10423, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Brenden Lake, Ruslan Salakhutdinov, and Joshua Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science*, 350:1332–1338.

Dong-Ho Lee, Akshen Kadakia, Kangmin Tan, Mahak Agarwal, Xinyu Feng, Takashi Shibuya, Ryosuke Mitani, Toshiyuki Sekiya, Jay Pujara, and Xiang Ren. 2022. Good examples make a faster learner: Simple demonstration-based learning for low-resource NER. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2687–2700, Dublin, Ireland. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. 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*, pages 7871–7880, Online. Association for Computational Linguistics.

Hongyu Lin, Yaojie Lu, Jialong Tang, Xianpei Han, Le Sun, Zhicheng Wei, and Nicholas Jing Yuan. 2020. A rigorous study on named entity recognition: Can fine-tuning pretrained model lead to the promised land? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7291–7300, Online. Association for Computational Linguistics.

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

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

Ruotian Ma, Xin Zhou, Tao Gui, Yiding Tan, Qi Zhang, and Xuanjing Huang. 2021. Template-free prompt tuning for few-shot NER. *CoRR*, abs/2109.13532.

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? *arXiv preprint arXiv:2202.12837*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al.

2022. Training language models to follow instructions with human feedback. *NeurIPS*.

Judea Pearl et al. 2000. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19(2).

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 shared task chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems*, volume 33, pages 6256–6268. Curran Associates, Inc.

Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6365–6375, Online. Association for Computational Linguistics.

Xiangji Zeng, Yunliang Li, Yuchen Zhai, and Yin Zhang. 2020. Counterfactual generator: A weakly-supervised method for named entity recognition. In *EMNLP*.

Hongxin Zhang, Yanzhe Zhang, Ruiyi Zhang, and Diyi Yang. 2022a. Robustness of demonstration-based learning under limited data scenario. In *EMNLP*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022b. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Dataset	$\|Y\|$	L	$\|D_{test}\|$
CoNLL03	4	18	3453
OntoNotes 5.0	11	21	12217
CoNLL00	6	36	2012

Table 4: **Data Statistics**. $|Y|$: # of entity types. L: average # of tokens in input sentence. $|D_{support}|$: average # of sentences in 5-shot support set over 5 different sub-samples. $|D_{test}|$: # of sentences in test set.

A Appendix

NRB Dataset Details The NRB dataset contains examples whose labels can be easily inferred from the local context, but they are difficult to be tagged by a popular NER system. The WTS dataset is a domain control set that includes the same query terms covered by NRB, but these can be correctly labeled by both the popular NER tagger and the local context-only tagger. Therefore, the gap between the NRB and WTS sets measures how effectively the model captures context information to predict token labels.

Effects of Sampling Probability We present two variants, Random-E[X] and Random-W[X], where X refers to how many times the probability of preferred tokens is higher. In this ablation study, we consistently observe that Random-E4 performs better than Random-E2, and Random-W4 outperforms Random-E4. However, if we increase the X value to a very large number, the performance deteriorates.

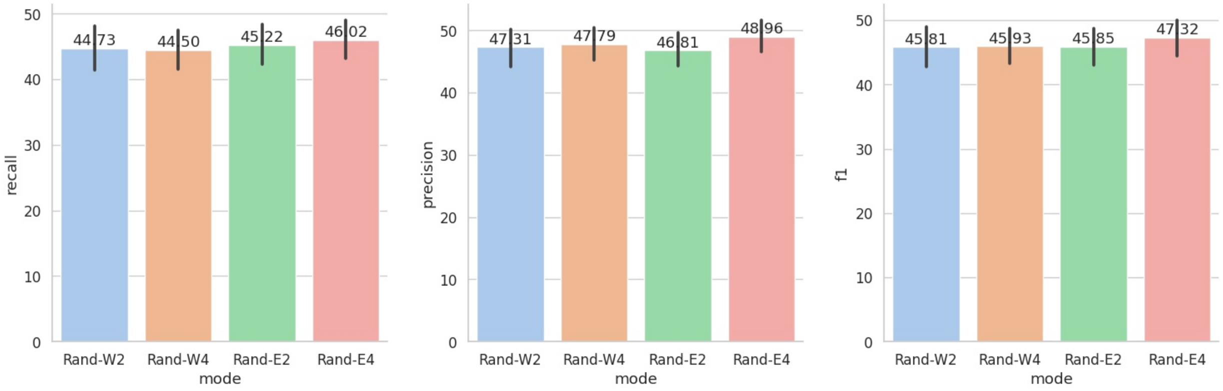

Figure 3: F1, Precision and Recall with more variants of Random Demonstrations on CoNLL03.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 6

☑ A2. Did you discuss any potential risks of your work?
Section 6

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

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

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
Not applicable. Left blank.

□ 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.?
Not applicable. Left blank.

□ 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.
Not applicable. Left blank.

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4

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?
Section 4

☑ 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?
Section 4

☑ 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.)?
Section 4

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.?
Not applicable. Left blank.

□ 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)?
Not applicable. Left blank.

□ 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?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.