Self-Improving for Zero-Shot Named Entity Recognition with Large Language Models

Tingyu Xie[1,2], **Qi Li**[1,2], **Yan Zhang**[3]*, **Zuozhu Liu**[2], **Hongwei Wang**[1,2]*
[1]College of Computer Science and Technology, Zhejiang University, China
[2]ZJU-UIUC Institute, Zhejiang University, China
[3]National University of Singapore, Singapore
{tingyuxie, hongweiwang}@zju.edu.cn, yanzhang.jlu@gmail.com

Abstract

Exploring the application of powerful large language models (LLMs) on the named entity recognition (NER) task has drawn much attention recently. This work pushes the performance boundary of zero-shot NER with LLMs by proposing a training-free self-improving framework, which utilizes an unlabeled corpus to stimulate the self-learning ability of LLMs. First, we use the LLM to make predictions on the unlabeled corpus using self-consistency and obtain a self-annotated dataset. Second, we explore various strategies to select reliable annotations to form a reliable self-annotated dataset. Finally, for each test input, we retrieve demonstrations from the reliable self-annotated dataset and perform inference via in-context learning. Experiments on four benchmarks show substantial performance improvements achieved by our framework. Through comprehensive experimental analysis, we find that increasing the size of unlabeled corpus or iterations of self-improving does not guarantee further improvement, but the performance might be boosted via more advanced strategies for reliable annotation selection.[1]

1 Introduction

There have been many works exploring new possibilities of the named entity recognition (NER) task in the era of large language models (LLMs) (OpenAI, 2022; Touvron et al., 2023; Chowdhery et al., 2022) recently. These studies include designing advanced prompting methods for zero-shot prediction or few-shot in-context learning (ICL) (Wei et al., 2023b; Wang et al., 2023; Xie et al., 2023; Li et al., 2023b), training task-specific LLMs for NER (Zhou et al., 2023; Sainz et al., 2023), and generating data with LLMs to train small specific models (Zhang et al., 2023; Ma et al., 2023; Josifoski et al., 2023).

In this work, we explore the possibility of pushing the performance boundary of zero-shot NER with LLMs via self-improving. We focus on the strict zero-shot scenarios where no annotated data is available but only an unlabeled corpus is accessible, and no training resource or auxiliary models are available. We propose a totally training-free self-improving framework for NER, which utilizes an unlabeled corpus to stimulate the self-learning ability of LLMs. The framework consists of the following three steps. (1) Step 1: we use LLMs to self-annotate the unlabeled corpus using self-consistency (SC, Wang et al., 2022). Each annotated entity is associated with a SC score, which is used as the measure of the reliability of this annotation. (2) Step 2: we select reliable annotation to form a reliable self-annotated dataset, during which diverse annotation selection strategies are explored, including entity-level threshold filtering, sample-level threshold filtering and two-stage majority voting. (3) Step 3: for each arrived test input, we perform inference via ICL with demonstrations from the reliable self-annotated dataset. Various strategies for demonstration retrieval are explored.

Our contributions include: (1) We proposed a training-free self-improving framework for zero-shot NER with LLMs. (2) This framework achieved significant performance improvements on four benchmarks. (3) We conduct comprehensive experimental analysis, finding that increasing the size of unlabeled corpus or iterations of self-annotating does not guarantee gains, but there might be room for improvements with more advanced strategies for reliable annotation selection.

2 Zero-Shot NER with Self-Improving

Motivation. To push the performance boundary of zero-shot NER with LLMs, we propose a self-improving framework under a strict zero-shot and low-resource setting: No annotated data but only an

*Corresponding authors.
[1]Code and data are publicly available: https://github.com/Emma1066/Self-Improve-Zero-Shot-NER

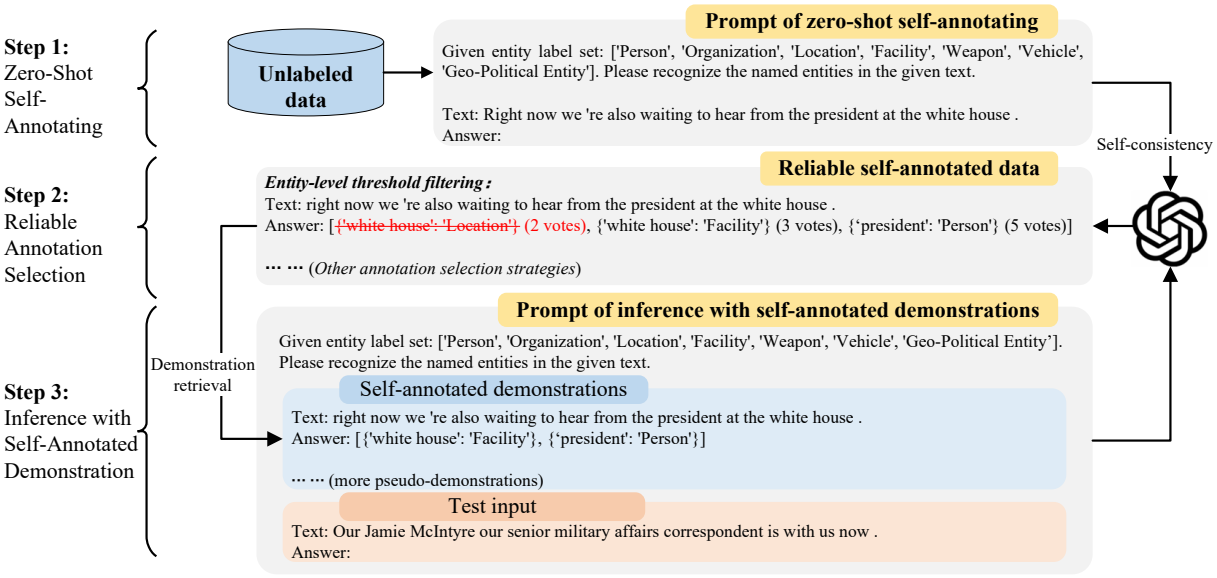

Figure 1: The overview of the proposed self-improving framework for zero-shot NER with LLM.

unlabeled corpus is available; No auxiliary model or training step is required. This study is orthogonal to previous prompt designing works, as any advanced prompting method can be applied to this framework. Fig. 1 shows the framework overview.

Task Formulation. Given an input sentence x, the NER task is to recognize the structure output y from x, which consists of a set of (e, t) pairs. e is an entity span, which is a sequence of tokens form x; t is the corresponding entity type, which belongs to a predefined entity type set.

2.1 Step 1: Zero-Shot Self-Annotating

We assume an unlabeled corpus $\mathcal{U} = \{x_i\}_{i=1}^{n}$ is available. We use the training set without labels as the unlabeled dataset in this work. For unlabeled sample x_i, we generate predictions with LLMs via zero-shot prompting, as shown in upper part of Fig. 1. This process is formulated as $y_i = \arg\max_y P(y|T, x_i)$, where T is the task instruction of NER, and $y_i = \{(e_i^j, t_i^j)\}_{j=1}^{m}$. We apply self-consistency (SC) (Wang et al., 2022) to obtain a SC score for each prediction, which will be used in step 2 for reliable annotation selection. We sample multiple answers from the model, and the vote for each predicted entity (e_i^j, t_i^j) is the times it appeared in all the sampled answers, which we denoted as entity-level SC score c_i^j. Then we get the sample-level SC score c_i for each input sentence x_i by taking the average SC score over all predicted entities in this sentence, i.e., $c_i = \frac{1}{m}\sum_j c_i^j$. For each self-annotated sample with SC scores, we can denote it as $(x_i, \{(e_i^j, t_i^j, c_i^j)\}_{j=1}^{m}, c_i)$.

2.2 Step 2: Reliable Annotation Selection

We assume that a higher SC score indicates a higher reliablity. Thus, we investigate the three following strategies for reliable annotation selection. (1) *Entity-level threshold filtering*, which drops the predicted entity e_i^j if $c_i^j < Th\_entity$, where Th_entity is the threshold for entity-level SC score. (2) *Sample-level threshold filtering*, which drops the sample x_i if $c_i < Th\_sample$, where Th_sample is the threshold for sample-level SC score. (3) *Two-stage majority voting* (Xie et al., 2023), is an entity-level selection method, which first votes for the most consistent entity spans, then the most consistent types based on the voted spans.

2.3 Step 3: Inference with Self-Annotated Demonstration

When a test input x^q arrives, we retrieve k demonstrations from the reliable self-annotated dataset to help the inference. [2] We investigate the following four methods for demonstration retrieval. (1) *Random retrieval*, which randomly select k demonstrations. (2) *Nearest retrieval*, which select the k nearest neighbors of x^q. The distance of samples is measured by the cosine similarity in the representation space. (3) *Diverse nearest retrieval*, which first retrieve K nearest neighbors, where $K > k$, then uniformly samples a random set of k samples from the K neighbors. (4) *Diverse nearest with SC*

[2]Different from Lyu et al. (2023), our demonstrations are obtained through self-annotating with LLMs instead of randomly assignment. Besides, randomly assigning label is not feasible for NER task as it naturally requires label information on each token.

Method	CoNLL03	ACE05	WikiGold	GENIA	Avg
No-demos	68.97 $_{0.22}$	27.29 $_{0.58}$	70.8	47.41 $_{0.29}$	53.62
ICL with self-annotated demonstrations (Zero-shot)					
Without annotation selection					
Random	71.45 $_{0.10}$	30.38 $_{0.93}$	70.51	48.78 $_{0.06}$	55.28
Nearest	72.07 $_{0.11}$	**32.20** $_{0.92}$	**71.81**	49.54 $_{1.88}$	**56.40**
Diverse Nearest, random	**72.15** $_{0.65}$	31.07 $_{1.45}$	70.72	**50.01** $_{1.20}$	55.99
Entity-level threshold filtering					
Random	70.91 $_{0.55}$	30.41 $_{0.95}$	72.33	50.70 $_{1.53}$	56.09
Nearest	73.24 $_{0.53}$	32.22 $_{0.38}$	72.53	49.85 $_{1.20}$	56.96
Diverse Nearest, random	74.11 $_{0.12}$	**32.29** $_{0.31}$	72.01	50.68 $_{0.14}$	57.27
Diverse Nearest, SC ranking	**74.99** $_{0.20}$	31.65 $_{0.97}$	**73.53**	**51.11** $_{0.28}$	**57.82**
Sample-level threshold filtering					
Random	72.41 $_{1.28}$	30.00 $_{1.26}$	**73.38**	51.61 $_{1.21}$	56.86
Nearest	72.28 $_{0.14}$	**32.00** $_{0.08}$	73.27	**52.72** $_{0.80}$	**57.57**
Diverse Nearest, random	72.32 $_{0.08}$	30.74 $_{0.06}$	72.09	52.50 $_{0.50}$	56.91
Diverse Nearest, SC ranking	**73.97** $_{0.12}$	31.08 $_{0.54}$	72.80	51.67 $_{0.93}$	57.38
Two-stage majority voting					
Random	72.12 $_{0.59}$	31.18 $_{0.38}$	72.32	50.17 $_{0.93}$	56.45
Nearest	71.66 $_{0.37}$	31.45 $_{1.32}$	72.84	50.19 $_{1.59}$	56.53
Diverse Nearest, random	72.45 $_{0.41}$	30.84 $_{0.56}$	70.83	51.03 $_{0.73}$	56.28
Diverse Nearest, SC ranking	**74.51** $_{0.03}$	**32.27** $_{0.25}$	**73.98**	**52.06** $_{0.09}$	**58.20**
ICL with gold labeled demonstrations					
Random (Gold)	78.36 $_{0.31}$	42.12 $_{0.30}$	74.27	54.50 $_{1.14}$	62.31
Nearest (Gold)	84.30 $_{0.39}$	52.72 $_{0.44}$	78.20	54.78 $_{0.94}$	67.50
Random (Gold), full data	78.35 $_{1.44}$	41.33 $_{0.79}$	78.47	52.77 $_{2.03}$	62.73
Nearest (Gold), full data	83.51 $_{0.02}$	55.54 $_{0.61}$	79.73	58.72 $_{1.52}$	69.37

Table 1: Main results. The right subscript number are standard deviations. *Gold* indicates the method has access to the gold labeled data, thus is not comparable with the rest of methods. *Full data* indicates the method has access to the full training set. Results of $Th\_entity = 4.0$ and $Th\_sample = 4.0$ is shown here. Texts in **bold** are the best results in each category; Text underlined are the best results among all methods. The proposed framework significantly improves the zero-shot performances. On average, two-stage majority voting combined with the proposed diverse nearest with SC ranking achieves the best results.

ranking, proposed by this work to achieve a better trade-off between the similarity, diversity and reliability of self-annotated demonstrations. After retrieving K nearest neighbors, we select samples with the top-k sample-level SC scores.

Let $S = \{x_i, y_i\}_{i=1}^{k}$ denotes the self-annotated demonstrations retrieved for the test input x^q. Finally, our framework conduct ICL by concatenating these k samples as well as the test input sentence x^q, as shown in the below part in Fig. 1. The prediction is obtained via $y^q = \arg\max_y P(y|T, S, x^q)$.

3 Experiment

3.1 Setup

We experiment on four widely-used NER datasets, CoNLL03 (Sang and De Meulder, 2003), ACE05 (Walker et al., 2006), WikiGold (Balasuriya et al., 2009) and GENIA (Ohta et al., 2002). We use GPT3.5 (gpt-3.5-turbo) as the LLM backbone and text-embedding-ada-002 model to get sentence representations.[3] We set $k = 16$ and $K = 50$. For SC, we set temperature to 0.7 and sample 5 answers. For cost saving, we randomly sample 300 test samples twice then report the means and standard deviations, and we randomly sample 500 training samples without labels to form the unlabeled corpus $\mathcal{U}$. The naive zero-shot prompting is our baseline, which we denote as *No-demos*. We report F1 scores throughout this paper.

3.2 Results

The main results are shown in Table 1. Results of other values for thresholds Th_entity and Th_sample can be found in Appendix E. (1) Without annotation selection, we only generate one answer for each unlabeled sample. The results show improvements over *No-demos*, revealing that our framework is helpful even without any carefully designed annotation selection step. (2) The performance is further improved under three annotation selection strategies respectively. (3) The proposed diverse nearest with SC ranking shows consistent improvements under various settings and achieves the best results when combined with two-stage majority voting. This confirms that this strategy

[3]The results of GPT-3.5 are obtained during October and November 2023 with official API.

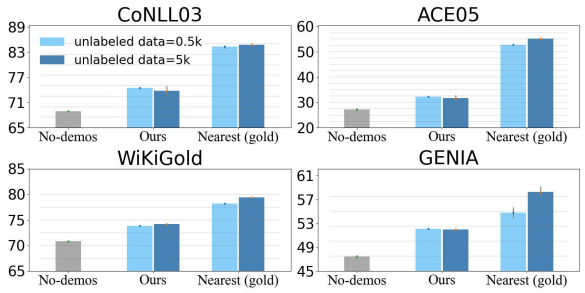

Figure 2: Results of increasing the size of unlabeled dataset. Vertical axes represent F1 scores. *Ours* refers to the combination of two-stage majority voting and diverse nearest with SC ranking. Increasing unlabeled data does not guarantee performance gains.

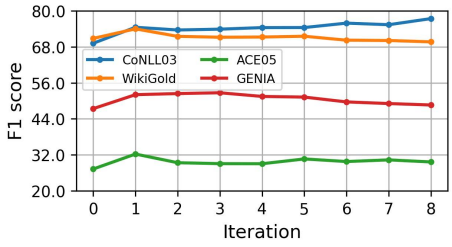

Figure 3: Increasing the iterations of self-improving does not guarantee performance improvements.

achieves a better trade-off between similarity, diversity and reliability of the demonstrations. (4) Random retrieval lags behind nearest retrieval in self-improving scenario but is not as much as in the gold label scenario, likely because of the noise contained in self-annotated labels. The model may directly copy the wrong answers in the most similar self-annotated demonstrations due to the copy mechanism of ICL (Lyu et al., 2023).

3.3 Analysis

Increasing unlabeled data. We expanded the size of $\mathcal{U}$ by 10 times and randomly sampled 5000 samples from the original training set. Results are shown in Fig. 2. Increasing the size of the unlabeled corpus does not guarantee performance improvements under the self-improving scenario. Meanwhile, increasing the size of the demonstration pool only brings marginal improvement, even under the gold label scenario. The reason may be that the small dataset already approximately captures the data distribution.

Iterative self-improving. We use the self-annotated data as demonstrations to guide the next iteration of self-annotating, forming a bootstrapping process. The illustration of iterative self-improving process can be found in Appendix G.

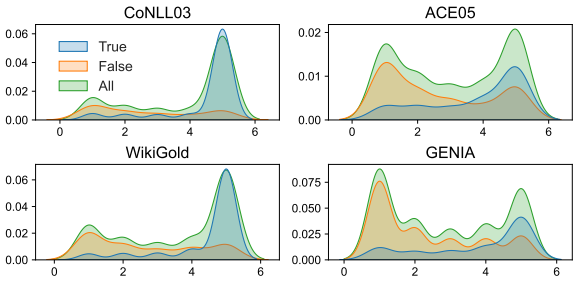

Figure 4: Kernel density estimation for SC scores. Vertical axes represent density, horizontal axes represent SC scores.

Method	CoNLL03	ACE05	WikiGold	GENIA	Avg
No-demos	68.97	27.29	70.8	47.41	53.62
TSMV	74.51	32.27	73.98	52.06	58.20
Upper bound	81.65	37.82	76.57	56.24	63.07
Gold label	84.30	52.72	78.20	54.78	67.50

Table 2: Results of the upper bound of reliable annotation selection. *TSMV* represents two-stage majority voting. We display the best results for each strategy. The setting of *Upper bound* performs on par with the setting of *Gold label*, showing that there might be space to be improved for reliable annotation selection.

We experiment up to 8 iterations. The 0-th iteration indicates the *No-demos* setting. Results are shown in Fig. 3. Increasing iterations of self-improving cannot guarantee improvements on most datasets. This may due to the fact that error accumulation in self-annotating is difficult to be eliminated in this training-free process.

Upper bound of reliable annotation selection. We keep only the true predictions and discard the false predictions in all the sampled answers to evaluate the upper bound of reliable annotation selection. Results are shown in Table 2. More detailed results can be found in Appendix F. *Upper bound* setting performs on par with the *Gold label* setting, indicating that there might still be space to be improved for reliable annotation selection.

SC score analysis. We plot the kernel density estimation for entity-level SC scores in Fig. 4. Most true predictions gather in the interval of high SC scores, while most false predictions have low SC scores. This shows that SC scores effectively reflect the reliability of annotations.

Self-verification. Besides SC, we also explore self-verification (SV) to measure the confidence of self-annotation by asking the LLM to score its own answer about its own confidence. After the LLM outputs the recognized entities, we obtain the

Method	SC	SV
No-demos	68.97 $_{0.22}$	
Entity-level threshold filtering		
Random	70.91 $_{0.55}$	70.91 $_{0.56}$
Nearest	73.24 $_{0.53}$	71.23 $_{0.01}$
Diverse Nearest, random	74.11 $_{0.12}$	**71.44** $_{0.93}$
Diverse Nearest, score ranking	**74.99** $_{0.20}$	68.09 $_{0.60}$
Sample-level threshold filtering		
Random	72.41 $_{1.28}$	**71.00** $_{0.32}$
Nearest	72.28 $_{0.14}$	70.45 $_{0.46}$
Diverse Nearest, random	72.32 $_{0.08}$	70.06 $_{1.29}$
Diverse Nearest, score ranking	**73.97** $_{0.12}$	68.95 $_{0.35}$

Table 3: Comparison between SC and SV on CoNLL03 dataset. $Th\_entity = 4.0$ and $Th\_sample = 4.0$ is used. Right subscript number are standard deviations. Texts in **bold** are the best results in each category; Text underlined are the best results among all methods.

Method	CoNLL03	WikiGold
No-demos	42.24	28.57
Nearest	23.55	8.94

Table 4: Results on the Llama2 chat 13B. Two-stage majority voting is used here. The negative results show that the proposed framework is more suitable for models with a strong zero-shot capability. The negative effect is obvious on the first sampled test set, thus we do not continue to test on other seeds.

SV score by asking the LLM: "How confident are you in providing the above answers? Please give each named entity in your answer a confidence score of 0-5." The comparison results between SC and SV are in Table 3. As shown in the table, SV also achieves some improvements compared with the No-demos baseline. However, it lags behind the SC measurement. This is presumably because the LLM tends to be over-confident about its own answer, since we found that no sample gets a confidence score lower than 3 under the SV measurement in CoNLL03 benchmark. The overconfidence problem is also mentioned in Li et al. (2023a).

Evaluation on weaker LLMs. To explore the performance of the proposed self-improving framework on weaker LLMs, we conduct experiments on the Llama2 chat 13B model (Touvron et al., 2023),[4] the results are shown in Table 4. Two-stage majority voting selection strategy and the nearest neighbor retrieval method are used in this experiment. With a much weaker ability in zero-shot scenarios, Llama2 13B model shows negative results under the self-improving framework. This indicates that the proposed framework is more suitable for models with a strong zero-shot capability. For the models with a relatively weaker zero-shot ability, improving the prompt designing might be a more effective strategy to boost performance.

[4] https://huggingface.co/meta-llama/Llama-2-13b-chat-hf

4 Related Work

Information extraction with LLM. The research of information extraction (IE) with LLMs includes prompt designing (Wei et al., 2023b; Wang et al., 2023; Xie et al., 2023; Li et al., 2023b), task-specific LLMs instruction-tuning (Zhou et al., 2023; Sainz et al., 2023) and data augmentation (Zhang et al., 2023; Ma et al., 2023; Josifoski et al., 2023). Zhang et al. (2023) use LLM to annotate data, which is used to fine-tune a specific IE model, then the fine-tuned model is used to help select the data to be annotated in the next iteration. Unlike previous works, this work propose a training-free self-improving framework to push the zero-shot boundary of LLM on NER. Different from Zhang et al. (2023), no seed labeled data, expert small model nor training resources are required in our framework. In addition, our work is **orthogonal** to previous prompt designing works. They explored various advanced prompt formats to boost performance, and did not utilize unlabeled corpus. Unlike them, this work improves zero-shot NER by using unlabeled corpus without designing any complex prompt format.

Demonstrations in ICL. Some works explored factors that have impacts on ICL (Lyu et al., 2023; Min et al., 2022; Wei et al., 2023a). Lyu et al. (2023) investigate the impact of randomly assigning labels to demonstrations in ICL. However, this random labeling method is not suitable for tasks like NER, which requires label information on the token-level instead of sentence-level. Different from them, we first use LLM to make predictions on the unlabeled corpus, then select reliable self-annotated data as demonstrations.

5 Conclusion

We propose a training-free self-improving framework for zero-shot NER with LLMs, which achieves significant performance improvements on four benchmarks. Comprehensive experimental analysis shows that, simply increasing the size of unlabeled corpus or the iterations of self-annotation do not guarantee further improvement, but there might still be room for improvement with more advanced strategies for reliable annotation selection.

Limitations

We acknowledge the following limitations of this study.

- This work focus on exploring the zero-shot self-improving framework on NER task. The investigation of this paradigm on other IE tasks are not studied yet.
- We explored the commonly-used self-consistency and the self-verification method to obtain the confidence score for measuring the quality of self-annotated data. There might be other approaches to measure the quality of self-annotation.
- The zero-shot performance still lag behind previous state-of-the-art of fully-supervised methods.
- Although this framework achieves significant improvement on the strong LLM, GPT-3.5, it gets negative results on a much weaker LLM, Llama2 13B. Improving the zero-shot NER on the weaker and smaller LLMs remains to be explored.

Acknowledgements

This research is supported by Zhejiang Provincial Natural Science Foundation of China (LDT23F02023F02). We would like to thank the anonymous reviewers for their insightful comments and constructive suggestions. We would also like to thank Chen Wang and Xinlong Qiao for their help at the visualization.

References

Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R Curran. 2009. Named entity recognition in wikipedia. In *Proceedings of the 2009 workshop on the people's web meets NLP: Collaboratively constructed semantic resources (People's Web)*, pages 10–18.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *arXiv preprint arXiv:2305.14450*.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting asymmetry for synthetic training data generation: Synthie and the case of information extraction.

Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023a. Evaluating chatgpt's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness. *arXiv preprint arXiv:2304.11633*.

Peng Li, Tianxiang Sun, Qiong Tang, Hang Yan, Yuanbin Wu, Xuanjing Huang, and Xipeng Qiu. 2023b. CodeIE: Large code generation models are better few-shot information extractors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15339–15353, Toronto, Canada. Association for Computational Linguistics.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023c. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. 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 3036–3046, Minneapolis, Minnesota. Association for Computational Linguistics.

Xinxi Lyu, Sewon Min, Iz Beltagy, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Z-ICL: Zero-shot in-context learning with pseudo-demonstrations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2304–2317, Toronto, Canada. Association for Computational Linguistics.

Mingyu Derek Ma, Xiaoxuan Wang, Po-Nien Kung, P. Jeffrey Brantingham, Nanyun Peng, and Wei Wang. 2023. Star: Improving low-resource information extraction by structure-to-text data generation with large language models.

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? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tomoko Ohta, Yuka Tateisi, Jin-Dong Kim, Hideki Mima, and Junichi Tsujii. 2002. The genia corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the human language technology conference*, pages 73–77. Citeseer.

OpenAI. 2022. Introducing chatgpt.

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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2023. Gollie: Annotation guidelines improve zero-shot information-extraction.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus ldc2006t06, 2006. *URL https://catalog. ldc. upenn. edu/LDC2006T06*.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. 2023a. Larger language models do in-context learning differently.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023b. Zero-shot information extraction via chatting with chatgpt.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical study of zero-shot ner with chatgpt.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023. Llmaaa: Making large language models as active annotators.

Suxiang Zhang, Ying Qin, Juan Wen, and Xiaojie Wang. 2006. Word segmentation and named entity recognition for SIGHAN bakeoff3. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 158–161, Sydney, Australia. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2023. Universalner: Targeted distillation from large language models for open named entity recognition.

A Dataset Statistics

We evaluate on four commonly-used NER English datasets, CoNLL03 (Sang and De Meulder, 2003), ACE05 (Walker et al., 2006), WikiGold (Balasuriya et al., 2009) and GENIA (Ohta et al., 2002), among which CoNLL03, WikiGold and GENIA are public datasets, and ACE05[5] can be accessed on Linguistic Data Consortium (LDC) platform with specific license. In addition, we also evaluate on two Chinese datasets, Ontonotes 4[6] and MSRA (Zhang et al., 2006), in Appendix B. Table 5 and 6 shows the statistics of the processed datasets used in this work. For CoNLL03, we use the processed version shared by Han et al. (2023). For ACE05, we follow Luan et al. (2019)'s processing steps.

Dataset	CoNLL03	ACE05	WikiGold	GENIA
#Train	14382	12475	1422	16692
#Test	3453	2050	274	1854

Table 5: Statistics of the processed English datasets used in this work. The training set is formed by combining the original training split and development split.

Dataset	Ontonotes 4	MSRA
#Train	20025	46364
#Test	4346	4365

Table 6: Statistics of the processed Chinese datasets used in this work. The training set is formed by combining the original training split and development split.

B Results on Additional Benchmarks

We additionally evaluate on two widely-used Chinese benchmarks, the results are in Table 7.

Method	Ontonotes 4	MSRA
No-demos	31.71 $_{1.14}$	39.21 $_{0.93}$
ICL with self-annotated demonstrations		
Random	32.45 $_{0.19}$	39.55 $_{0.75}$
Nearest	31.54 $_{1.60}$	36.31 $_{1.76}$
Diverse Nearest, SC ranking	**35.57** $_{1.22}$	**40.84** $_{2.83}$
ICL with gold labeled demonstrations		
Random (Gold)	49.42 $_{0.22}$	53.51 $_{1.38}$
Nearest (Gold)	64.16 $_{1.08}$	61.58 $_{1.58}$

Table 7: Results on Chinese benchmarks. Right subscript numbers are standard deviations. *Gold* indicates access to the gold labeled data, thus is not comparable with the rest of methods. Two-stage majority voting is used here. Texts in **bold** are the best results.

[5] https://catalog.ldc.upenn.edu/LDC2006T06

[6] https://catalog.ldc.upenn.edu/LDC2011T03

C Results on Other Embedding Models

We explore the effect of using other embedding models for retrieval, SBERT (Reimers and Gurevych, 2019)[7] and GTE (Li et al., 2023c)[8]. Results are in Table 8.

D Results on Various Number of Demonstrations

We investigate the performance on various number of demonstrations in the input context, the results are in Table 9. As shown in the table, the quantity of examples is not always proportional to the final performance. Similar findings have also been mentioned in Min et al. (2022). We hypothesize that after the LLM learns the mapping between the input-output examples, new information gained from more examples is marginal and might be offset by the more noise introduced.

E More Results on Threshold Filtering

Table 10 shows the results of various values of entity-level and sample-level SC thresholds.

F Upper Bound of Reliable Annotation Selection

Table 11 summarizes the complete results of the upper bound of reliable annotation selection.

G Illustration of Iterative Self-improving

The bootstrapping process of iterative self-improving is shown in Fig. 5.

H Case Study

We take a closer look at the cases where the errors in predictions are corrected with self-annotated demonstrations, as shown in Fig. 6. The proposed framework makes the model reuse its own knowledge and correct its own errors, forming a process of self-improving.

I Prompts

We show the prompts use in this work in Table 12. We take samples from ACE05 for demonstrations.

[7] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

[8] https://huggingface.co/thenlper/gte-large

Datasets	**CoNLL2003**			**WikiGold**		
Embedding Models	embed-ada	SBERT	GTE	embed-ada	SBERT	GTE
No-demos	68.97 $_{0.22}$	68.97 $_{0.22}$	68.97 $_{0.22}$	70.80	70.80	70.80
ICL with self-annotated demonstrations (Zero-shot)						
Random	72.12 $_{0.59}$	72.12 $_{0.59}$	72.12 $_{0.59}$	72.32	72.32	72.32
Nearest	71.66 $_{0.37}$	72.07 $_{0.22}$	72.37 $_{1.17}$	72.84	72.39	72.24
Diverse Nearest, SC ranking	**74.51** $_{0.03}$	**72.67** $_{0.37}$	**72.53** $_{0.96}$	**73.98**	**76.08**	**73.60**
ICL with gold labeled demonstrations						
Random (Gold)	77.25 $_{1.39}$	77.25 $_{1.39}$	77.25 $_{1.39}$	75.82	75.82	75.82
Nearest (Gold)	84.71 $_{0.39}$	83.28 $_{1.34}$	83.59 $_{0.09}$	79.40	78.18	79.03

Table 8: Results on various embedding models. Right subscript numbers are standard deviations. *embed-ada* refers to text-embedding-ada. *Gold* indicates access to the gold labeled data, thus is not comparable with the rest of methods. Two-stage majority voting is used here. Texts in **bold** are the best results.

Numbe of demonstrations	**0**	**2**	**4**	**8**	**16**	**32**
WikiGold						
Random	70.80	70.25	70.86	**71.74**	71.39	70.35
Nearest	70.80	70.41	71.32	70.47	**72.57**	71.81
Random (Gold)	70.80	71.75	71.54	**75.79**	73.95	74.43
Nearest (Gold)	70.80	76.14	77.66	**78.97**	78.34	77.05
CoNLL03						
Random	68.97	69.54	70.84	70.53	70.72	**71.95**
Nearest	68.97	70.12	69.15	70.90	71.81	**72.44**
Random (Gold)	68.97	71.94	72.76	75.12	77.81	**80.43**
Nearest (Gold)	68.97	79.07	80.81	83.20	**84.12**	83.94

Table 9: Results on various number of demonstrations in the input context. *Gold* indicates access to the gold labeled data, thus is not comparable with the rest of methods. Two-stage majority voting is used here. Texts in **bold** are the best results. Since the standard deviation values of CoNLL03 are around the same level as in Table 1, we omit them here.

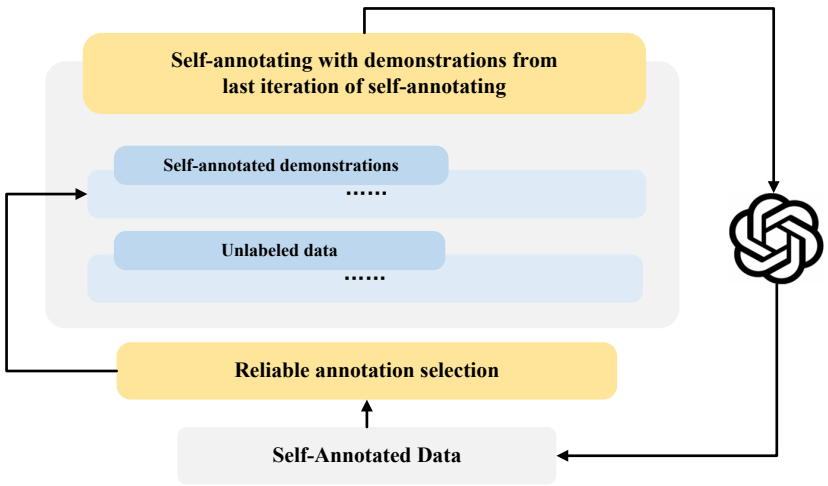

Figure 5: The pipeline of iterative self-improving.

Input Sentence: Angelo has reached out to corporate America , the young and successful , the trendy , ···
Gold Label: [{'America': 'Location', 'Angelo': 'Person'}].
Self-annotated demonstrations:
Text: The Anguilla United Front is an alliance of political parties in Anguilla .
Answer: [{'Anguilla United Front': 'Organization'}, {'Anguilla': 'Location'}]
……
No-demos pred.: [{'Angelo': 'Person'}].
Ours pred.: [{'America': 'Location', 'Angelo': 'Person'}].

Input Sentence: Ben now also helps run Movement Bodyboarding MagAzine.
Gold Label: [{'Movement Bodyboarding Magazine': 'Organization', 'Ben': 'Person'}].
Self-annotated demonstrations:
Text: Bobick had now improved enough as a boxer to be a legitimate title threat .
Answer: [{'Bobick': 'Person'}]
……
No-demos pred.: [{'Movement Bodyboarding Magazine': 'Organization'}].
Ours pred.: [{'Ben': 'Person', 'Movement Bodyboarding Magazine': 'Organization'}].

Figure 6: Case study of self-improving. Examples from WikiGold are illustrated. The errors in predictions of *No-demos* are corrected by our framework with self-annotated demonstrations. Texts in green are entities corrected by our method. Texts in blue are entities in demonstrations that potentially help with the error correction.

Method	**CoNLL03**	**ACE05**	**WikiGold**	**GENIA**	**Avg**
No-demos	68.97 $_{0.22}$	27.29 $_{0.58}$	70.8	47.41 $_{0.29}$	53.62
Entity-level SC threshold = 3.0					
Random	71.17 $_{0.13}$	30.16 $_{0.66}$	71.79	50.41 $_{0.00}$	55.88
Nearest	71.41 $_{0.66}$	31.58 $_{0.76}$	73.16	51.24 $_{1.79}$	56.85
Diverse Nearest, random	72.68 $_{1.31}$	31.39 $_{1.62}$	72.01	50.65 $_{0.11}$	56.68
Diverse Nearest, SC ranking	73.68 $_{0.03}$	31.86 $_{0.13}$	73.36	51.15 $_{0.69}$	57.51
Entity-level SC threshold = 4.0					
Random	70.91 $_{0.55}$	30.41 $_{0.95}$	72.33	50.70 $_{1.53}$	56.09
Nearest	73.24 $_{0.53}$	32.22 $_{0.38}$	72.53	49.85 $_{1.20}$	56.96
Diverse Nearest, random	74.11 $_{0.12}$	32.29 $_{0.31}$	72.01	50.68 $_{0.14}$	57.27
Diverse Nearest, SC ranking	74.99 $_{0.20}$	31.65 $_{0.97}$	73.53	51.11 $_{0.28}$	57.82
Entity-level SC threshold = 5.0					
Random	72.53 $_{0.07}$	29.44 $_{0.73}$	72.13	50.65 $_{0.57}$	56.18
Nearest	74.24 $_{0.03}$	29.65 $_{1.30}$	72.45	48.12 $_{0.45}$	56.11
Diverse Nearest, random	73.50 $_{0.14}$	30.55 $_{0.27}$	71.34	49.34 $_{0.27}$	56.18
Diverse Nearest, SC ranking	72.50 $_{0.66}$	30.14 $_{0.35}$	74.01	49.57 $_{0.61}$	56.55
Sample-level SC threshold = 3.0					
Random	70.17 $_{0.00}$	28.78 $_{1.71}$	71.81	50.45 $_{0.34}$	55.30
Nearest	69.48 $_{0.90}$	30.39 $_{0.17}$	70.33	51.76 $_{0.29}$	55.49
Diverse Nearest, random	68.98 $_{0.86}$	30.04 $_{0.34}$	69.71	51.71 $_{1.41}$	55.11
Diverse Nearest, SC ranking	74.32 $_{1.37}$	30.73 $_{0.04}$	74.44	52.31 $_{0.34}$	57.95
Sample-level SC threshold = 4.0					
Random	72.41 $_{1.28}$	30.05 $_{1.26}$	73.38	51.61 $_{1.21}$	56.86
Nearest	72.28 $_{0.14}$	32.00 $_{0.08}$	73.27	52.72 $_{0.80}$	57.57
Diverse Nearest, random	72.32 $_{0.08}$	30.74 $_{0.06}$	72.09	52.50 $_{0.50}$	56.91
Diverse Nearest, SC ranking	73.97 $_{0.12}$	31.08 $_{0.54}$	72.80	51.67 $_{0.93}$	57.38
Sample-level SC threshold = 5.0					
Random	73.66 $_{0.69}$	29.19 $_{0.26}$	71.92	51.34 $_{0.97}$	56.52
Nearest	74.19 $_{0.30}$	30.94 $_{0.11}$	74.96	52.01 $_{0.23}$	58.02
Diverse Nearest, random	73.16 $_{0.66}$	27.98 $_{0.08}$	74.55	50.64 $_{0.18}$	56.58
Diverse Nearest, SC ranking	74.53 $_{0.51}$	30.00 $_{0.73}$	73.60	51.02 $_{0.98}$	57.28

Table 10: Results of various entity-level SC thresholds and sample-level SC thresholds. Right subscript numbers are standard deviations.

Method	CoNLL03	ACE05	WikiGold	GENIA	Avg
No-demos	68.97 $_{0.22}$	27.29 $_{0.58}$	70.8	47.41 $_{0.29}$	53.62
Two-stage majority voting					
Random	72.12 $_{0.59}$	31.18 $_{0.38}$	72.32	50.17 $_{0.93}$	56.45
Nearest	71.66 $_{0.37}$	31.45 $_{1.32}$	72.84	50.19 $_{1.59}$	56.53
Diverse Nearest, random	72.45 $_{0.41}$	30.84 $_{(0.56}$	70.83	51.03 $_{0.73}$	56.28
Diverse Nearest, SC ranking	74.51 $_{0.03}$	32.27 $_{0.25}$	73.98	52.06 $_{0.09}$	58.20
Upper bound					
Random	73.72 $_{0.41}$	32.71 $_{0.56}$	73.83	52.67 $_{0.09}$	58.23
Nearest	81.65 $_{0.17}$	37.82 $_{0.59}$	76.57	56.24 $_{0.44}$	63.07
Diverse Nearest, random	78.84 $_{1.43}$	35.79 $_{0.26}$	76.20	54.46 $_{0.98}$	61.32
Diverse Nearest, SC ranking	80.12 $_{0.02}$	35.23 $_{0.63}$	76.64	54.58 $_{0.57}$	61.64

Table 11: Complete results of the upper bound of reliable annotation selection. Right subscript numbers are standard deviations.

Prompts of zero-shot setting
Given entity label set: ['Person', 'Organization', 'Location', 'Facility', 'Weapon', 'Vehicle', 'Geo-Political Entity']. Please recognize the named entities in the given text. Based on the given entity label set, provide answer in the following JSON format: [{'Entity Name': 'Entity Label'}]. If there is no entity in the text, return the following empty list: []. Text: right now we 're also waiting to hear from the president at the white house . Answer:
Prompts of ICL
Given entity label set: ['Person', 'Organization', 'Location', 'Facility', 'Weapon', 'Vehicle', 'Geo-Political Entity']. Please recognize the named entities in the given text. Based on the given entity label set, provide answer in the following JSON format: [{'Entity Name': 'Entity Label'}]. If there is no entity in the text, return the following empty list: []. Text: right now we 're also waiting to hear from the president at the white house . Answer: [{'white house': 'Location'}, {'president': 'Person'}] Text: At the Pentagon , Barbara Starr reports officials say today begins a new strategy in the skies over Baghdad . Answer: [{'Barbara Starr': 'Person'}, {'Pentagon': 'Facility'}, {'officials': 'Person'}, {'skies':'Location'}, {'Baghdad': 'Geo-Political Entity'}] Text: John Irvine , ITV News , Baghdad . Answer: [{'John Irvine': 'Person'}, {'ITV News': 'Organization'}, {'Baghdad': Geo-Political Entity'}] ... ... Text: right now we 're also waiting to hear from the president at the white house . Answer:

Table 12: Prompts used in this work. A few samples from ACE05 are displayed for demonstrations.