# LSP: Empowering Few-Shot NER with Demonstration Augmentation via Label Subset Partition

**Anonymous ACL submission**

## Abstract

Leveraging the strong generalization capabilities of Large Language Models (LLMs) for data augmentation is an effective means to address the data sparsity of few-shot named entity recognition (FS-NER). Typically, existing methods manage to select appropriate demonstrations from a large amount of labeled data to be filled into the context of LLMs, thereby significantly enhancing the ability for in-context learning (ICL) in FS-NER. However, on the one hand, we have not yet figured out how demonstrations affect ICL in FS-NER so that we cannot do targeted optimization. On the other hand, labeled data is not abundant to select demonstrations from in real low-resource scenarios. In this study, we first systematically explore the impact of demonstrations on the ICL for FS-NER from 5 perspectives: sentence inclusion, number of demonstrations, label accuracy, label diversity, and label coverage. We find that label diversity and label coverage are important factors for ICL in FS-NER. So, we propose three metrics to quantify them: Label Space Per Instance (LSPI), Label Coverage (LC), and Label Measure(LM). Second, focusing on improving LSPI, LC, and LM, we devise a method named label subset partition (LSP) to augment demonstrations. It's an out-of-the-box augmentation method which is training-free, prompt-agnostic, and model-agnostic. Experiments on extensive NER datasets have demonstrated that LSP can effectively improve the performance of ICL for FS-NER.

## 1 Introduction

Named entity recognition (NER) aims to recognize pre-defined named entities in unstructured text, which is a fundamental task for other NLP (Natural Language Processing) downstream applications like information retrieval (IE) and question answering (QA). Due to the high labor cost of high-quality labeled data, NER technology in low-resource scenarios (or FS-NER) has been widely explored, particularly in recent years (Huang et al., 2021; Huang et al., 2022; Moscato et al., 2023). Thanks to the abundant pre/post-trained knowledge, the in-context learning (ICL) ability has been observed in large language models (LLMs) (Dong et al., 2024) and widely explored in FS-NER (Santoso et al., 2024; Zhang et al., 2023).

**# Instruction**
You are a professional and helpful crowdsourcing data annotator using English with the help of description of types.
Identify the entities and recognize their types in the sentence.
The output should be a string in the format of the tuple list, like'[(type 0, entity 0), (type 1, entity 1), ...]'.
**# types**
1) PER, indicates person...
2) ORG, indicates organization...
3) LOC, indicates location...
4) MISC, indicates miscellaneous...
**# demonstrations**
1) Sentence: Good news for Milan is that Udinese's German striker Oliver Bierhoff is out through injury.
Output: [('Milan', 'ORG'), ('Udinese', 'ORG'), ('German', 'MISC'), ('Oliver Bierhoff', 'PER')]
2) Sentence: Only France and Britain backed Fischler's proposal.
Output: [('France' , 'LOC'), ('Fischler', 'PER')]
...
**# Query**
Sentence: EU rejects German call to boycott British lamb.
Output:

Figure 1: The prompt template for FS-NER. **Instruction** zone is used to describe tasks. **Type** zone illustrates all the labels of the NER task. **Demonstration** zone shows some demonstrations for reference. **Query** zone is the target instance that needs to be annotated.

Compared to the zero-shot setting, performances of structured prediction like NER can be greatly improved in ICL by filling demonstrations into the context window of LLMs as references (as shown in Figure 1) under few-shot settings (Han et al., 2024; Han et al., 2024). How do demonstrations boost ICL? Min et al. (2022) have explored the role of demonstrations in ICL on classification and multi-choice tasks (e.g., sentiment analysis and question answering). They have identified that the label space, the distribution of the input text, and the format of the input-label pairs are crucial learning signals provided by demonstrations for ICL. However, unlike classification and multi-choice tasks, structured prediction tasks have complex output space and they are enhanced with the help of structure information in the input (Dev et al., 2021). Therefore, we cannot easily generalize the findings from Min et al. (2022) to structured prediction tasks. In this work, we take the FS-NER task as an example of structured prediction tasks. We manage to systematically explore the impact of demonstrations on the ICL for FS-NER, so as to do targeted optimization[1] for ICL on FS-NER and provide motivation for future works.

In Section 3, we conduct explorations from 5 aspects: sentence inclusion, number of demonstrations, label accuracy, label diversity, and label coverage. In addition, we introduce 3 novel metrics to measure label diversity and label coverage: Label Space Per Instance (LSPI), Label Coverage (LC), and Label Measure (LM). It should be noted that LM is a metric that combines LSPI and LC, which has a high correlation with the micro-F1 score. Our experiments indicate that an appropriate number of demonstrations, accurate labels, diverse labels, and labels with high coverage to the test set are essential for ICL in FS-NER.

Based on the above conclusion, we propose Label Subset Partition (LSP) in Section 4 to augment demonstrations to improve label diversity and label coverage when keeping an appropriate number of accurate demonstrations. LSP augments demonstrations by decomposing the original labels into different label subsets, allowing demonstrations with original labels to be transformed into multiple copies with different label subsets. Furthermore, it's an out-of-the-box demonstration augmentation method which is training-free, prompt-agnostic, and model-agnostic. Experiments show that LSP can improve LM so that it can improve ICL ability on FS-NER.

To sum up, our contributions include: (1) To the best of our knowledge, we investigate factors of demonstrations that matter for ICL on FS-NER for the first time. (2) We observe that the label diversity and the label coverage are crucial for ICL in FS-NER. Meanwhile, we devise 3 metrics (i.e., LSPI, LC, and LM) to measure the label diversity and the label coverage. (3) We propose LSP, an out-of-the-box demonstration augmentation method, to improve LM and the ICL performance on FS-NER.

[1]Targeted optimization means designing optimization strategies directly based on the metrics that perform poorly in benchmarking (Qian et al., 2023).

## 2 Related Work

### 2.1 Few-shot NER

Few-shot NER (i.e., FS-NER) identifies entities using only a small number of labeled data (Moscato et al., 2023). Recent research can be roughly categorized into algorithm-based and data-based ones.

#### 2.1.1 Algorithm-based Methods

Algorithm-based methods primarily focus on how to construct and train models in few-shot settings to achieve high performance. They are typically grounded in transfer learning or meta-learning. **Transfer learning** is used to transfer knowledge from resource-rich domains(Zhang et al., 2024; Zhang et al., 2024), languages(Rahimi et al., 2019; Wang et al., 2022), and tasks (Radford et al.; Brown et al., 2020) to low-resource scenarios. Due to the extensive pre/post-training knowledge, pre-trained models (i.e., PTMs) and large language models (i.e., LLMs) are commonly employed as the backbone in transfer learning. For example, the In-Context Learning (i.e., ICL) capability of LLMs is leveraged to conduct FS-NER(Wang et al., 2023a; Wu et al., 2024) with suitable demonstrations retrieved from a large amount of labeled data. However, those methods contradict the real scene that there is only a small amount of labeled data available in low-resource scenarios. **Meta-learning** enables models to "learn how to learn", allowing models to rapidly adapt to new tasks with only a minimal number of data. For instance, Model-Agnostic Meta-Learning (i.e., MAML) (Li et al., 2022; Ma et al., 2022b) and Prototypical Networks (de Lichy et al., 2021; Tong et al., 2021).

#### 2.1.2 Data-based Methods

Data-based methods focus on how to manipulate data to increase the size of the available training corpora, in order to address the issue of data scarcity. These methods can be primarily categorized into four strategies: active learning, distant supervision, self-training, and data augmentation. **Active learning** is a strategy of selecting the most informative example for manual annotation, to balance model performance and annotation cost (Agrawal et al., 2021; Rouzegar and Makrehchi, 2024). **Distant supervision** methods leverage external resources, such as ontologies and knowledge bases, to generate weakly labeled examples from unannotated data or to identify potential entities through heuristic rules (Liang et al., 2020; Qu et al., 2023). **Self-**

**training** methods utilize the model's inherent capabilities to generate labels for unannotated data, subsequently employing these labels to further enhance the model (Fu et al., 2023; Xie et al., 2024). **Data augmentation** methods generate synthesized data to increase the available dataset by employing heuristic rules (Dai and Adel, 2020; Liu et al., 2021), PTMs (Liu et al., 2022; Song et al., 2024) or LLMs (Santoso et al., 2024; Xie et al., 2024). Here, our work is a data augmentation method that enhances the NER performance of LLMs by synthesizing higher-quality NER examples from the original labeled data.

### 2.2 Exploration on ICL

In-context learning (ICL) has been the focus of significant studies to utilize LLMs since its introduction (Sanh et al., 2022; Dong et al., 2024). It is widely used for various tasks especially in few-shot settings (Hu et al., 2022; Cahyawijaya et al., 2024). Some work has been done to understand why in-context learning works. For example, Xie et al. (2022) explains ICL as implicit Bayesian inference. Min et al. (2022) provides an empirical analysis that investigates why ICL works on 6 tasks (e.g., sentiment analysis and question answering) except for FS-NER. Thus, in this work, we especially explore why ICL is effective on FS-NER based on the demonstrations in the LLMs' context window.

## 3 Exploration on Demonstrations

So as to thoroughly investigate how demonstrations impact the performance of ICL on FS-NER, we conduct a series of experiments in this section from 5 aspects: sentence inclusion, number of demonstrations, label accuracy, label diversity, and label coverage. The experiment setup is detailed in Appendix A. As shown in Figure 1, a demonstration consists of a sentence and its corresponding output. The output should be recognized from the sentence during inference contains entity mentions (e.g., "Milan") and their labels (e.g., "ORG").

### 3.1 Sentence Inclusion

Intuitively, there must be a strong correlation between the sentence and its output in a demonstration, because the entity mentions and labels in the output are meaningful only when we consider the contextual semantics of the sentence. Nevertheless, how much does the sentence inclusion of demonstrations matter to ICL on FS-NER? We use the prompt template shown in Figure 1 and experiment with masked sentences in demonstrations by replacing the words with "***". In Table 1, we can see that the FS-NER performance of the LLMs does not decrease drastically even if the sentence is masked, and in some cases it even increases. Hence, we can draw a counterintuitive conclusion: sentence inclusion may not directly affect the effectiveness of demonstrations. The learning signal for ICL on FS-NER is mainly provided by the output (i.e., the pairs composed of entity mentions and labels).

### 3.2 Number of Demonstrations

According to previous works (Ma et al., 2023; Han et al., 2024; Wu et al., 2024) and Appendix C, the performance of FS-NER using $k$-shot settings usually improves with increasing $k$. By intuition, the larger $k$, the more demonstrations there are in the context window. Therefore, an intuitive question is: does simply duplicating demonstrations to increase the number of demonstrations improve ICL capability on FS-NER? We conduct a simple experiment to investigate the question by directly duplicating demonstrations $n$ times. Specifically, we first use Algorithm 2 (Ma et al., 2023) to sample $k$-shot instances as base demonstrations. Then, we duplicate them $n$ times and fill the duplicated demonstrations into the context window. As shown in Figure 2 and Figure 7, we can see that the FS-NER performance of Qwen (Bai et al., 2023) and DeepSeek (DeepSeek-AI, 2024) slightly improved compared to not duplicating when the number of duplications is within 2. However, duplicating demonstrations can cause fluctuations for Mixtral (Jiang et al., 2024) and ultimately lead to deterioration in most cases. This may be due to Mixtral's inability to handle constantly growing contexts. In summary, the results indicate that simply increasing the number of demonstrations does not consistently improve ICL ability on FS-NER.

### 3.3 Label Accuracy

Label accuracy of the output in a demonstration may potentially affect ICL ability on FS-NER, as incorrect labels introduce noise into the context, misleading LLMs with wrong learning signals. To validate such a hypothesis, we adjust the accuracy of the labels in demonstrations from 100% to 0% using a simple heuristic method shown in Algorithm 3. For example, when the label accuracy is 75%, 25% of entities (e.g., "Udinese" whose gold label is "ORG") in the output need to be randomly

| datasets | Onto5-EN | | Movie | | Onto5-ZH | | CMeEE-V2 | |
|---|---|---|---|---|---|---|---|---|
| **methods** | *k*=1 | *k*=5 | *k*=1 | *k*=5 | *k*=1 | *k*=5 | *k*=1 | *k*=5 |
| Qwen | $35.19_{\pm 1.48}$ | $38.48_{\pm 2.07}$ | $67.27_{\pm 1.79}$ | $64.68_{\pm 2.54}$ | $38.27_{\pm 4.59}$ | $40.48_{\pm 1.65}$ | $43.48_{\pm 0.94}$ | $42.79_{\pm 0.60}$ |
| w/ mask | $34.74_{\pm 1.34}$ | $39.49_{\pm 2.56}$ | $67.24_{\pm 0.92}$ | $64.00_{\pm 2.61}$ | $38.38_{\pm 2.54}$ | $34.63_{\pm 3.05}$ | $46.09_{\pm 1.00}$ | $45.91_{\pm 1.35}$ |
| Mixtral | $28.33_{\pm 1.00}$ | $19.08_{\pm 1.30}$ | $67.22_{\pm 2.17}$ | $71.03_{\pm 0.70}$ | $26.84_{\pm 1.57}$ | $10.28_{\pm 3.67}$ | $15.94_{\pm 2.41}$ | $31.05_{\pm 1.06}$ |
| w/ mask | $26.39_{\pm 2.95}$ | $16.52_{\pm 1.87}$ | $68.25_{\pm 1.47}$ | $71.02_{\pm 1.04}$ | $23.14_{\pm 4.41}$ | $19.47_{\pm 6.02}$ | $31.10_{\pm 1.71}$ | $29.00_{\pm 1.13}$ |
| DeepSeek | $58.37_{\pm 5.71}$ | $59.59_{\pm 3.91}$ | $76.48_{\pm 1.18}$ | $79.89_{\pm 2.08}$ | $59.39_{\pm 3.18}$ | $57.93_{\pm 2.58}$ | $52.53_{\pm 3.14}$ | $51.45_{\pm 1.89}$ |
| w/ mask | $55.56_{\pm 4.26}$ | $57.10_{\pm 3.40}$ | $73.07_{\pm 1.86}$ | $73.97_{\pm 1.64}$ | $54.49_{\pm 2.32}$ | $53.50_{\pm 3.28}$ | $47.21_{\pm 3.93}$ | $46.75_{\pm 2.47}$ |

Table 1: Micro-F1 (%) results w/o mask and w/ mask using different LLMs in (*k*=1, 5)-shot settings. Red represents degradation. Green represents an increase.

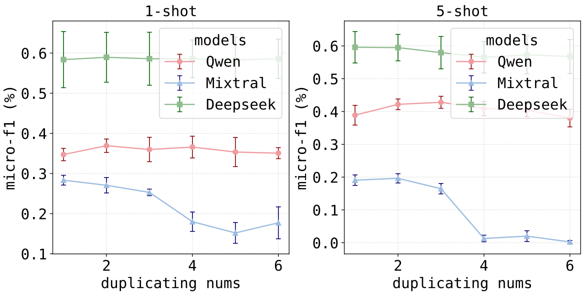



Figure 2: Micro-F1 (%) results with different duplicating times on Onto5-EN when we only duplicate demonstrations. Detailed results are shown in Appendix E.1.

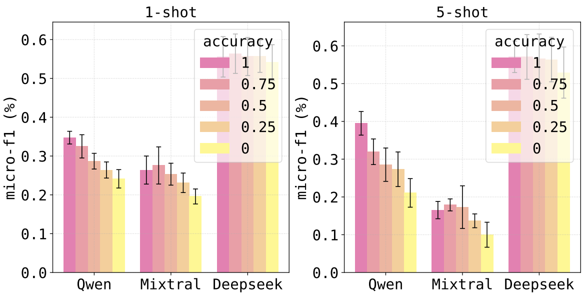



Figure 3: Micro-F1 (%) results with different label accuracy on Onto5-EN. Detailed results shown in Appendix E.2.

assigned an incorrect label (e.g., "PER") to it. The experimental results are shown in Figure 3 and Figure 8. We can observe that the FS-NER performance of the two LLMs declines as the label accuracy decreases, particularly in 5-shot setting. Note that when label accuracy is 0%, LLMs can still correctly recognize some entities due to their strong generalization, though such performance is far from that when the accuracy is 100%. Thus, we can validate our hypothesis that label accuracy is positively correlated with ICL ability on FS-NER.

### 3.4 Label Diversity, Coverage and Measure

In addition to the number of demonstrations mentioned in Section 3.2, the differences among demonstrations under various *k*-shot settings also include label diversity and label coverage. Before introducing them, we first introduce the concept of **label counter**. The label counter of a demonstration is a counter recording the numbers of different labels in the output. It can, to some extent, reflect the *label distribution* of demonstrations. For example, for the 1st demonstration in Figure 1, its label counter is {"ORG": 2, "MISC": 1, "PER": 1}, which means that there are two "ORG" labels, a "MISC" label and a "PER" label in this demonstration. Similarly, the label counter for the 2nd demonstration is {"LOC":1, "PER": 1}. Note that the label counter is order-agnostic, e.g., {"LOC":1, "PER": 1} is equivalent to {"PER": 1, "LOC":1}.

**Label Diversity.** Label diversity reflects the diversity of label counters in a context window. We believe that more diverse label counters in a context window may provide LLMs with richer reference information. To measure the label diversity, we define the LSPI (i.e., label space per instance) metric:

$$LSPI = \frac{n_{ld}}{n_d} \quad (1)$$

where $LSPI \in [0, 1]$, $n_{ld}$ is the number of unique label counters in a context window and $n_d$ is the total number of demonstrations in the context window. For example, assuming there are only two demonstrations in a context window, whose label counters are {"LOC":1, "PER": 1} and {"PER": 1, "LOC":1}, respectively. Therefore, $n_{ld}$ is 1 and $n_d$ is 2. LSPI represents the average number of unique label counters that each demonstration can provide, namely *diversity*. The larger the LSPI, the more diverse the label counter (or label distribution) in a context window.

**Label Coverage.** Label coverage indicates the degree to which the label counters in a context window cover the label counters in the test set[2]. We

[2] In practical situations, the test set are not accessible during inference. Therefore, label coverage can only be measured to

| dataset | *k*-shot | LSPI | LC | $LM_1$↑ | $LM_{0.5}$↑ |
|---|---|---|---|---|---|
| Onto5-EN | 1 | 50.00 | 1.01 | 1.97 | 4.65 |
| | 3 | 40.38 | 2.49 | 4.69 | 9.98 |
| | 5 | 46.05 | 3.56 | **6.61** | **13.59** |
| | 7 | 44.55 | 2.49 | 4.71 | 10.16 |
| Movie | 1 | 50.00 | 3.97 | 7.35 | 15.05 |
| | 3 | 41.67 | 4.82 | 8.63 | 16.47 |
| | 5 | 42.59 | 6.88 | **11.85** | **20.90** |
| | 7 | 34.15 | 6.98 | 11.59 | 19.20 |
| Onto5-ZH | 1 | 50.00 | 1.72 | 3.36 | 7.63 |
| | 3 | 47.92 | 2.61 | 4.95 | 10.72 |
| | 5 | 48.21 | 4.17 | 7.68 | 15.49 |
| | 7 | 50.00 | 8.73 | **14.87** | **25.70** |
| CMeEE-V2 | 1 | 50.00 | 4.29 | 7.89 | 15.96 |
| | 3 | 50.00 | 5.95 | 10.63 | 20.15 |
| | 5 | 41.67 | 7.69 | 12.98 | 22.11 |
| | 7 | 50.00 | 8.64 | **14.33** | **25.54** |

Table 2: LSPI(%), LC(%), and LM(%) results in (*k*=1, 3, 5, 7)-shot settings on 4 datasets. We use prompt template shown in Figure 1.

hypothesize that the more label counters of demonstrations appear in the test set, the more information of the test set is exposed to LLMs to learn, and the more likely LLMs are to output correct label counters. To measure label coverage in a context window, we define the LC (i.e., label coverage) as:

$$LC = \frac{n_{co}}{n_t} \tag{2}$$

where $LC \in [0, 1]$, $n_{co}$ is the co-occurrence number of label counters in the context window and the test set. For example, if the label counter (e.g., {"LOC":1, "PER": 1}) of a demonstration in the context window also appears in the test set, then add one to $n_{co}$. $n_t$ is the number of instances in the test set[3]. LC measures the probability of label counters in the test set that are also present in the context window, namely *coverage*. The larger the LC, the higher the label coverage.

**Label Measure.** To comprehensively consider label diversity and label coverage, we combine LC with LSPI to form the LM (i.e., label measure) metric:

$$LM_\beta = \frac{(1+\beta^2) \times LSPI \times LC}{\beta^2 \times LSPI + LC} \tag{3}$$

where $LM_\beta \in [0, 1]$, $\beta \in \mathbb{R}$ is a weighted factor. We set it to 1 (i.e., $LM_1$) or 0.5 (i.e., $LM_{0.5}$).

It's noted that LSPI, LC, and LM are model-agnostic. LSPI only measures the distribution of label counters in a context window. LC only measures the overlapping of label counters between demonstrations and the test set. In Table 2, we can observe that as *k* increases, LSPI mostly decreases, LC mostly increases, and LM shows a fluctuating upward trend. As shown in Table 3, $LM_1$ and $LM_{0.5}$ exhibit a moderate or higher degree of correlations[4] with F1 scores across nearly all datasets when using 3 different LLMs. The negative outcomes in Table 3 may be attributable to the increase of *k* in *k*-shot NER, which leads to an extended context length and consequently a decline in the performance of LLMs when processing long contexts. Based on these observations, we can conclude that both label diversity and label coverage exhibit a moderate to high degree of correlation with the performance of ICL on FS-NER.

| metrics | models | Onto5-EN | Movie | Onto5-CH | CMeEE-V2 |
|---|---|---|---|---|---|
| $LM_1$ | Qwen | 0.706 | -0.258 | -0.508 | 0.314 |
| | Mixtral | -0.591 | 0.365 | -0.300 | 0.904 |
| | DeepSeek | 0.587 | 0.965 | 0.567 | 0.223 |
| $LM_{0.5}$ | Qwen | 0.689 | -0.426 | -0.451 | 0.433 |
| | Mixtral | -0.604 | 0.417 | -0.355 | 0.911 |
| | DeepSeek | 0.609 | 0.938 | 0.514 | 0.329 |

Table 3: The Pearson correlation coefficient between LM and micro-F1 on 4 datasets ($p < 0.05$).

## 4 Label Subset Partition

It can be inferred from Section 3 that an appropriate number of demonstrations, accurate labels, diverse labels, and high-coverage labels are essential to ensure the high performance of ICL on FS-NER. Based on such a conclusion, we propose a novel method named label subset partition (i.e., LSP) to augment demonstrations in the LLMs' context window, improving label diversity and label coverage while keeping an appropriate number of accurate demonstrations. A detailed motivation is explained in Appendix D. Meanwhile, the experiment setup is same to Section 3 (detailed in Appendix A).

### 4.1 Methodology

As illustrated in Figure 4, LSP augments a demonstration by partitioning the label set of size $s$ into multiple exclusive label subsets of size $k$ ($k < s$) as many as possible[5] and thus for a sentence to produce a separate demonstration for each label subset. In detail, step 1, we randomly partition the

analyze ICL performance in this study.

[3]We set it to 200 in experiments. See Appendix A.3.

[4]The absolute value of a Pearson correlation coefficient between 0.4 and 0.6 indicates a moderate correlation, while an absolute value greater than 0.6 signifies a strong correlation.

[5]The remaining labels less than $k$ still form a label subset.

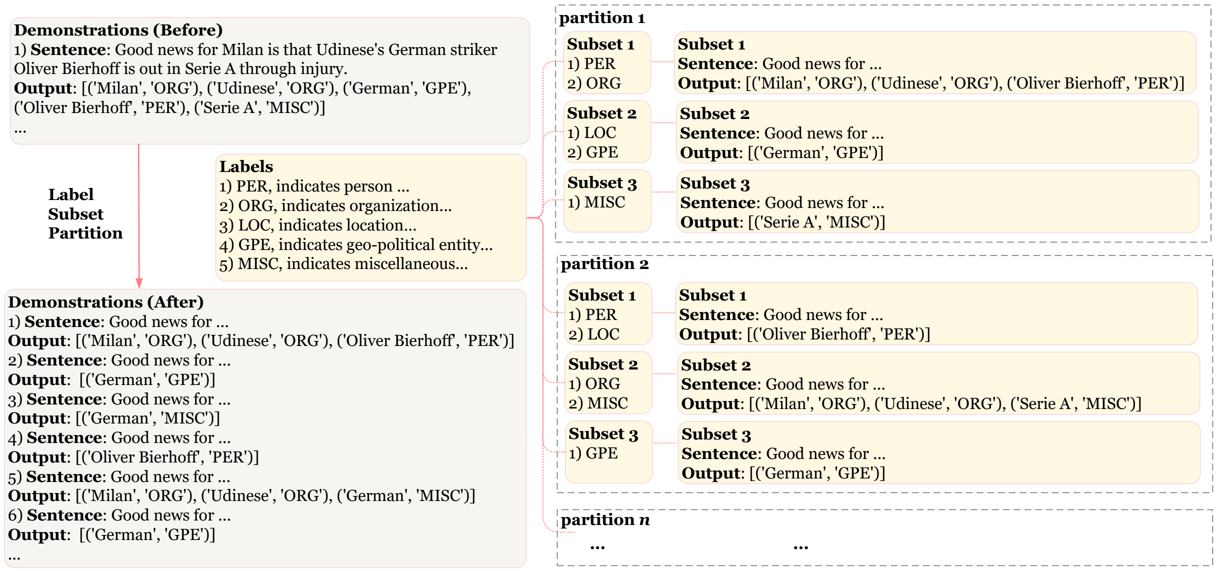



Figure 4: Overview of our proposed LSP.

original label set of size $s$ into multiple label subsets $\mathcal{L}_i$ of size $k$ ($k \leq \lfloor \frac{s}{2} \rfloor$[6]) as many as possible, where $\mathcal{L}_i \cap \mathcal{L}_j = \emptyset$ if $i \neq j$[7]. For example, in the top right of Figure 4, we partition the original label set (i.e., [PER, ORG, LOC, GPE, MISC]) of size 5 into three label subsets, including two label subsets of size $k = 2$ (i.e., [PER, ORG] and [LOC, GPE]) and a label subset (i.e., [MISC]) composed of the remaining one label. Step 2, for each label subset, we filter out entities that do not belong to this label subset in the output. For example, "German" with the "GPE" label is filtered out when we use the label subset [PER, ORG]. Now, we can obtain $\lceil \frac{s}{k} \rceil$ (i.e., $\lceil \frac{5}{2} \rceil = 3$) *new* demonstrations with distinct outputs, e.g., "['Milan', 'ORG'], ['Udinese', 'ORG'], ['Oliver Bierhoff', 'PER']" for the 1st demonstration and "['German', 'GPE']" for the 2nd demonstration. Step 3, we can repeat such partition process $n$ times to ensure that no identical subset exists in all partitions. For example, in the 2nd partition process, "PER" and "LOC" are grouped together, while they are not in the same label subset in the 1st partition process. Step 4, we concatenate all demonstrations from different label subsets and fill them into the context. It can be observed that the original single demonstration has been expanded to 6 (i.e., $\lceil \frac{s}{k} \rceil \times n$) demonstrations. It's worth noting that LSP is an augmentation method that operates only on demonstrations. So, we don't need to train LLMs (i.e., training-free), or design specific prompts (i.e., prompt-agnostic). It can also be applied to any LLMs (i.e., model-agnostic). The detailed algorithm is shown in Algorithm 1 in Appendix B.1.

### 4.2 Comparison with different ICL Methods

We compare LSP with other ICL methods for FS-NER: **Vanilla** (Ma et al., 2023) use the prompt-template shown in Figure 1. It simultaneously outputs entity mentions across all types for each query. **Vanilla+rep** purely duplicates demonstrations multiple times based on the **Vanilla** method. We duplicate demonstrations 1 time here. **Multi-qa** (Xie et al., 2023) method processes each query in a batch using a multi-turn question-answer style. **Single-type** (Wang et al., 2023a) method processes and outputs entities for only one type at a time, subsequently aggregating the results from all types. **Self-consistency** (Wang et al., 2023b) selects the final answer as the most common one across output entities. In order to establish a similar few-shot experimental setting, we remove the step of retrieving the optimal demonstrations from a large amount of labeled data from **Multi-qa** and **Single-type**. For **LSP**, we set the size of a label subset to half of the original label size (i.e., $p = 0.5$[8]). For **LSP+2**, we partition label subsets 2 times[9]. As shown in Table 4, we can observe that: (1) LSP generally achieves the best results compared to other ICL methods for FS-NER, which demonstrates the superiority of LSP. (2) After repeating partitioning,

[6]We consider that the entity labels of a demonstration are usually sparse, with no more than half of the total types.

[7]In set partitioning, each set don't intersect with each other.

[8]See detail in Section 4.3.1.

[9]See detail at Section 4.3.2

| datasets | | Onto5-EN | | Movie | | Onto5-ZH | | CMeEE-V2 | |
|---|---|---|---|---|---|---|---|---|---|
| **models** | **methods** | *k*=1 | *k*=5 | *k*=1 | *k*=5 | *k*=1 | *k*=5 | *k*=1 | *k*=5 |
| Qwen | vanilla | $35.19_{\pm1.48}$ | $38.48_{\pm2.07}$ | $67.27_{\pm1.79}$ | $64.68_{\pm2.54}$ | $38.27_{\pm4.59}$ | $40.48_{\pm1.65}$ | $43.48_{\pm0.94}$ | $42.79_{\pm0.60}$ |
| | vanilla+rep | $36.92_{\pm1.36}$ | $42.19_{\pm1.33}$ | $66.01_{\pm1.40}$ | $68.29_{\pm1.83}$ | $39.35_{\pm3.78}$ | $37.93_{\pm3.16}$ | $43.57_{\pm1.31}$ | $43.06_{\pm0.64}$ |
| | multi-qa | $35.46_{\pm3.28}$ | $40.64_{\pm1.11}$ | $66.32_{\pm2.30}$ | $64.12_{\pm1.40}$ | $37.51_{\pm2.77}$ | $36.71_{\pm2.83}$ | $42.51_{\pm1.10}$ | $41.80_{\pm0.98}$ |
| | single-type | $18.11_{\pm1.23}$ | $22.04_{\pm1.17}$ | $34.19_{\pm1.19}$ | $41.20_{\pm0.57}$ | $39.03_{\pm3.71}$ | $37.88_{\pm2.36}$ | $34.19_{\pm1.19}$ | $41.20_{\pm0.57}$ |
| | self-consistency | $35.60_{\pm1.24}$ | $38.10_{\pm3.34}$ | $\mathbf{67.74_{\pm0.57}}$ | $65.69_{\pm1.30}$ | $34.19_{\pm1.19}$ | $41.20_{\pm0.57}$ | $44.90_{\pm1.06}$ | $43.88_{\pm0.71}$ |
| | LSP | $39.37_{\pm2.01}$ | $43.09_{\pm2.06}$ | $65.58_{\pm1.09}$ | $67.33_{\pm0.62}$ | $41.15_{\pm4.73}$ | $\mathbf{43.88_{\pm3.98}}$ | $\mathbf{45.86_{\pm2.49}}$ | $\mathbf{44.82_{\pm1.16}}$ |
| | LSP+2 | $\mathbf{40.58_{\pm2.87}}$ | $\mathbf{44.81_{\pm3.14}}$ | $67.59_{\pm1.55}$ | $\mathbf{69.15_{\pm1.93}}$ | $\mathbf{42.76_{\pm1.03}}$ | $40.36_{\pm5.20}$ | $44.98_{\pm0.22}$ | $41.39_{\pm0.93}$ |
| Mixtral | vanilla | $28.33_{\pm1.00}$ | $19.08_{\pm1.30}$ | $67.22_{\pm2.17}$ | $71.03_{\pm0.70}$ | $26.84_{\pm1.57}$ | $10.28_{\pm3.67}$ | $15.94_{\pm2.41}$ | $\mathbf{31.05_{\pm1.06}}$ |
| | vanilla+rep | $27.07_{\pm1.56}$ | $19.64_{\pm1.15}$ | $68.71_{\pm0.71}$ | $71.10_{\pm2.69}$ | $28.63_{\pm1.89}$ | $\mathbf{16.16_{\pm2.45}}$ | $2.80_{\pm2.69}$ | $5.22_{\pm1.47}$ |
| | multi-qa | $26.87_{\pm3.35}$ | $18.91_{\pm1.09}$ | $61.13_{\pm0.69}$ | $66.93_{\pm0.89}$ | $28.54_{\pm1.36}$ | $15.97_{\pm1.90}$ | $\mathbf{27.05_{\pm0.97}}$ | $27.62_{\pm1.88}$ |
| | single-type | $4.82_{\pm0.17}$ | $5.34_{\pm0.47}$ | $11.31_{\pm0.54}$ | $12.99_{\pm0.42}$ | $11.31_{\pm0.54}$ | $12.99_{\pm0.42}$ | $11.31_{\pm0.54}$ | $12.99_{\pm0.42}$ |
| | self-consistency | $30.53_{\pm2.86}$ | $24.48_{\pm0.60}$ | $65.95_{\pm0.84}$ | $69.73_{\pm1.25}$ | $\mathbf{29.65_{\pm3.29}}$ | $4.53_{\pm1.64}$ | $17.74_{\pm1.65}$ | $30.22_{\pm0.58}$ |
| | LSP | $\mathbf{29.96_{\pm2.68}}$ | $\mathbf{21.11_{\pm0.21}}$ | $69.09_{\pm1.31}$ | $\mathbf{72.52_{\pm2.43}}$ | $28.42_{\pm2.21}$ | $14.57_{\pm4.55}$ | $14.65_{\pm1.34}$ | $20.95_{\pm1.96}$ |
| | LSP+2 | $26.07_{\pm0.78}$ | $11.41_{\pm4.74}$ | $\mathbf{69.85_{\pm1.81}}$ | $57.89_{\pm1.97}$ | $24.88_{\pm3.01}$ | $1.06_{\pm1.50}$ | $0.00_{\pm0.00}$ | $3.04_{\pm0.51}$ |
| DeepSeek | vanilla | $58.37_{\pm5.71}$ | $59.59_{\pm3.91}$ | $76.48_{\pm1.18}$ | $\mathbf{79.89_{\pm2.08}}$ | $59.39_{\pm3.18}$ | $57.93_{\pm2.58}$ | $\mathbf{52.53_{\pm3.14}}$ | $51.45_{\pm1.89}$ |
| | vanilla+rep | $\mathbf{59.95_{\pm5.07}}$ | $59.48_{\pm3.29}$ | $77.65_{\pm1.49}$ | $78.33_{\pm0.99}$ | $57.73_{\pm2.08}$ | $60.66_{\pm2.36}$ | $51.87_{\pm1.69}$ | $51.78_{\pm1.65}$ |
| | multi-qa | $55.61_{\pm4.80}$ | $56.19_{\pm3.12}$ | $69.65_{\pm1.31}$ | $70.69_{\pm1.46}$ | $50.45_{\pm2.57}$ | $52.20_{\pm2.23}$ | $44.78_{\pm2.24}$ | $45.03_{\pm0.93}$ |
| | single-type | $28.30_{\pm5.20}$ | $31.42_{\pm3.42}$ | $61.28_{\pm2.23}$ | $62.62_{\pm2.67}$ | $36.00_{\pm1.16}$ | $37.81_{\pm0.85}$ | $37.56_{\pm1.42}$ | $39.08_{\pm0.86}$ |
| | self-consistency | $59.14_{\pm5.45}$ | $58.09_{\pm1.73}$ | $76.43_{\pm2.71}$ | $78.69_{\pm1.56}$ | $59.93_{\pm1.16}$ | $59.71_{\pm2.52}$ | $52.33_{\pm2.98}$ | $52.18_{\pm1.68}$ |
| | LSP | $58.69_{\pm5.33}$ | $\mathbf{61.32_{\pm2.64}}$ | $76.66_{\pm2.00}$ | $77.36_{\pm0.87}$ | $57.87_{\pm4.68}$ | $\mathbf{60.88_{\pm1.93}}$ | $50.92_{\pm2.11}$ | $51.30_{\pm2.71}$ |
| | LSP+2 | $59.81_{\pm4.14}$ | $61.00_{\pm4.27}$ | $\mathbf{77.81_{\pm2.26}}$ | $77.62_{\pm3.40}$ | $\mathbf{61.94_{\pm3.32}}$ | $59.66_{\pm1.33}$ | $51.88_{\pm2.02}$ | $\mathbf{53.17_{\pm2.19}}$ |

Table 4: Micro-F1 (%) results using different ICL methods and different LLMs in (*k*=1, 5)-shot settings on 4 datasets. **Bold** results represent the best method using the same LLMs.

LSP shows better results when using Qwen and DeepSeek, though this observation does not apply to Mixtral. We conjecture that the extended context length, resulting from the subset partition and expansion of demonstrations, leads to a degradation in the performance of Mixtral. (3) Compared to using LLMs with larger parameters like DeepSeek, the performance improvement of LSP is more significant when using LLMs with smaller parameters like Qwen.

## 4.3 Analysis

### 4.3.1 Size of Label Subsets

We conduct experiments to explore the optimal size of label subsets. Given that the original label sets of different datasets vary in size, we use subset proportion *p* to determine the size of label subsets. If the size of the original label set is $s$ and the size of a label subset is $k$, the subset proportion is defined as $p = \frac{k}{s}$. Due to the non-overlapping nature of any two subsets (i.e., $\mathcal{L}_i \cap \mathcal{L}_j = \emptyset$ if $i \neq j$) when the subset proportion is set to exceed 0.5, the sizes of the subsets become uneven (e.g., 0.6 for one label subset and 0.4 for the other). Thus, we set the $p$ from 0.1 to 0.5 here. From Figure 5 and Figure 9, it can be observed that as the proportion increases from 0.1 to 0.5, the micro-F1 score generally exhibits an upward trend. It can also be determined that the model performance is generally optimal when $p = 0.5$. We speculate that the larger the size of the label subset, the richer the combinations of labels in a demonstration, and the more information available for ICL. Consequently, we select $p = 0.5$ as the optimal configuration for LSP.

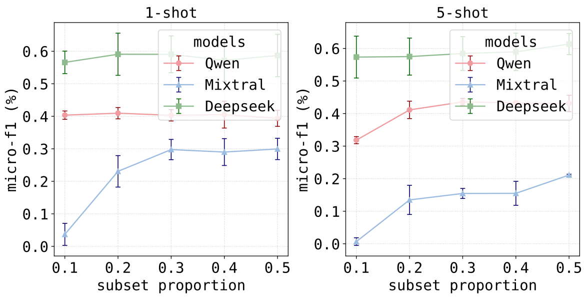



Figure 5: Micro-F1 (%) results with different subset proportions on Onto5-EN. Detailed results are shown in Appendix E.3.

### 4.3.2 Partition Times

As mentioned in Section 4.1, we can repeatedly partition label subsets *n* times. So, in this section, we aim to investigate the optimal partition times. In Figure 6 and Figure 10, we can observe that, across nearly all datasets, appropriately increasing partition times improves the FS-NER performance of Qwen and DeepSeek using 1-shot and 5-shot setting. This is because the more partition times is, the more label subsets can cover more combinations of the original labels. However, this conclusion is only valid for Mixtral under the 1-shot setting. When using the 5-shot setting, the FS-NER performance of Mixtral deteriorates with the increasing

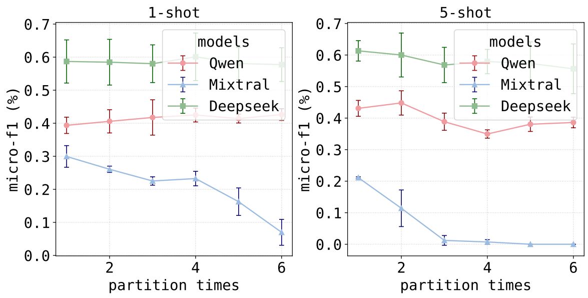



Figure 6: Micro-F1 (%) results with different partition times on Onto5-EN. Detailed results are shown in Appendix E.4.

| settings | | 1-shot | | | 5-shot | | |
|---|---|---|---|---|---|---|---|
| dataset | methods | LSPI | LC | $LM_1$↑ | LSPI | LC | $LM_1$↑ |
| Onto5-EN | vanilla | 50.00 | 1.01 | 1.97 | 46.05 | 3.56 | 6.61 |
| | vanilla+rep | 33.33 | 1.01 | 1.95 | 30.70 | 3.56 | 6.38 |
| | LSP | 90.00 | 9.59 | 17.33 | 75.13 | 15.71 | 25.99 |
| | LSP+2 | 65.00 | 11.87 | **20.07** | 57.28 | 20.98 | **30.71** |
| Movie | vanilla | 50.00 | 3.97 | 7.35 | 42.59 | 6.88 | 11.85 |
| | vanilla+rep | 33.33 | 3.97 | 7.09 | 28.40 | 6.88 | 11.08 |
| | LSP | 80.95 | 7.43 | 13.61 | 55.73 | 14.36 | **22.83** |
| | LSP+2 | 58.33 | 11.32 | **18.96** | 36.96 | 16.00 | 22.33 |
| Onto5-ZH | vanilla | 50.00 | 1.72 | 3.33 | 48.21 | 1.25 | 2.44 |
| | vanilla+rep | 33.33 | 1.72 | 3.27 | 32.14 | 1.25 | 2.41 |
| | LSP | 85.86 | 15.34 | 21.48 | 85.15 | 12.29 | 21.48 |
| | LSP+2 | 62.88 | 21.32 | **31.84** | 70.51 | 22.94 | **34.62** |
| CMeEE-V2 | vanilla | 50.00 | 0.75 | 1.48 | 41.67 | 1.23 | 2.39 |
| | vanilla+rep | 33.33 | 0.75 | 1.47 | 32.45 | 1.23 | 2.37 |
| | LSP | 80.56 | 4.05 | 7.71 | 71.96 | 5.66 | 10.49 |
| | LSP+2 | 59.72 | 4.37 | **8.14** | 46.57 | 6.53 | **11.45** |

Table 5: LSPI(%), LC(%) and LM(%) for different ICL methods on 4 datasets

partition times, due to its inability to handle the increasing context length. Based on our observation, we choose $n = 2$ as the optimal configuration for LSP.

### 4.3.3 Why is LSP effective

To explain why LSP is effective, we adopt LSPI, LC, and $LM_1$ to measure label diversity and label coverage of our proposed LSP. From Table 5, we can see that, compared to **Vanilla**[10] and **Vanilla+rep**, LSP can improve LSPI, LC and $LM_1$. When we partition label subsets 2 times (i.e., LSP+2), $LM_1$ is getting greater. This trend indicates that LSP augments demonstrations by increasing label diversity and coverage. This suggests that LSP can improve ICL performance on FS-NER by increasing label diversity and coverage, providing LLMs with more diverse and targeted label information for inference, thereby enhancing their ICL ability on FS-NER.

[10]The demonstrations used in vanilla, multi-qa, single-type and self-consistency are the same, so their LSPI, LC and LM are the same.

| *k*-shot | methods | APL | SPI↓ | t-Δ(%) | F1(%)↑ | F1-Δ(%) |
|---|---|---|---|---|---|---|
| 1 | vanilla | 1475 | 0.764 | \ | 35.19 | \ |
| | LSP | 1920 | 0.891 | 16.63 | 39.37 | 11.88 |
| | LSP+2 | 3164 | 1.389 | 81.83 | 40.58 | 15.32 |
| 5 | vanilla | 3717 | 1.611 | \ | 38.48 | \ |
| | LSP | 5229 | 2.192 | 36.10 | 43.09 | 11.98 |
| | LSP+2 | 9724 | 4.160 | 158.28 | 44.81 | 16.45 |

Table 6: Efficiency cost for different methods using Qwen on Onto5-EN. **APL** indicates average prompt length. **SPI** means seconds per instance. **t-Δ** represents the degree of improvement of each variant relative to vanilla on **SPI**. **F1-Δ** represents the degree of improvement of each variant relative to vanilla on **F1**.

### 4.3.4 Efficiency Cost

To balance FS-NER performance and computational cost, we measure prompt length, inference speed, and micro-F1 using different methods. In Table 6 and Table 9, it can be observed that: (1) When using Qwen on general domain datasets like Onto5-EN and Onto5-ZH, the increase in inference time is tolerable, compared to the FS-NER performance improvement brought about by LSP. For example, LSP achieve an improvement of 11.88% on F1 when it only spends an additional 16.63% of inference time under the 5-shot setting on Onto5-EN. Similarly, LSP+2 spends an additional 19.83% on inference costs in exchange for a 11.73% F1 boost, when using the 5-shot setting on Onto5-ZH. (2) When using Qwen on domain-specific datasets like Movie and CMeEE-V2, the inference consumption increases, but the desired performance improvement is not achieved. For example, we consume an additional 111.51% of inference time but only achieve a 6.91% F1 improvement using the 5-shot setting on Movie.

## 5 Conclusion

In this paper, we systematically explore the impact of demonstrations on the ICL on FS-NER. To measure label diversity and label coverage, we devise LSPI, LC, and LM metrics. We find that an appropriate number of demonstrations, accurate labels, diverse labels, and labels with high coverage of the test set are essential to ensure the performance of ICL on FS-NER. Based on this conclusion, we propose LSP to augment demonstrations in the context window of LLMs. Extensive experiments prove the superiority of LSP.

## Limitation

This paper only explores the effect of demonstrations for ICL on FS-NER, excluding instructions, labels, and queries. We are not yet clear whether instructions, labels, demonstrations, and queries affect each other for ICL on FS-NER. So, we leave this to future work. In addition, all conclusions from this study may not generalize for other structured prediction tasks (e.g., event extraction, coreference resolution).

According to the analysis section, LSP improves FS-NER performance at the cost of inference consumption. This means that LSP is not a universal method and should be used selectively considering specific usage scenarios.

In our study, we observed that the performances of Mixtral are generally worse than those of Qwen in most cases. And in most charts, the performance trend of Mixtral is inconsistent with that of Qwen. This may be due to differences in their abilities caused by different pre-training processes, or it may be performance bias caused by quantization. However, those observation do not affect our conclusion that LSP benefit the ICL performance on FS-NER for different LLMs.

## Acknowledgments

Acknowledgments is empty for double-blind review.

## References

Ankit Agrawal, Sarsij Tripathi, and Manu Vardhan. 2021. Active learning approach using a modified least confidence sampling strategy for named entity recognition. *Progress in Artificial Intelligence*, 10(2):113–128.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *Preprint*, arXiv:2309.16609.

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. *Preprint*, arXiv:2005.14165.

Samuel Cahyawijaya, Holy Lovenia, and Pascale Fung. 2024. LLMs are few-shot in-context low-resource language learners. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 405–433, Mexico City, Mexico. Association for Computational Linguistics.

Xiang Dai and Heike Adel. 2020. An analysis of simple data augmentation for named entity recognition. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3861–3867, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Cyprien de Lichy, Hadrien Glaude, and William Campbell. 2021. Meta-learning for few-shot named entity recognition. In *Proceedings of the 1st Workshop on Meta Learning and Its Applications to Natural Language Processing*, pages 44–58, Online. Association for Computational Linguistics.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Chauhan Dev, Naman Biyani, Nirmal P. Suthar, Prashant Kumar, and Priyanshu Agarwal. 2021. Structured prediction in NLP - A survey. *CoRR*, abs/2110.02057.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, Miami, Florida, USA. Association for Computational Linguistics.

Yingwen Fu, Nankai Lin, Xiaohui Yu, and Shengyi Jiang. 2023. Self-training with double selectors for low-resource named entity recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1265–1275.

Ridong Han, Chaohao Yang, Tao Peng, Prayag Tiwari, Xiang Wan, Lu Liu, and Benyou Wang. 2024. An empirical study on information extraction using large language models. *Preprint*, arXiv:2305.14450.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA. Association for Computational Linguistics.

Yushi Hu, Chia-Hsuan Lee, Tianbao Xie, Tao Yu, Noah A. Smith, and Mari Ostendorf. 2022. In-context learning for few-shot dialogue state tracking. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2627–2643, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

Yucheng Huang, Kai He, Yige Wang, Xianli Zhang, Tieliang Gong, Rui Mao, and Chen Li. 2022. COPNER: Contrastive learning with prompt guiding for few-shot named entity recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2515–2527, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Jing Li, Billy Chiu, Shanshan Feng, and Hao Wang. 2022. Few-shot named entity recognition via meta-learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(9):4245–4256.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. Bond: Bert-assisted open-domain named entity recognition with distant supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 1054–1064, New York, NY, USA. Association for Computing Machinery.

Jian Liu, Yufeng Chen, and Jinan Xu. 2022. Low-resource ner by data augmentation with prompting. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4252–4258. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Jingjing Liu, Panupong Pasupat, Scott Cyphers, and Jim Glass. 2013. Asgard: A portable architecture for multilingual dialogue systems. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8386–8390.

Linlin Liu, Bosheng Ding, Lidong Bing, Shafiq Joty, Luo Si, and Chunyan Miao. 2021. MulDA: A multilingual data augmentation framework for low-resource cross-lingual NER. 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 5834–5846, Online. Association for Computational Linguistics.

Jie Ma, Miguel Ballesteros, Srikanth Doss, Rishita Anubhai, Sunil Mallya, Yaser Al-Onaizan, and Dan Roth. 2022a. Label semantics for few shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1956–1971, Dublin, Ireland. Association for Computational Linguistics.

Tingting Ma, Huiqiang Jiang, Qianhui Wu, Tiejun Zhao, and Chin-Yew Lin. 2022b. Decomposed meta-learning for few-shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1584–1596, Dublin, Ireland. Association for Computational Linguistics.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10572–10601, Singapore. Association for Computational Linguistics.

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.

Vincenzo Moscato, Marco Postiglione, and Giancarlo Sperlí. 2023. Few-shot named entity recognition: Definition, taxonomy and research directions. *ACM Trans. Intell. Syst. Technol.*, 14(5).

Kai Qian, Yinqiu Liu, Chaoran Shu, Yanfei Sun, and Kun Wang. 2023. Fine-grained benchmarking and targeted optimization: Enabling green iot-oriented blockchain in the 6g era. *IEEE Transactions on Green Communications and Networking*, 7(2):1036–1051.

Xiaoye Qu, Jun Zeng, Daizong Liu, Zhefeng Wang, Baoxing Huai, and Pan Zhou. 2023. Distantly-supervised named entity recognition with adaptive teacher learning and fine-grained student ensemble. 37:13501–13509.

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

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for*

Computational Linguistics, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Hamidreza Rouzegar and Masoud Makrehchi. 2024. Enhancing text classification through LLM-driven active learning and human annotation. In *Proceedings of the 18th Linguistic Annotation Workshop (LAW-XVIII)*, pages 98–111, St. Julians, Malta. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

Joan Santoso, Patrick Sutanto, Billy Cahyadi, and Esther Setiawan. 2024. Pushing the limits of low-resource NER using LLM artificial data generation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9652–9667, Bangkok, Thailand. Association for Computational Linguistics.

Sihan Song, Furao Shen, and Jian Zhao. 2024. Ropda: Robust prompt-based data augmentation for low-resource named entity recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19017–19025.

Meihan Tong, Shuai Wang, Bin Xu, Yixin Cao, Minghui Liu, Lei Hou, and Juanzi Li. 2021. Learning from miscellaneous other-class words for few-shot named entity recognition. 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 6236–6247, Online. Association for Computational Linguistics.

Guanghai Wang, Yudong Liu, and James Hearne. 2022. Few-shot learning for Sumerian named entity recognition. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 136–145, Hybrid. Association for Computational Linguistics.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023a. Gpt-ner: Named entity recognition via large language models. *Preprint*, arXiv:2304.10428.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Chenxiao Wu, Wenjun Ke, Peng Wang, Zhizhao Luo, Guozheng Li, and Wanyi Chen. 2024. Consistner: Towards instructive ner demonstrations for llms with the consistency of ontology and context. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19234–19242.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical study of zero-shot NER with ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7935–7956, Singapore. Association for Computational Linguistics.

Tingyu Xie, Qi Li, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2024. Self-improving for zero-shot named entity recognition with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 583–593, Mexico City, Mexico. Association for Computational Linguistics.

Jiasheng Zhang, Xikai Liu, Xinyi Lai, Yan Gao, Shusen Wang, Yao Hu, and Yiqing Lin. 2023. 2INER: Instructive and in-context learning on few-shot named entity recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3940–3951, Singapore. Association for Computational Linguistics.

Ningyu Zhang, Mosha Chen, Zhen Bi, Xiaozhuan Liang, Lei Li, Xin Shang, Kangping Yin, Chuanqi Tan, Jian Xu, Fei Huang, Luo Si, Yuan Ni, Guotong Xie, Zhifang Sui, Baobao Chang, Hui Zong, Zheng Yuan, Linfeng Li, Jun Yan, Hongying Zan, Kunli Zhang, Buzhou Tang, and Qingcai Chen. 2022. CBLUE: A Chinese biomedical language understanding evaluation benchmark. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7888–7915, Dublin, Ireland. Association for Computational Linguistics.

Xinghua Zhang, Bowen Yu, Xin Cong, Taoyu Su, Quangang Li, Tingwen Liu, and Hongbo Xu. 2024. Cross-domain ner under a divide-and-transfer paradigm. *ACM Trans. Inf. Syst.*, 42(5).

# A Experiments Setup

## A.1 Models

Due to limited hardware resources, we locally deploy 4-bit GPTQ quantized Qwen1.5-32B-Chat[11] (i.e., Qwen) (Bai et al., 2023) and Mixtral-8x7B-Instruct-v0.1[12] (i.e., Mixtral) (Jiang et al., 2024) on 2 V100-32G GPUs using vLLM[13] which is a fast library for LLM inference and serving. For model with larger parameter sizes, we use DeepSeek-V3 (i.e., DeepSeek) (DeepSeek-AI, 2024) API[14].

## A.2 Datasets

In our work, we use 4 datasets to carry out experiments. For English datasets, we use OntoNotes5-EN[15] (Hovy et al., 2006) (i.e., Onto5-EN) and MIT-Movie[16] (i.e, Movie) (Liu et al., 2013). For Chinese datasets, we use OntoNotes5-ZH[17] (Hovy et al., 2006) (i.e., Onto5-ZH) and CMeEE-V2[18] (Zhang et al., 2022). Onto5-EN and Onto5-ZH are datasets in the general domain. Movie is a dataset in the movie domain. CMeEE-V2 is a dataset in the domain of biomedicine. Specific statistics are illustrated in Table 7. It's noted that #train is the official training split. We did not train any model in our work.

## A.3 Settings

Three standard metrics including precision (P), recall (R), and micro-averaging F1-score (micro-F1) are used to evaluate performance. Aiming to reduce evaluation costs, we used 3 random seeds (i.e., 22, 32, 42) to extract 3 test subsets of size 200 from different datasets and let each model variant run once on those test subsets. In other words, each model variant was run 3 times on each dataset, and the average results were reported in all of our experiments.

# B Algorithm

## B.1 Augment demonstrations by LSP

We explain the methodology of LSP in Section 4.1. The specific algorithm is shown in Algorithm 1.

| datasets | # train | # dev | # test | # types |
|---|---|---|---|---|
| Onto5-EN | 59924 | 8528 | 8262 | 18 |
| Movie | 6900 | 760 | 1521 | 12 |
| Onto5-ZH | 37557 | 6217 | 4293 | 18 |
| CMeEE-V2 | 15000 | 5000 | 3000 | 9 |

Table 7: Statistics of datasets in our experiments. # indicates the number of corresponding entries.

**Algorithm 1** Label subset partition to augment demonstrations

**Input:** demonstrations $\mathcal{S}_k = \{(\mathcal{X}_i, \mathcal{Y}_i)\}_1^N$, labels $\mathcal{L}_\mathcal{D}$, partition times $n$, subset partition proportion $p$

**Output:** augmented demonstrations $\mathcal{S}_a$

1: Initialize $\mathcal{S}_a = \emptyset$
2: Label subset size $k = \lceil |\mathcal{L}_\mathcal{D}| \times p \rceil$
3: **for** $i$ in $n$ **do**
4:     Shuffle $\mathcal{L}_\mathcal{D}$, $s = 0$ ▷ $s$ is the start position
5:     **while** $s \leq |\mathcal{L}_\mathcal{D}|$ **do**
6:         $\mathcal{L}_s \leftarrow \mathcal{L}_\mathcal{D}[s : s + k]$ ▷ Take $k$ labels in order as label subset from $\mathcal{L}_\mathcal{D}$
7:         $s \leftarrow s + k$
8:         **for** $(\mathcal{X}, \mathcal{Y})$ in $\mathcal{S}_k$ **do**
9:             Initialize $\hat{\mathcal{Y}} = \emptyset$
10:             **for** $y_i$ in $\mathcal{Y}$ **do** ▷ $y_i = (m_{y_i}, l_{y_i})$ is a label-mention pair
11:                 $\hat{\mathcal{Y}} \leftarrow \hat{\mathcal{Y}} \cup y_i$ if $l_{y_i} \in \mathcal{L}_s$ ▷ filter out labels that do not belong to $\mathcal{L}_s$
12:             **end for**
13:             $\mathcal{S}_a \leftarrow \mathcal{S}_a \cup (\mathcal{X}, \hat{\mathcal{Y}})$ ▷ Add a new demonstration
14:         **end for**
15:     **end while**
16: **end for**
        **return** $\mathcal{S}_a$

From line 3 to line 16, we partition original labels (i.e., $\mathcal{L}_\mathcal{D}$) $n$ times. In detail, from line 4 to line 7, we obtain a label subset $\mathcal{L}_s$ of size $k$. From line 8 to line 13, we add new demonstrations whose labels in the output belong to $\mathcal{L}_s$ to $\mathcal{S}_a$. It's worth noting that a demonstration $(\mathcal{X}, \mathcal{Y})$ contains a sentence $\mathcal{X}$ and an output $\mathcal{Y} = \{y_i\}_1^m$, where the output is composed of $m$ label-mention pairs. For example, ('Milan', 'MISC') is a label-mention pair $y_i = (m_{y_i}, l_{y_i})$ (i.e., $m_{y_i}$ is 'Milan' and $l_{y_i}$ is 'MISC') in the output of a demonstration. The time complexity of this algorithm is $\mathcal{O}(n^4)$. If we partition 1 time, the time complexity of this algorithm is $\mathcal{O}(n^3)$.

[11] https://huggingface.co/Qwen/Qwen1.5-32B-Chat-GPTQ-Int4
[12] https://huggingface.co/TheBloke/Mixtral-8x7B-Instruct-v0.1-GPTQ
[13] https://docs.vllm.ai/en/stable/
[14] https://api-docs.deepseek.com/
[15] https://catalog.ldc.upenn.edu/LDC2013T19
[16] https://sls.csail.mit.edu/downloads/movie/
[17] https://catalog.ldc.upenn.edu/LDC2013T19
[18] https://tianchi.aliyun.com/dataset/95414

**Algorithm 2** Greedy algorithm to sample $k$-shot demonstrations

**Input:** shot $k$, dataset $\mathcal{D} = \{(\mathcal{X}_i, \mathcal{Y}_i)\}_1^N$, labels $\mathcal{L}_\mathcal{D}$
**Output:** $k$-shot demonstrations $\mathcal{S}_k$

```
1:  Initialize S_k = ∅, Count_{l_i} = 0 (∀l_i ∈ L_D)
2:  for l in L_D do
3:      while Count_l < k do
4:          Sample (X, Y) from D \ S_k that Y includes l
5:          S_k ← S_k ∪ (X, Y)
6:          Update all Count_{l_i} (∀l_i ∈ L_D)
7:      end while
8:  end for
9:  for (X, Y) in S_k do
10:     S_k = S_k \ (X, Y)
11:     Update all Count_{l_i} (∀l_i ∈ L_D)
12:     if Any Count_{l_i} < k then
13:         S_k ← S_k ∪ (X, Y)
14:     end if
15: end for
        return S_k
```

**Algorithm 3** Get demonstrations with accuracy $\beta$

**Input:** demonstrations $\mathcal{S}_k = \{(\mathcal{X}_i, \mathcal{Y}_i)\}_1^M$, labels $\mathcal{L}$, accuracy $\beta$
**Output:** demonstrations $\mathcal{S}_\beta$ with accuracy $\beta$

```
1:  Initialize S_β = ∅
2:  for (X, Y) in S_k do
3:      Shuffle all label-mention pairs in Y
4:      n ← |Y| × (1 − β)        ▷ number of incorrect pairs
5:      Y_w ← Y[: n]             ▷ first n pairs of Y are wrong pairs
6:      Y_c ← Y[n :]             ▷ remaining pairs of Y are correct pairs
7:      for y_i in Y_w do        ▷ y_i = (m_{y_i}, l_{y_i}) is a label-mention pair
8:          replace l_{y_i} with other label l_j ∈ L that l_j ≠ l_{y_i}
9:      end for
10:     Y ← Y_w ∪ Y_c
11:     S_β ← S_β ∪ (X, Y)
12: end for
        return S_β
```

### B.2 Demonstration sampling

We sample demonstrations from different datasets using Algorithm 2 (Ma et al., 2022a). From line 1 to line 8, we sample an instance that includes class $l$ if the number of all the $l$-class entities is less than $k$. From line 9 to line 15, we try to remove redundant instances from the $k$-shot demonstrations $\mathcal{S}_k$. Note that the actual sample number of each label can be larger than $k$ using this greedy sampling strategy. The time complexity of this algorithm is $\mathcal{O}(n^3)$.

### B.3 Control Label Accuracy

In Section 3.3, we control the accuracy of demonstrations using Algorithm 3. From line 3 to line 6, we randomly select $n = |\mathcal{Y}| \times (1 - \beta)$ pairs as incorrect pairs $\mathcal{Y}_w$. From line 7 to line 11, we randomly replace the label with another label for each mention in $\mathcal{Y}_w$. It's worth noting that a demonstration $(\mathcal{X}, \mathcal{Y})$ contains a sentence $x$ and an output $\mathcal{Y} = \{y_i\}_1^n$, where the output is composed of $n$ label-mention pairs. For example, ('Milan', 'MISC') is a label-mention pair $y_i = (m_{y_i}, l_{y_i})$ (i.e., $m_{y_i}$ is 'Milan' and $l_{y_i}$ is 'MISC') in the output of a demonstration. The time complexity of this algorithm is $\mathcal{O}(n^2)$. It is worth noting that when the number of labels is small (e.g., less than 2), the number of correct labels is almost the same under different accuracy settings. Therefore, we ensure that demonstration with less than 2 labels only accounted for 30% when we sample $k$-shot demonstrations using Algorithm 2.

## C Preliminary Experiment

We conduct a preliminary experiment to explore the impact of $k$ on NER performance using different $k$-shot settings. In Table 8, we can observe that as $k$ increases, the NER performances generally improve on 4 datasets.

## D Motivation behind LSP

It can be inferred from Section 3 that an appropriate number of demonstrations, accurate labels, diverse labels, and high-coverage labels are essential to ensure the high performance of ICL on FS-NER. Therefore, if we can maximize label diversity (measured by $LSPI = \frac{n_{ld}}{n_d}$) and label coverage (measured by $LC = \frac{n_{co}}{n_t}$) in the appropriate number of correct demonstrations, we may be able to improve ICL performance on FS-NER. $LM_\beta$ is the combination of LSPI and LC. Without loss of generality, let's discuss the case where $\beta = 1$.

According to GM-HM Inequality (i.e., $\frac{2}{(\frac{1}{a}+\frac{1}{b})} \leq$

$\sqrt{ab}$). We can carry out the following derivation:

$$\begin{aligned} LM_1 &= \frac{2}{(\frac{1}{LSPI} + \frac{1}{LC})} \\ &\leq \sqrt{LSPI \times LC} \\ \Rightarrow \frac{1}{LM_1} &= \frac{1}{2}(\frac{1}{LSPI} + \frac{1}{LC}) \\ &\geq \frac{1}{\sqrt{LSPI \times LC}} \\ &= \frac{1}{\sqrt{\frac{n_{ld} n_{co}}{n_d n_t}}} \end{aligned}$$

where $n_t$ is a constant value. Hence, if we want to improve $LM_1$, we should improve $\frac{n_{ld} n_{co}}{n_d}$. As described in Section 3.4, both $n_{ld}$ and $n_{co}$ are determined by the number of label counters in a context window. On the one hand, the more unique label counter in a context window, the bigger $n_{ld}$ is. On the other hand, the more unique label counter in a context window, the more likely it is to have the same label counter as in the test set (i.e., the bigger $n_{co}$ is).

Based on the above reasoning, we need to enrich the label counters as many as possible. Consequently, we propose LSP that can augment demonstrations by partitioning the label set of size $s$ into multiple exclusive label subsets of size $k$ ($k < s$) as many as possible. Those label subsets can construct diverse label counters for each demonstration in a context window to improve $n_{ld}$ and $n_{co}$, thereby improve $LM_1$.

In addition to LSP, there is another intuitive method to improve $n_{ld}$ and $n_{co}$: directly construct different label combinations based on the labels of each demonstration to construct label counters. However, such a method cannot generalize to demonstrations at the paragraph level because longer demonstrations have more types (i.e., labels) of entities. Assuming we are performing a paragraph level NER task, a demonstration has a very long text containing $l$ types of entities. For this demonstration, we can take 1 to $l$ labels to construct a label counter. There is a total of $2^l - 1$[19] construction ways. Here comes a question: when we use datasets like mit-movie ($l = 12$) or Ontonotes5 ($l = 18$), there are so many label counters that we cannot fill all augmented demonstrations into the context window. Considering the generalization to paragraph-level tasks, we did not use this intuitive construction method instead of LSP.

[19] $C_l^0 + C_l^1 + \ldots + C_l^l = 2^l$

# E Detaild Results

Due to page length limitations, we present detailed experimental result figures and tables in this Appendix Section.

## E.1 The Number of Demonstrations

The detailed performance with different duplicating numbers on 4 datasets is shown in Figure 7. We can draw the same conclusion as Section 3.2: Simply duplicating demonstrations to increase the number of demonstrations does not necessarily improve ICL ability on FS-NER.

## E.2 Label Accuracy

The detailed performance with different label accuracy on 4 datasets is shown in Figure 8. We can draw the same conclusion as Section 3.3: Label accuracy is positively correlated with ICL ability on FS-NER.

## E.3 The Size of Label Subsets

The detailed results with different subset proportions on 4 datasets are shown in Figure 9. Similarly to Section 4.3.1, we can observe that the model performance is generally optimal when $p = 0.5$. Consequently, we select $p = 0.5$ as the optimal configuration for LSP.

## E.4 The Partition Times

The detailed results with partition times on 4 datasets are shown in Figure 10. We can observe the same trend shown in Section 4.3.2.

## E.5 Efficiency Cost

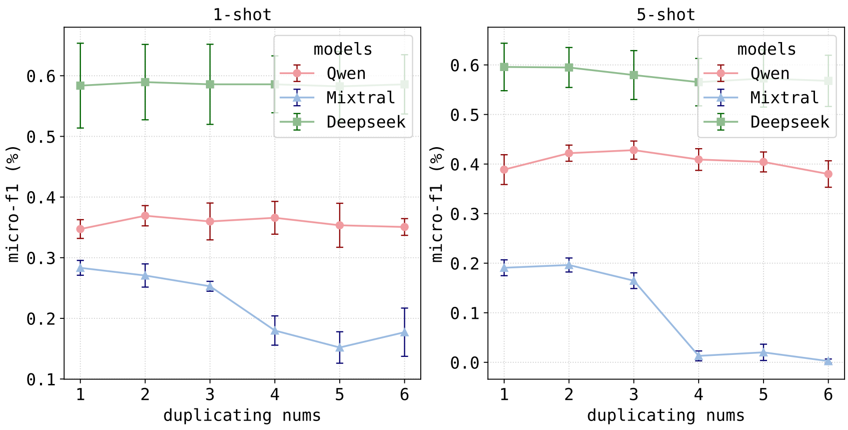
The detailed results on efficiency cost using Qwen are shown in Table 9. The same observation can be found in Section 4.3.4.

| models | $k$-shot | Onto5-EN | Movie | Onto5-ZH | CMeEE-V2 |
|---|---|---|---|---|---|
| Qwen | 1 | $34.72_{\pm1.26}$ | $67.27_{\pm1.79}$ | $38.27_{\pm4.59}$ | $43.48_{\pm0.94}$ |
| | 3 | $34.36_{\pm2.01}$ | $\mathbf{70.60_{\pm1.93}}$ | $39.28_{\pm4.84}$ | $45.34_{\pm0.81}$ |
| | 5 | $\mathbf{38.87_{\pm2.45}}$ | $64.68_{\pm2.54}$ | $\mathbf{40.48_{\pm1.65}}$ | $42.79_{\pm0.60}$ |
| | 7 | $34.66_{\pm2.80}$ | $69.92_{\pm1.09}$ | $37.11_{\pm3.72}$ | $\mathbf{45.50_{\pm0.84}}$ |
| Mixtral | 1 | $28.33_{\pm1.00}$ | $67.22_{\pm2.17}$ | $\mathbf{26.84_{\pm1.57}}$ | $15.94_{\pm2.41}$ |
| | 3 | $\mathbf{29.03_{\pm2.49}}$ | $\mathbf{72.09_{\pm1.66}}$ | $24.85_{\pm0.20}$ | $29.87_{\pm2.32}$ |
| | 5 | $19.08_{\pm1.30}$ | $71.03_{\pm0.70}$ | $10.28_{\pm3.67}$ | $31.05_{\pm1.06}$ |
| | 7 | $16.07_{\pm1.08}$ | $69.27_{\pm2.03}$ | $21.25_{\pm3.03}$ | $\mathbf{32.23_{\pm1.67}}$ |
| DeepSeek | 1 | $58.37_{\pm5.71}$ | $76.48_{\pm1.18}$ | $59.39_{\pm3.18}$ | $52.53_{\pm3.14}$ |
| | 3 | $59.10_{\pm2.43}$ | $78.35_{\pm1.88}$ | $58.03_{\pm1.85}$ | $52.70_{\pm1.73}$ |
| | 5 | $59.59_{\pm3.91}$ | $\mathbf{79.89_{\pm2.08}}$ | $57.93_{\pm2.58}$ | $51.45_{\pm1.89}$ |
| | 7 | $\mathbf{60.65_{\pm3.38}}$ | $79.74_{\pm1.82}$ | $\mathbf{60.18_{\pm1.54}}$ | $\mathbf{53.65_{\pm0.84}}$ |

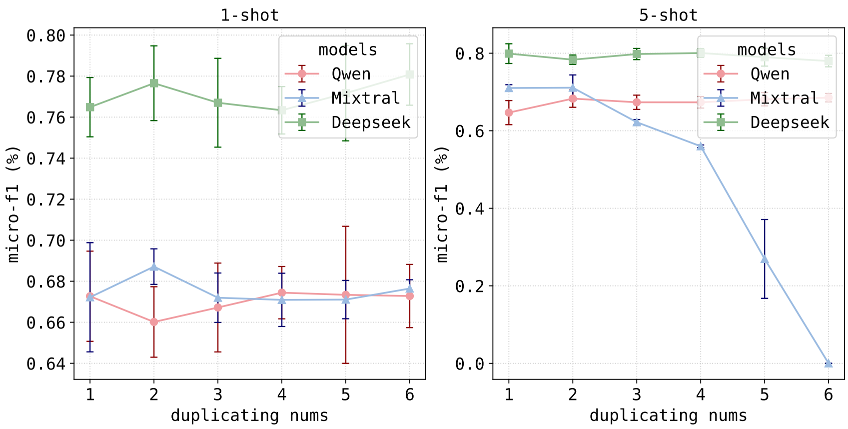
Table 8: Micro-F1 (%) results using different LLMs in ($k$=1, 3, 5, 7)-shot settings on 4 datasets. We use the prompt template shown in Figure 1. **Bold** results represent the best setting using the same LLMs.

| datasets | $k$-shot | methods | APL | SPI↓ | t-Δ(%) | F1↑ | F1-Δ(%) |
|---|---|---|---|---|---|---|---|
| Onto5-EN | 1 | vanilla | 1475 | 0.764 | \ | 35.19 | \ |
| | | LSP | 1920 | 0.891 | 16.63 | 39.37 | 11.88 |
| | | LSP+2 | 3164 | 1.389 | 81.83 | 40.58 | 15.32 |
| | 5 | vanilla | 3717 | 1.611 | \ | 38.48 | \ |
| | | LSP | 5229 | 2.192 | 36.10 | 43.09 | 11.98 |
| | | LSP+2 | 9724 | 4.160 | 158.28 | 44.81 | 16.45 |
| Movie | 1 | vanilla | 817 | 0.455 | \ | 67.27 | \ |
| | | LSP | 961 | 0.506 | 11.24 | 65.58 | -2.51 |
| | | LSP+2 | 1420 | 0.688 | 51.39 | 67.59 | 0.48 |
| | 5 | vanilla | 1751 | 0.825 | \ | 64.68 | \ |
| | | LSP | 2176 | 0.976 | 18.25 | 67.33 | 4.10 |
| | | LSP+2 | 4101 | 1.745 | 111.51 | 69.15 | 6.91 |
| Onto5-ZH | 1 | vanilla | 1163 | 0.807 | \ | 38.27 | \ |
| | | LSP | 1511 | 0.924 | 14.50 | 41.15 | 7.53 |
| | | LSP+2 | 2592 | 0.967 | 19.83 | 42.76 | 11.73 |
| | 5 | vanilla | 3237 | 1.812 | \ | 40.48 | \ |
| | | LSP | 4455 | 2.372 | 30.91 | 43.88 | 8.40 |
| | | LSP+2 | 9116 | 3.273 | 80.63 | 40.36 | -0.30 |
| CMeEE-V2 | 1 | vanilla | 2286 | 1.478 | \ | 43.48 | \ |
| | | LSP | 4557 | 2.516 | 70.23 | 45.86 | 5.47 |
| | | LSP+2 | 8366 | 7.605 | 414.55 | 44.98 | 3.45 |
| | 5 | vanilla | 1992 | 1.366 | \ | 42.79 | \ |
| | | LSP | 2973 | 1.729 | 26.57 | 44.82 | 4.74 |
| | | LSP+2 | 5424 | 4.764 | 248.76 | 41.39 | -3.27 |

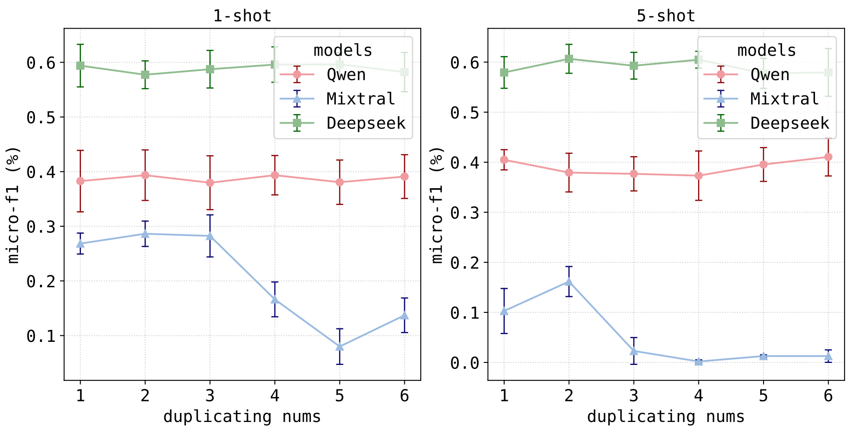
Table 9: Efficiency cost for different methods using Qwen on 4 datasets. **APL** indicates average prompt length. **SPI** means seconds per instance. **t-Δ** represents the degree of improvement of each variant relative to vanilla on **SPI**. **F1-Δ** represents the degree of improvement of each variant relative to vanilla on **F1**.



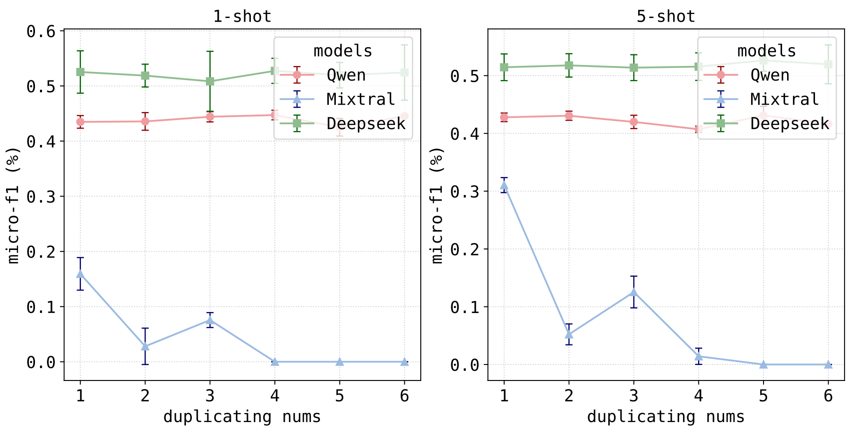
(a) Onto5-EN



(b) Movie

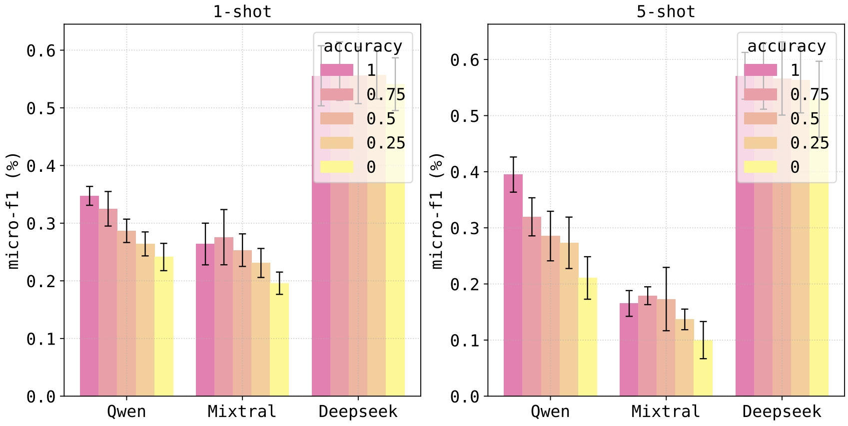


(c) Onto5-ZH

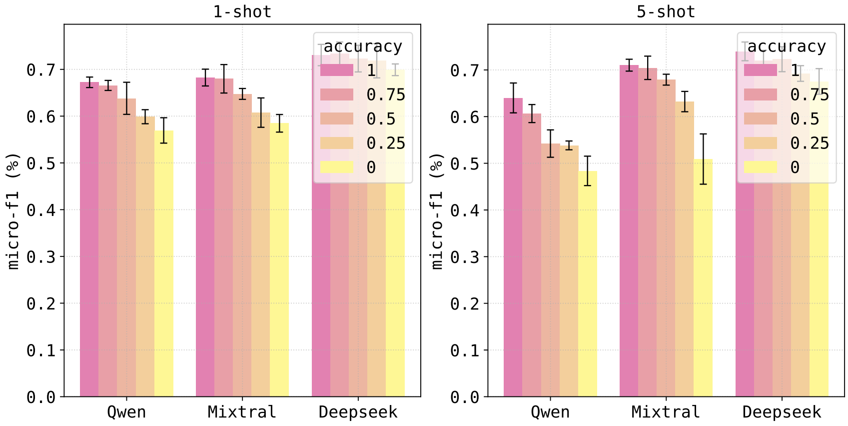


(d) CMeEE-V2

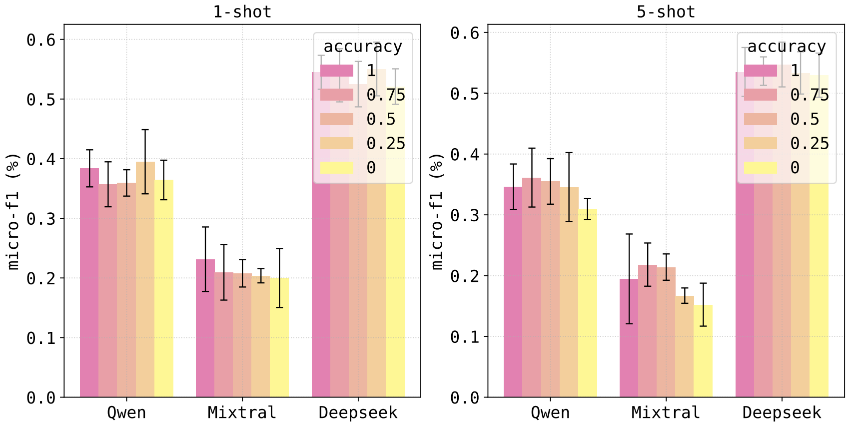
Figure 7: Micro-F1 (%) results with different duplicating times on 4 datasets when we only duplicate demonstrations.



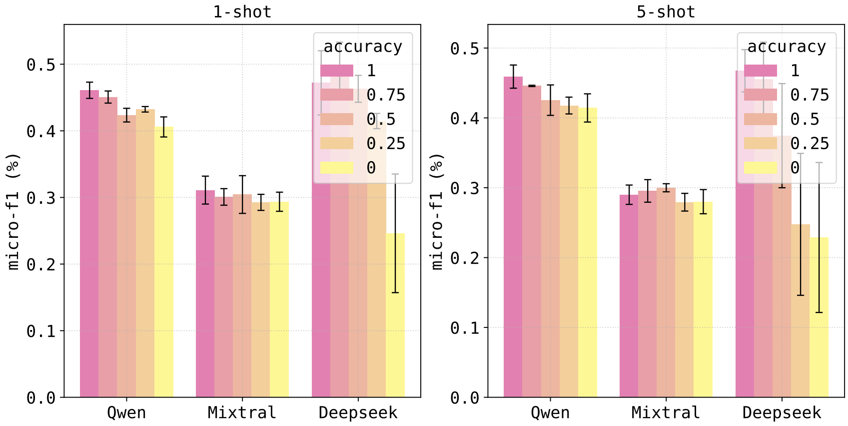

(a) Onto5-EN




(b) Movie

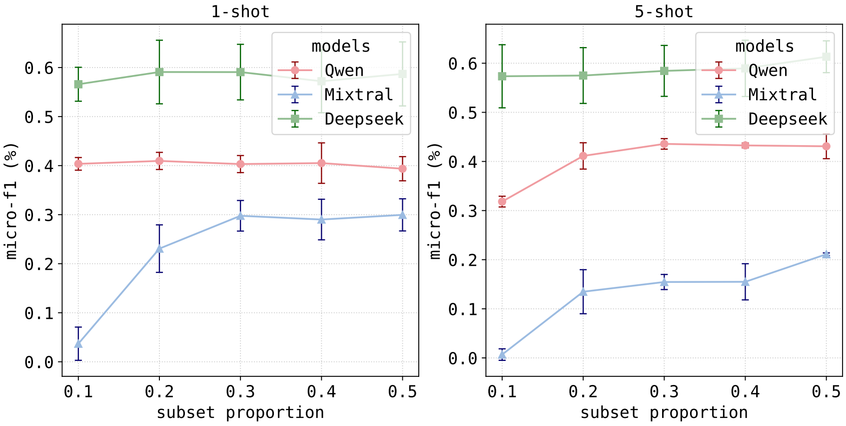



(c) Onto5-ZH

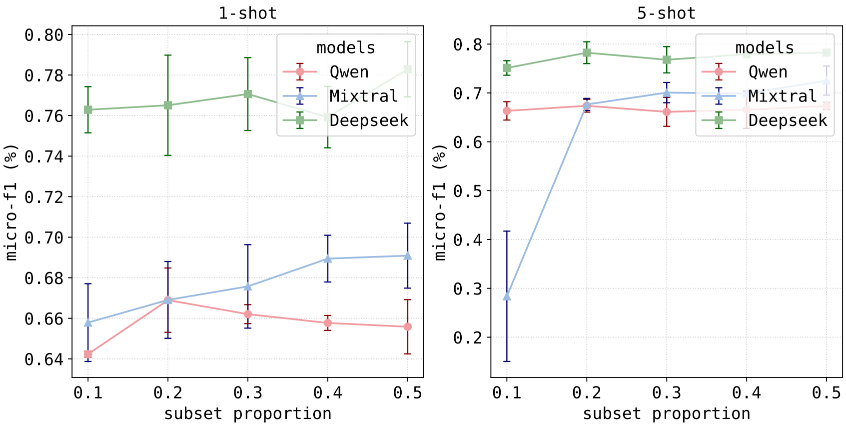



(d) CMeEE-V2

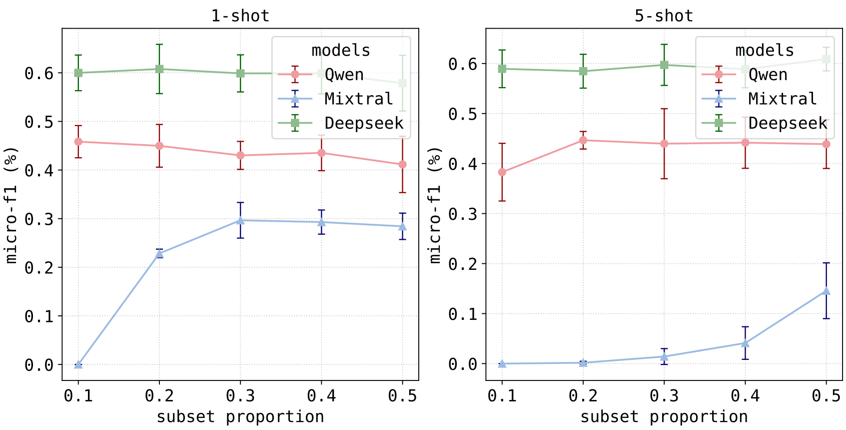
Figure 8: Micro-F1 (%) results with different label accuracy on 4 datasets.



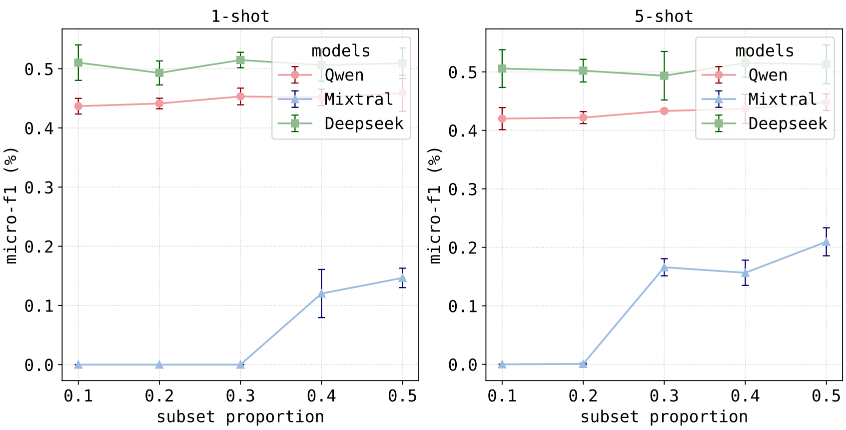

(a) Onto5-EN




(b) Movie

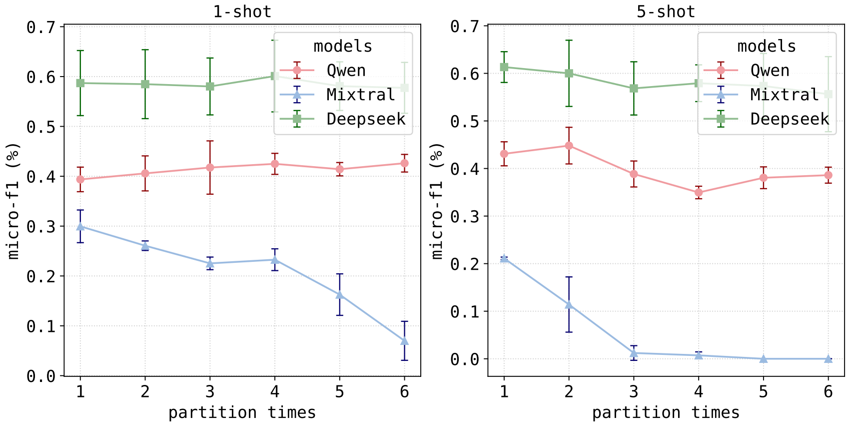



(c) Onto5-ZH

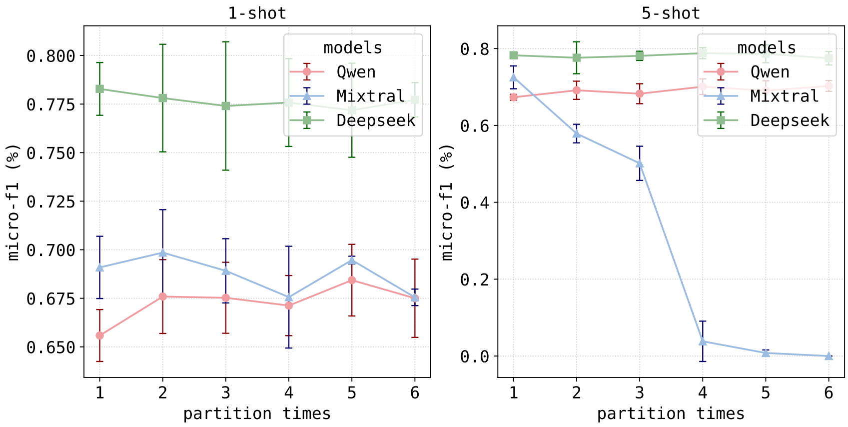



(d) CMeEE-V2

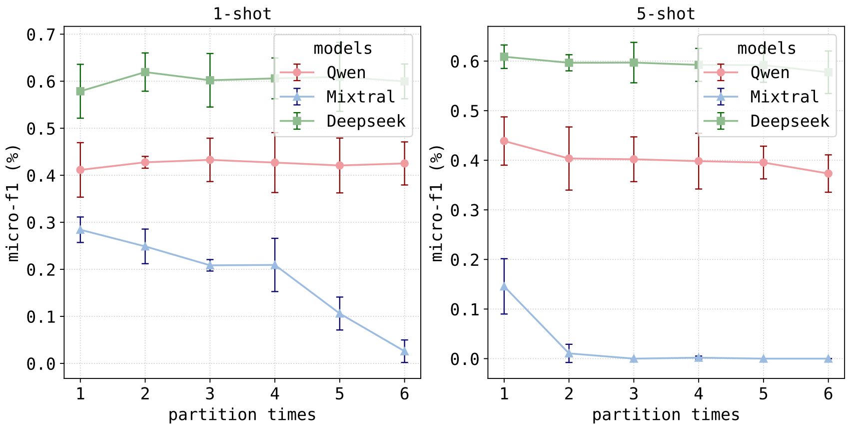
Figure 9: Micro-F1 (%) results with different subset proportions on 4 datasets when we use LSP.



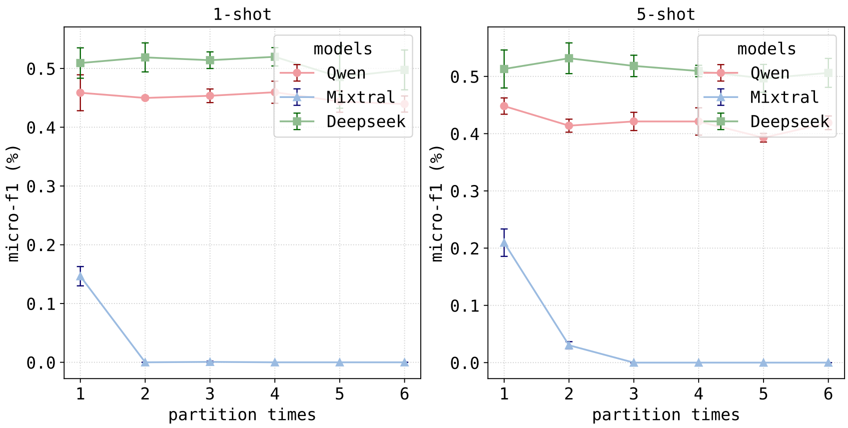

(a) Onto5-EN




(b) Movie




(c) Onto5-ZH




(d) CMeEE-V2

Figure 10: Micro-F1 (%) results with different partition times on 4 datasets when we use LSP.