ALLabel: Three-stage Active Learning for LLM-based Entity Recognition using Demonstration Retrieval

Zihan Chen[1], Lei Shi[1*], Weize Wu[1], Qiji Zhou[2], Yue Zhang[2]

[1]Beihang University [2]Westlake University
{chenzihan2001, leishi, wuweize}@buaa.edu.cn
{zhouqiji, zhangyue}@westlake.edu.cn

Abstract

Many contemporary data-driven research efforts in the natural sciences, such as chemistry and materials science, require large-scale, high-performance entity recognition from scientific datasets. Large language models (LLMs) have increasingly been adopted to solve the entity recognition task, with the same trend being observed on all-spectrum NLP tasks. The prevailing entity recognition LLMs rely on fine-tuned technology, yet the fine-tuning process often incurs significant cost. To achieve a best performance-cost trade-off, we propose **ALLabel**, a three-stage framework designed to select the most informative and representative samples in preparing the demonstrations for LLM modeling. The annotated examples are used to construct a ground-truth retrieval corpus for LLM in-context learning. By sequentially employing three distinct active learning strategies, ALLabel consistently outperforms all baselines under the same annotation budget across three specialized domain datasets. Experimental results also demonstrate that selectively annotating only 5%-10% of the dataset with ALLabel can achieve performance comparable to the method annotating the entire dataset. Further analyses and ablation studies verify the effectiveness and generalizability of our proposal.

1 Introduction

With the increasing prominence of AI-driven research in all science domains, many specialized scientific discovery processes demand fast and accurate textual knowledge extraction, commonly known as the named entity recognition (NER) task. As the rapid adoption of large language models (LLMs) in mainstream NLP tasks, the performance of NER has also been boosted with LLM-based solution, normally using the fine-tuning technology (Gutiérrez et al., 2022; Zheng et al., 2023; Zhang et al., 2024; Dagdelen et al., 2024). However, fine-tuning incurs significant cost due to the high computational resource required for updating model weights. To this end, entity recognition by fine-tuned LLMs becomes impractical under limited budget.

Most recently, in-context learning (ICL) introduces a paradigm shift compared with fine-tuning and is widely applied to many downstream NLP tasks (Brown et al., 2020; Dong et al., 2022). In particular, ICL with demonstration retrieval selects a small number of contextually relevant demonstrations, which are integrated into the LLM prompt and guide the model to perform inference and make predictions on the test queries for specific tasks. The demonstrations serve as examples of the task, helping LLMs to learn the input and label spaces, as well as the mapping between them (Wei et al., 2023; Pan et al., 2023). ICL is a remarkable capability of LLMs that reduces the reliance on large annotated training datasets and avoids the need for model weight update.

The latest studies have shown that the effectiveness of ICL heavily relies on the selection of demonstrations and the accuracy of ground-truth annotations (Min et al., 2022). High-quality annotations usually outperform low-quality machine-generated annotations (Liu et al., 2022). Yet, research in many fields of the natural sciences, such as chemistry and materials science, suffers from a lack of high-quality data annotations. Manual annotations in these fields demand intensive human effort and specialized domain expertise, inducing unacceptable cost (Wang et al., 2021). Therefore, selecting the most valuable subset of data for manual annotation becomes a critical challenge to improve the LLM-based entity recognition performance under the constraint of annotation budget.

To address this challenge, active learning (AL),

[*] Corresponding author

[†] Our code is available at https://github.com/Thanksyagaj/ALLabel_coding/tree/master.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 25162–25176
November 4-9, 2025 ©2025 Association for Computational Linguistics

a classical method for improving annotation efficiency through selective labeling, has garnered increasing attention (Ren et al., 2021; Diao et al., 2024; Xu et al., 2024). AL aims to achieve high model performance on a withheld test set by strategically selecting a small subset of unlabeled data for annotation. However, the current application of AL focuses primarily on tasks such as text classification and knowledge reasoning, with limited exploration in NER. Furthermore, the few existing applications in NER mostly use general-domain entity recognition datasets (e.g., CONLL (Sang and De Meulder, 2003)), which typically have lower extraction difficulty than those in specialized domains (Kazemi et al., 2023). Therefore, it is crucial to better integrate AL with LLM-based entity recognition tasks to address the challenges in specialized domains, including limited pre-trained knowledge and high accuracy requirements.

To enhance the entity extraction performance of LLMs with reduced annotation costs, we propose ALLabel, a selective annotation framework that integrates AL into the annotation process of LLMs. By leveraging three different active acquisition strategies, ALLabel empowers LLMs to retrieve the most informative and representative examples that benefit extraction performance. Assuming a limited annotation budget, the maximum number of samples that can be manually annotated is denoted as M. ALLabel adopts a three-stage workflow to construct an optimized retrieval corpus, each stage corresponding to a specific AL sampling strategy: **diversity**, **similarity**, and **uncertainty** sampling. By selecting the M most valuable unlabeled samples for annotation, ALLabel ensures efficient use of the annotation budget.

We conduct experiments on three datasets related to materials science and chemistry. The experimental results demonstrate the superiority of our proposed framework, which consistently outperforms baseline methods at each pool size. Further analysis of the varying few-shot setting confirms the universality of our method. The ablation study also reveals the contribution of each component.

The contribution of this work can be summarized as follows:

- We introduce ALLabel, a novel framework that employs LLMs as annotators for entity recognition in specialized domains, achieving superior performance compared to baseline methods with the same annotation cost.
- We conduct experiments to reveal that selectively annotating only 5%-10% subset with ALLabel can achieve the same level of extraction performance by baseline methods which annotates the entire dataset. Our result demonstrates the feasibility of selective sampling in reducing annotation costs.
- To the best of our knowledge, we are the first to integrate uncertainty, diversity, and similarity sampling strategies into a unified active learning framework for LLM in-context learning. Our ablation experiments show that the combination of the three strategies outperforms any pairwise combination.

2 Related Work

2.1 In-Context Learning with Demonstration Retrieval

In-context learning has gained considerable attention for its flexibility and performance-cost tradeoff (Brown et al., 2020; Liu et al., 2023). This approach relies on providing demonstrations in the input prompt to guide LLMs' behavior. Initially, few-shot demonstrations are randomly sampled (Zhang et al., 2022; Chung et al., 2024), which can be suboptimal especially when there are high variations among the test queries. An alternative is to retrieve demonstrations that are tailored to the current query. Previous work has shown that demonstration retrieval can lead to substantial improvement in the task metrics compared to randomly selected demonstrations (Luo et al., 2023; Ye et al., 2023).

Recent studies have explored methods to identify the most informative demonstrations. One prominent line of work employs retrieval-based strategies, using trainable retrievers to source relevant examples (Rubin et al., 2022). Some studies leverage uncertainty metrics to evaluate example utility, finding that examples with low perplexity often yield superior performance (Gonen et al., 2023). (Levy et al., 2023) indicates that diversity and coverage of the demonstrations are crucial when the model is unfamiliar with the output symbols space.

2.2 Active Learning for NLP

Active learning aims to select the most valuable samples for annotation from an unlabeled sample pool to enhance model performance with minimal annotation cost (Settles, 2009). AL has been widely used with LLMs in many NLP tasks, including text classification (Su et al., 2022; Xiao et al., 2023; Schröder et al., 2023), question reasoning (Diao et al., 2024; Snijders et al., 2023) and so on. Unlike traditional AL settings, which involves multiple iterations of data selection and model training, (Margatina et al., 2023) performs only a single iteration since no weight update is required during the process of ICL. Yet, their test tasks do not include entity recognition. Similarly to our work, (Zhang et al., 2023; Ming et al., 2024) apply AL strategies to NER, but their experiments are carried out on general-domain datasets with short text lengths (typically less than 25 tokens) and few entity types (typically 2-4 types), resulting in lower extraction difficulty than those in specialized domains.

3 Background

Given an unlabeled dataset $\mathcal{D} = \{x_1, x_2, \ldots, x_N\}$ with N samples, where each sample is an unstructured text segment containing multiple entities to be extracted, we concentrate on entity recognition tasks with LLM in-context learning. Based on the background prompt methods in prompt engineering (Dong et al., 2022), we define a natural language prompt template $T(\cdot)$, which includes the task description of entity extraction, the definition of each to-be-extracted entity, the output format specification, and the demonstrations. Our ICL prompt template is shown in Appendix A. To enhance the extraction performance, we adopt the popular setting of ICL, adaptively retrieving top k similar examples (also called k-shots) for each test query from a demonstration pool (Ram et al., 2023). Formally, given a demonstration pool $\mathcal{D}_{\text{demo}} = \{d_1, d_2, \ldots, d_n\}$ and an input text x, we can obtain k-shots by:

$$k\text{-shots} = \text{sort}\left((\text{score}(x, d_i), d_i)_{i=1}^{n}\right)[:k] \quad (1)$$

Here, the score function is used to estimate the similarity between the text of demonstration d_i and test query x. The demonstration pool $\mathcal{D}_{\text{demo}}$ is typically a subset of $\mathcal{D}$. Each demonstration in $\mathcal{D}_{\text{demo}}$ consists of two parts: a text segment x_i and a corresponding entity list y_i, which is manually annotated as the ground truth. Evidently, when the size of $\mathcal{D}_{\text{demo}}$ approaches or equals the entire dataset $\mathcal{D}$, the performance of LLM-based few-shot entity recognition tends to reach its optimum since a larger demonstration pool provides broader coverage and enables the selection of more relevant examples for each test query, thereby improving the model's generalization capability. However, under the constraint of a limited annotation budget, it becomes crucial to strategically select the most valuable samples from $\mathcal{D}$ for manual annotation. These annotated samples serve as demonstrations to guide the annotation of the remaining unlabeled samples in $\mathcal{D}$, striking a balance between annotation cost and extraction performance.

4 ALLabel

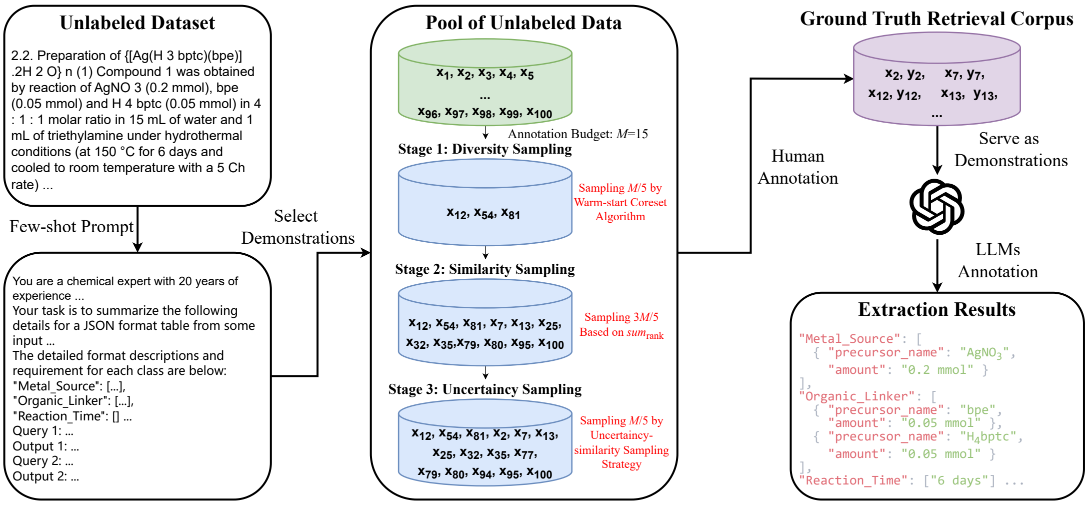
In this section, we introduce our proposed framework ALLabel. By adaptively selecting a small subset $\mathcal{D}_{\text{selected}}$ of the entire unlabeled dataset $\mathcal{D}$, which includes the most informative and representative samples, ALLabel can reduce the annotation cost while retaining the extraction performance of LLMs. Given the annotation budget M, where $M \ll N$, ALLabel iterates through three stages to gradually construct the retrieval corpus $\mathcal{D}_{\text{selected}}$, corresponding to three different active learning strategies: *diversity sampling*, *similarity sampling*, and *uncertainty sampling*. The overall pipeline of ALLabel is shown in Figure 1.

4.1 Diversity Sampling

Algorithm 1 Warm-start Core-Set Selection Using Text Similarity

```
Input: Dataset $\mathcal{D}$, annotation budget $M/5$
Output: Selected subset $\mathcal{D}_{\text{selected}}$
  Initialize: $\mathcal{D}_{\text{selected}} \leftarrow \emptyset$
  Compute similarity matrix $U$ for $\mathcal{D}$
  repeat
      if $|\mathcal{D}_{\text{selected}}| = 0$ then
          $u \leftarrow \arg\min_{i \in \mathcal{D}} \frac{1}{|\mathcal{D}|-1} \sum_{j \in \mathcal{D}, j \neq i} U(x_i, x_j)$
      else
          $u \leftarrow \arg\max_{i \in \mathcal{D}} \min_{j \in \mathcal{D}_{\text{selected}}} (1 - U(x_i, x_j))$
      end if
      $\mathcal{D}_{\text{selected}} \leftarrow \mathcal{D}_{\text{selected}} \cup \{u\}$
      $\mathcal{D} \leftarrow \mathcal{D} \setminus \{u\}$
  until $|\mathcal{D}_{\text{selected}}| = M/5$
```

The first stage of ALLabel is diversity sampling, based on the intuition that a representative subset of examples can act as a surrogate for the full data. As shown in Algorithm 1, we design a **warm-start** core-set selection algorithm based on text similarity to select the initial $M/5$ samples, ensuring the representativeness of the demonstrations at the initial stage of sampling.


Unlabeled Dataset
Few-shot Prompt
Select Demonstrations
Pool of Unlabeled Data
x1, x2, x3, x4, x5
...
x96, x97, x98, x99, x100
Annotation Budget: M=15
Stage 1: Diversity Sampling
x12, x54, x81
Sampling M/5 by Warm-start Coreset Algorithm
Stage 2: Similarity Sampling
x12, x54, x81, x7, x13, x25, x32, x35,x79, x80, x95, x100
Sampling 3M/5 Based on sumrank
Stage 3: Uncertaincy Sampling
x12, x54, x81, x2, x7, x13, x25, x32, x35, x77, x79, x80, x94, x95, x100
Sampling M/5 by Uncertaincy-similarity Sampling Strategy
Human Annotation
Ground Truth Retrieval Corpus
x2, y2, x7, y7, x12, y12, x13, y13,
...
Serve as Demonstrations
LLMs Annotation
Extraction Results

2.2. Preparation of {[Ag(H 3 bptc)(bpe)] .2H 2 O} n (1) Compound 1 was obtained by reaction of AgNO 3 (0.2 mmol), bpe (0.05 mmol) and H 4 bptc (0.05 mmol) in 4 : 1 : 1 molar ratio in 15 mL of water and 1 mL of triethylamine under hydrothermal conditions (at 150 °C for 6 days and cooled to room temperature with a 5 Ch rate) ...

You are a chemical expert with 20 years of experience ...
Your task is to summarize the following details for a JSON format table from some input ...
The detailed format descriptions and requirement for each class are below:
"Metal_Source": [...],
"Organic_Linker": [...],
"Reaction_Time": [] ...
Query 1: ...
Output 1: ...
Query 2: ...
Output 2: ...

```
"Metal_Source": [
  { "precursor_name": "AgNO3",
    "amount": "0.2 mmol" }
],
"Organic_Linker": [
  { "precursor_name": "bpe",
    "amount": "0.05 mmol" },
  { "precursor_name": "H4bptc",
    "amount": "0.05 mmol" }
],
"Reaction_Time": ["6 days"] ...
```

Figure 1: ALLabel combines human annotation with LLM annotation in an active learning workflow, which consists of three stages for selecting a small number of unlabeled examples to annotate under the annotation budget M:(1) diversity sampling, (2) similarity sampling, and (3) uncertainty sampling. Then LLMs can retrieve informative and representative demonstrations from the human-annotated retrieval corpus.

Regarding each sample as a point, the algorithm uses a greedy strategy to sequentially add the point with the largest distance from the labeled set $\mathcal{D}_{\text{selected}}$ to it. Given that distance is inversely correlated with similarity, we use text similarity as a surrogate measure for distance, which can be computed with methods such as BM25 or Sentence-BERT (Robertson et al., 2009; Reimers, 2019). The distance between an unlabeled sample u and $\mathcal{D}_{\text{selected}}$ is defined as the minimum distance (i.e., the maximum similarity) between u and all samples in $\mathcal{D}_{\text{selected}}$. Specifically, we compute the text similarity for each sample in the whole dataset $\mathcal{D}$ with all others, resulting in an $N \times N$ similarity matrix U, where diagonal elements are meaningless. Next, we select the sample with the lowest average similarity to others and move it to $\mathcal{D}_{\text{selected}}$ as the seed data. We iteratively execute the algorithm until $|\mathcal{D}_{\text{selected}}| = M/5$. It is noteworthy that we enhance the traditional core-set algorithm (Sener and Savarese, 2018) by identifying and fixing representative seed data rather than randomly selecting, thereby avoiding the randomness and low performance caused by the cold start problem (Wei et al., 2024). The discussion of this improvement is available in Appendix D.1.

4.2 Similarity Sampling

The second stage of ALLabel is similarity sampling, which leverages the fact that in-context learning performs optimally when the most similar demonstrations are adaptively retrieved for each test input. Due to the limited annotation budget, not all optimal demonstrations for the test inputs can be annotated. As an alternative, we define a composite similarity metric sum_{rank} for each sample, which represents the overall similarity between a sample and all others in the unlabeled dataset $\mathcal{D}$. In this section, we elaborate on our similarity sampling strategy based on sum_{rank}.

After the diversity sampling stage, $M/5$ samples have been removed from $\mathcal{D}$ to $\mathcal{D}_{\text{selected}}$. Note that the selected $M/5$ samples can still be used as test inputs. The similarity matrix U mentioned in Section 4.1 thereby becomes $N \times (N - M/5)$, denoted as S, where each row represents a similarity dictionary of a test query, and each column corresponds to a demonstration. Initially, set the values of sum_{rank} to 0 for each sample. Assume sample b serves as a demonstration for a test query a. We apply the following heuristic rule to increment the sum_{rank} of b:

$$sum_{\text{rank}}[b] += \begin{cases} 1, & \text{rank} \le k, \\ \dfrac{1}{\text{rank} - k + 1}, & k < \text{rank} \le x, \\ 0, & \text{rank} > x \end{cases} \tag{2}$$

Here, k represents the number of examples retrieved for each test query in the ICL setting, x denotes the sampling size at the similarity sampling stage ($x = 3M/5$ and $x > k$), and rank refers to the position of sample b in the similarity dictionary of a, sorted in descending order of text similarity.

Specifically, when rank is between 1 and k, we consider b as one of the best demonstrations for the test query a, and the increment to sum_{rank} is correspondingly higher. When rank is between $k+1$ and x, sample b still holds sampling value since it retains some similarity to the test query. As b ranks lower, its sampling value decreases, and so does the increment to sum_{rank}. When rank exceeds x, sample b is considered to have no further sampling value for a without any scoring operation.

By applying the above process to traverse the similarity dictionaries of all test queries, we obtain the composite similarity score sum_{rank} for each sample. The top $3M/5$ samples with the highest sum_{rank} values are then selected to complement the retrieval corpus $\mathcal{D}_{\text{selected}}$.

4.3 Uncertainty Sampling

The third stage of ALLabel is uncertainty sampling, selecting samples that LLMs predict with low confidence. Previous research on uncertainty sampling strategies mostly relies on metrics such as maximum entropy and minimum confidence (Settles, 2009; Culotta and McCallum, 2005). However, since the pre-trained LLMs used in our work have not been fine-tuned with specific classification layers, we are unable to compute model probabilities associated with each class. As a result, the above metrics cannot be used to assess the uncertainty of samples in our work. To adopt the uncertainty metric in entity recognition task, we propose an uncertainty-similarity sampling strategy, which means that examples with lower similarity lead to higher uncertainty in the LLM's predictions for test queries. We conducted experiments to indicate the negative correlation between similarity score and LLM prediction uncertainty (measured by perplexity). The experimental results can be seen in Appendix C, which explain the rationality of our uncertainty-similarity sampling strategy.

After the first two stages of sampling, $\mathcal{D}_{\text{selected}}$ has already contained $4M/5$ samples. We denote the current retrieval corpus as $\mathcal{D}_1$, and select the final $M/5$ samples (denoted as $\mathcal{D}_2$) to complete the full $\mathcal{D}_{\text{selected}}$. The final stage begins by identifying weak test points. For each test query in the similarity matrix S mentioned in Section 4.2, we find the sample with the highest similarity in $\mathcal{D}_1$ and record the ranking index of this sample. The test queries are then sorted in descending order based on these indices, and the top $M/5$ queries are selected as weak test points, representing data points that retrieve the least similar demonstrations from $\mathcal{D}_1$. These data points exhibit the highest uncertainty in LLMs' few-shot extraction, thus requiring special attention. The next step is to filter S, retaining only the rows corresponding to the weak test points while removing entries that overlap with the selected samples in $\mathcal{D}_1$. We then reapply the similarity sampling strategy described in Section 4.2 on the filtered matrix S', selecting the top $M/5$ samples with the highest sum_{rank} values to form the set $\mathcal{D}_2$. Finally, the complete retrieval corpus $\mathcal{D}_{\text{selected}}$ is combined by merging $\mathcal{D}_1$ and $\mathcal{D}_2$.

Dataset	Domain	Size	Entity Types	F1(%)
CSD-MOFs	Materials	696	10	94.4
NC 2024 General	Materials	549	6	85.8
USPTO	Chemistry	498	10	90.8

Table 1: Dataset statistics for entity recognition tasks in specialized domains. **Size** is the total number of samples in the dataset. **Entity types** refers to the number of entity types to be extracted in a sample. **F1(%)** is the average F1-score of all entity types with the entire dataset as the retrieval corpus.

5 Experiment

5.1 Setup

Datasets We experiment on three entity extraction datasets across materials science and chemistry, since research in both fields requires large-scale, high-quality data annotation efforts: (1) CSD-MOFs (Zhang et al., 2024), which extracts ten kinds of synthesis conditions in the synthesis paragraphs of Metal-Organic Frameworks (MOFs), such as reaction temperature and time. (2) NC 2024 General (Dagdelen et al., 2024), which labels the properties and applications of a variety of general materials. (3) USPTO (Vangala et al., 2024), which extracts chemical reaction entities including reactants, solvents, catalysts, etc., along with their quantities from reaction paragraphs. All three datasets are manually annotated by experts in their respective fields. As a representation of extraction difficulty for each dataset above, we report the F1-score taking the full dataset as the retrieval corpus, which employs a 3-shot ICL setting with GPT-4o (OpenAI, 2024) as the annotator. Detailed information of the datasets is listed in Table 1.

Baselines In our experiments, we compare ALLabel with the following baselines: (1) Random, which samples to-be-labeled data randomly. (2) Core-set (Sener and Savarese, 2018), which selects to-be-labeled data using the core-set approach with a **cold start**, meaning the seed data is randomly initialized. We choose this baseline as the representative of the traditional diversity sampling strategy. (3) Perplexity (Gonen et al., 2023), referring to the strategy of prioritizing samples for annotation based on the average perplexity of the LLM's predictions across different entity types. Samples with high perplexity present greater difficulty in few-shot extraction. (4) BATCHER (Fan et al., 2024), a covering-based demonstration selection strategy with question batching, which is initially used in entity resolution.

Implementation We adopt GPT-4o as the LLM annotator and use BM25 (Robertson et al., 2009) to compute text similarity in our main experiments. F1-score is used as the evaluation metric. Note that we use the in-domain setting, which means that each sample can be regarded as both a test query and an example except for itself, so the entire dataset can be seen as the test set (Luo et al., 2023). That is, each sample in the three datasets has been manually annotated. When a sample is used as the test query, we consider it unlabeled and compute F1-score by comparing the LLM extraction result with its ground-truth annotation. We ensure that each test query is evaluated independently, without interference from others. The detailed calculation rules of similarity and F1-score can be found in Appendix B.

In our main experiments, all the ICL prompts consist of $k = 3$ in-context examples as demonstrations, retrieving the most similar examples for each test query from the retrieval corpus $\mathcal{D}_{\text{selected}}$ except itself. The analysis of extraction performance *vs.* k is discussed in Section 6.1. The size of the demonstration pool is set to span from 10 to 60, with experiments conducted at intervals of 5. Note that given a set of unlabeled data, ALLabel is deterministic without any randomness, which is also an advantage of our proposed framework. For fair comparison, we repeat five separate runs for random sampling and core-set sampling, as both baselines involve a degree of randomness. We report the mean and standard deviation for the two baselines in the results.

5.2 Results

The experimental results are displayed in Table 2, which compare the extraction performance of ALLabel and baselines at different pool sizes from 10 to 60. Overall, our framework consistently outperforms all baselines across all datasets by a large margin. For instance, ALLabel achieves superior results with an average of 5.51%, 5.78%, and 5.12% improvement over random sampling on the CSD-MOFs, NC 2024 General, and USPTO dataset, respectively, which showcases the effectiveness of our proposed selective annotation framework. Moreover, ALLabel is a deterministic method without standard deviation, significantly enhancing the robustness of ICL.

As shown in Table 2, ALLabel achieves performance comparable to using the entire dataset as the retrieval corpus, significantly enhancing data efficiency and reducing the annotation budget. Specifically, ALLabel selects only 5.0%, 9.1%, and 8.0% samples for manual annotation on the CSD-MOFs, NC 2024 General, and USPTO datasets respectively, achieving performance within 2% of that obtained by annotating the entire dataset shown in Table 1. The comparison of the proportion required for the convergence of F1 with different methods is shown in Table 3. On average, ALLabel reduces annotation budget by 9.1% compared to the best alternative, while achieving the same performance. We can infer from Table 3 that selectively annotating 5% to 10% of the data with ALLabel can achieve performance close to that of annotating the entire dataset, thereby demonstrating its favorable trade-off between performance and cost. Interestingly, we observe that this proportion is inversely correlated with F1 measured with the entire dataset as the retrieval corpus shown in Table 1, which may indicate that the higher the extraction difficulty of the dataset, the larger the proportion required for performance convergence.

In this work, we primarily focus on the performance of ALLabel in NER task. To further validate the generalizability of ALLabel, we conduct additional experiments on other NLP tasks and with more LLM annotators. Detailed experimental results are provided in Appendix D.2 and D.3.

6 Analysis

6.1 Impact of Different Number of Shots

To verify the generalizability of ALLabel with different numbers of shots k during the process of ICL, we conduct $\{1, 3, 5, 7\}$-shot extraction experiments on a randomly selected $1/5$ subset of CSD-MOFs dataset, considering the cost. As shown in Figure 2, we plot the experimental results as heat maps to visually demonstrate how the F1-score of ALLa-

Method	10	15	20	25	30	35	40	45	50	55	60
					CSD-MOFs						
Random	$83.2_{1.9}$	$84.8_{2.1}$	$85.1_{2.0}$	$85.6_{1.3}$	$86.9_{1.7}$	$87.5_{1.7}$	$87.3_{1.9}$	$87.0_{1.7}$	$87.2_{2.2}$	$88.8_{2.1}$	$89.6_{2.4}$
Core-set	$86.1_{0.7}$	$86.5_{0.7}$	$87.2_{0.8}$	$86.9_{1.2}$	$87.9_{0.9}$	$88.2_{1.1}$	$88.5_{1.0}$	$89.3_{0.6}$	$89.8_{0.5}$	$90.5_{0.6}$	$90.4_{0.6}$
Perplexity	84.1	84.3	84.4	84.9	85.5	87.0	87.6	88.1	88.9	89.9	91.5
BATCHER	85.2	85.6	86.3	87.1	87.5	87.3	87.9	88.4	88.7	90.2	91.0
ALLabel	**87.8**	**89.4**	**91.0**	**91.2**	**91.5**	**92.5**	**92.2**	**91.8**	**91.9**	**92.8**	**93.3**
					NC 2024 General						
Random	$75.0_{2.5}$	$75.9_{1.8}$	$77.1_{2.0}$	$76.6_{1.1}$	$76.7_{1.0}$	$77.9_{2.5}$	$79.5_{1.6}$	$78.9_{0.8}$	$79.8_{2.0}$	$79.9_{1.4}$	$80.8_{1.9}$
Core-set	$77.0_{0.9}$	$78.5_{0.9}$	$78.7_{0.5}$	$78.1_{1.0}$	$78.9_{0.4}$	$79.9_{0.5}$	$79.0_{0.5}$	$80.0_{0.9}$	$79.6_{1.0}$	$80.5_{0.7}$	$80.3_{0.5}$
Perplexity	75.5	76.8	76.5	77.0	77.2	77.7	78.5	79.1	80.7	81.6	82.0
BATCHER	75.1	76.0	76.5	76.9	76.5	77.1	77.9	78.8	79.1	79.7	80.8
ALLabel	**78.6**	**79.5**	**80.6**	**82.1**	**83.4**	**83.4**	**82.8**	**83.8**	**84.6**	**84.3**	**84.8**
					USPTO						
Random	$80.7_{2.0}$	$82.1_{1.5}$	$82.4_{2.2}$	$83.6_{0.7}$	$82.9_{1.2}$	$84.2_{1.9}$	$84.1_{1.3}$	$84.3_{0.9}$	$84.8_{2.0}$	$85.0_{1.5}$	$85.3_{1.8}$
Core-set	$82.6_{1.1}$	$83.2_{0.3}$	$83.4_{1.1}$	$82.8_{0.8}$	$83.4_{0.8}$	$83.2_{0.6}$	$83.6_{1.0}$	$84.0_{0.3}$	$84.3_{0.8}$	$84.7_{0.9}$	$85.6_{0.3}$
Perplexity	80.4	81.8	82.0	82.5	83.5	84.8	84.2	84.5	85.6	86.1	86.4
BATCHER	83.0	83.3	83.8	84.4	84.0	85.1	85.2	84.7	85.4	85.7	86.1
ALLabel	**86.0**	**85.8**	**86.5**	**87.3**	**86.8**	**88.1**	**88.9**	**88.9**	**89.2**	**89.4**	**89.7**

Table 2: Extraction performance of ALLabel and other baselines across three datasets using GPT-4o as the LLM annotator, where the demonstration pool size spans from 10 to 60. We report the mean and standard deviation of five separate runs for Random and Core-set. The highest F1-scores are highlighted in **bold**.

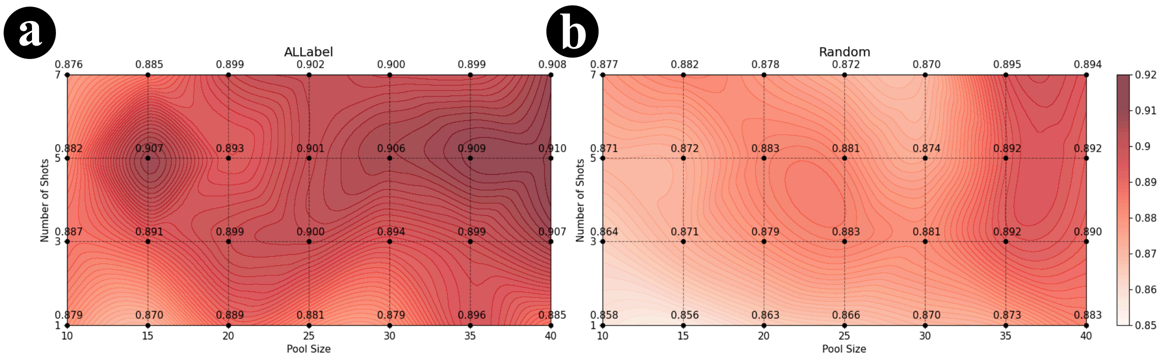


Figure 2: F1-scores of (a) ALLabel and (b) Random Sampling varying with demonstration pool size and number of shots, shown in the form of heat maps.

Dataset	Random	Core-set	Perp	BATCHER	ALLabel
CSD-MOFs	25.1	18.3	14.3	15.1	**5.0**
NC 2024 General	30.7	22.8	17.3	19.0	**9.1**
USPTO	27.2	24.9	18.1	17.9	**8.0**

Table 3: Comparsions of the proportion required for the convergence of F1 between ALLabel and other baselines. Here, *convergence* refers to the point at which the performance gap compared to using the entire dataset as the retrieval corpus first narrows to within 2%.

bel and the random sampling baseline vary with demonstration pool size and number of shots. It is evident that each point in Figure 2(a) consistently lies above the corresponding point in Figure 2(b), confirming that our proposed ALLabel framework outperforms random sampling across different pool sizes and numbers of shots. In addition, we can observe that with a demonstration pool size no smaller than 20, F1 becomes indistinguishable with shot settings of $k \geq 5$. This indicates that the extraction performance of ALLabel does not increase monotonically with the number of shots but instead converges at some point. In this case, F1 tends to peak at $k = 5$. Our finding suggests that the ICL performance of LLMs improves with more examples in the prompt up to a point, and then fluctuates within a small range.

6.2 Impact of Multiple Entity Types

In this section, we investigate the impact of having multiple entity types to be extracted from a single text. It can be observed from Figure 2 that an increase in demonstration pool size may lead to a decrease in F1 in some cases, which is coun-

Method	10	15	20	25	30	35	40	45	50	55	60
ALLabel	**87.8**	**89.4**	**91.0**	**91.2**	**91.5**	**92.5**	**92.2**	**91.8**	**91.9**	**92.8**	**93.3**
without uncertainty sampling	87.5	88.9	89.7	89.5	90.4	91.2	91.6	91.6	91.3	91.8	92.1
without similarity sampling	86.2	86.6	87.7	88.0	88.4	87.9	88.3	89.4	90.0	89.9	90.2
without diversity sampling	87.0	88.0	89.2	90.1	90.7	91.4	91.2	91.4	91.7	92.3	92.1

Table 4: Ablation study of ALLabel on CSD-MOFs dataset. The lowest F1-scores are highlighted with underlines.

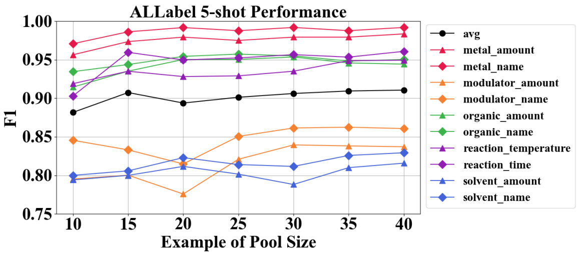


Figure 3: ALLabel's 5-shot extraction performance on a randomly selected 1/5 subset of CSD-MOFs dataset, showing the F1 for each entity type and their average.

terintuitive. To understand this phenomenon, we analyze the F1 curves with varying pool sizes for each entity type. Taking ALLabel's 5-shot F1 as an example, Figure 3 shows that the average F1 across ten entity types decreases by 1.4% when the pool size increases from 15 to 20. Specifically, for each entity type, six entities show an improvement in F1, while the remaining four entities experience a decrease. The notable decline of two modulator-related entity types contributes significantly to the overall decrease in this interval. That is to say, expanding the demonstration pool inevitably results in retrieving examples that benefit the extraction for some entity types, while negatively impacting others. When this negative impact outweighs the positive, F1 tends to decrease within a certain range.

As discussed in Section 5.2, the extraction performance of ALLabel does not continuously improve with increasing pool size. Instead, it tends to converge at approximately 5%-10% of the entire dataset. After that, F1 grows slowly and fluctuates within a small range, indicating that the benefits of annotating additional examples diminish significantly. Therefore, selective annotation is a viable approach to achieving a remarkable extraction performance-cost trade-off.

6.3 Ablation Study

ALLabel is a three-stage framework that uses three different AL strategies to select the most informative samples for annotation. We reveal the effect of the three components by ablating them one by one. Specifically, we conduct ablation experiments on CSD-MOFs dataset under the constraint of annotation budget M, comparing the whole process of ALLabel with the following benchmarks, each of which lacks one of the sampling stages: (1) Selecting $M/5$ samples through diversity sampling, followed by $4M/5$ through similarity sampling; (2) Selecting $4M/5$ through diversity sampling, followed by $M/5$ through uncertainty sampling; (3) Selecting $4M/5$ through similarity sampling, followed by $M/5$ through uncertainty sampling. As shown in Table 4, the experimental results indicate that the combination of three strategies consistently outperforms any pairwise combination of them across pool sizes ranging from 10 to 60, thereby demonstrating the indispensability of each component in ALLabel. We also find that the extraction performance without similarity sampling is the lowest among all benchmarks. This not only demonstrates the importance of the similarity sampling stage compared to the other two stages, but also further validates the effectiveness of our proposed similarity sampling strategy based on sum_{rank}.

Additionally, we conduct ablation experiments on the sampling order and proportion of the three stages, which confirm that the sequence (D-S-U) and division proportion (1:3:1) employed by ALLabel are both optimal. Detailed results and analysis can be found in Appendix D.4 and D.5.

7 Conclusion

In this work, we propose ALLabel, a novel framework that leverages LLMs as efficient and accurate annotators for entity recognition tasks in specialized domains. By adopting diversity, similarity, and uncertainty sampling strategies sequentially, ALLabel selects the most informative and representative samples to label for human annotators and constructs an optimized retrieval corpus for LLM annotators, effectively combining the supervision of human experts with the generalization capability of LLMs. We conduct experiments on three datasets. The experimental results demonstrate that ALLabel can achieve a remarkable performance-cost trade-off while consistently outperforming the baseline methods at the same annotation cost. Furthermore, we find that selectively annotating only 5%-10% data with ALLabel can achieve extraction performance comparable to annotating the entire dataset, revealing the feasibility of selective annotation. Further analyses and ablation studies verify the effectiveness and generalizability of ALLabel.

Limitations

We have shown that ALLabel demonstrates superior performance over previous active learning methods. Despite its effectiveness, there are still some limitations in the current work for improvement. First, due to cost and time constraints, we were only able to conduct experiments with GPT-4o and DeepSeek-V3. As a future direction, we plan to explore the use of more advanced models to further validate the effectiveness of our method on more datasets. In addition, using the number of samples as an estimate of the annotation budget may be somewhat rough since different texts vary in length and extraction difficulty. Although using LLMs as annotators is automated and convenient, it still incurs certain annotation costs. In future work, we will measure the annotation budget more precisely in terms of token-level charges, considering the costs generated by LLMs.

Acknowledgments

This work was supported by National Key R&D Program of China (2021YFB3500700), NSFC Grant 62172026, 62572026, National Social Science Fund of China 22&ZD153, the Fundamental Research Funds for the Central Universities, State Key Laboratory of Complex & Critical Software Environment (SKLCCSE), National Natural Science Foundation of China (NSFC No.62161160339).

Lei Shi is with School of Computer Science and Engineering, Beihang University, and the State Key Laboratory of Complex & Critical Software Environment. Lei Shi is the corresponding author.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Aron Culotta and Andrew McCallum. 2005. Reducing labeling effort for structured prediction tasks. In *AAAI*, volume 5, pages 746–751.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S Rosen, Gerbrand Ceder, Kristin A Persson, and Anubhav Jain. 2024. Structured information extraction from scientific text with large language models. *Nature Communications*, 15(1):1418.

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

Shizhe Diao, Pengcheng Wang, Yong Lin, Rui Pan, Xiang Liu, and Tong Zhang. 2024. Active prompting with chain-of-thought for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 1330–1350.

Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third international workshop on paraphrasing (IWP2005)*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, and 1 others. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Meihao Fan, Xiaoyue Han, Ju Fan, Chengliang Chai, Nan Tang, Guoliang Li, and Xiaoyong Du. 2024. Cost-effective in-context learning for entity resolution: A design space exploration. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 3696–3709. IEEE.

Hila Gonen, Srini Iyer, Terra Blevins, Noah A Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in language models via perplexity estimation. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Bernal Jiménez Gutiérrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about gpt-3 in-context learning for biomedical ie? think again. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4497–4512.

Mehran Kazemi, Sid Mittal, and Deepak Ramachandran. 2023. Understanding finetuning for factual knowledge extraction from language models. *arXiv preprint arXiv:2301.11293*.

Itay Levy, Ben Bogin, and Jonathan Berant. 2023. Diverse demonstrations improve in-context compositional generalization. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Man Luo, Xin Xu, Zhuyun Dai, Panupong Pasupat, Mehran Kazemi, Chitta Baral, Vaiva Imbrasaite, and Vincent Y Zhao. 2023. Dr. icl: Demonstration-retrieved in-context learning. *arXiv preprint arXiv:2305.14128*.

Katerina Margatina, Timo Schick, Nikolaos Aletras, and Jane Dwivedi-Yu. 2023. Active learning principles for in-context learning with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5011–5034.

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, EMNLP 2022*, pages 11048–11064.

Xuran Ming, Shoubin Li, Mingyang Li, Lvlong He, and Qing Wang. 2024. Autolabel: Automated textual data annotation method based on active learning and large language model. In *Knowledge Science, Engineering and Management - 17th International Conference, KSEM 2024*, volume 14887, pages 400–411.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. 2023. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8298–8319.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. 2021. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671.

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

Christopher Schröder, Lydia Müller, Andreas Niekler, and Martin Potthast. 2023. Small-text: Active learning for text classification in python. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 84–95.

Ozan Sener and Silvio Savarese. 2018. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*.

Burr Settles. 2009. Active learning literature survey.

Ard Snijders, Douwe Kiela, and Katerina Margatina. 2023. Investigating multi-source active learning for natural language inference. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2187–2209.

Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and 1 others. 2022. Selective annotation makes language models better few-shot learners. *arXiv preprint arXiv:2209.01975*.

Sarveswara Rao Vangala, Sowmya Ramaswamy Krishnan, Navneet Bung, Dhandapani Nandagopal, Gomathi Ramasamy, Satyam Kumar, Sridharan Sankaran, Rajgopal Srinivasan, and Arijit Roy. 2024. Suitability of large language models for extraction of high-quality chemical reaction dataset from patent literature. *Journal of Cheminformatics*, 16(1):131.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? gpt-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and 1 others. 2023. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*.

Jiarong Wei, Yancong Lin, and Holger Caesar. 2024. Basal: Size-balanced warm start active learning for lidar semantic segmentation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 18258–18264.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2018*, pages 1112–1122. Association for Computational Linguistics (ACL).

Ruixuan Xiao, Yiwen Dong, Junbo Zhao, Runze Wu, Minmin Lin, Gang Chen, and Haobo Wang. 2023. Freeal: Towards human-free active learning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14520–14535.

Zhipeng Xu, Zhenghao Liu, Yibin Liu, Chenyan Xiong, Yukun Yan, Shuo Wang, Shi Yu, Zhiyuan Liu, and Ge Yu. 2024. Activerag: Revealing the treasures of knowledge via active learning. *arXiv preprint arXiv:2402.13547*.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2023. Compositional exemplars for in-context learning. In *International Conference on Machine Learning*, pages 39818–39833. PMLR.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023. Llmaaa: Making large language models as active annotators. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13088–13103.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, and 1 others. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Wei Zhang, Qinggong Wang, Xiangtai Kong, Jiacheng Xiong, Shengkun Ni, Duanhua Cao, Buying Niu, Mingan Chen, Yameng Li, Runze Zhang, and 1 others. 2024. Fine-tuning large language models for chemical text mining. *Chemical Science*, 15(27):10600–10611.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling. 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 1298–1308.

Zhiling Zheng, Ali H Alawadhi, Saumil Chheda, S Ephraim Neumann, Nakul Rampal, Shengchao Liu, Ha L Nguyen, Yen-hsu Lin, Zichao Rong, J Ilja Siepmann, and 1 others. 2023. Shaping the water-harvesting behavior of metal–organic frameworks aided by fine-tuned gpt models. *Journal of the American Chemical Society*, 145(51):28284–28295.

A ICL Prompt Template

As shown in Table 5, we design an ICL prompt template for LLM entity extraction tasks, which consists of the following sections: <Role Description, Task Description, Background Information, Format, Example>. By adopting the prompt template, LLMs can leverage external knowledge to accurately understand the entities to be extracted.

B Calculation Rules of Extraction Metrics

B.1 Measurement Methods of Text Similarity

We use BM25 as the retriever to compute the similarity score between each test query and example in the main experiments. BM25 is a probabilistic information retrieval model that ranks documents based on the frequency of query terms within the documents. It balances term frequency (how often a term appears in a document) with inverse document frequency (how rare a term is across the entire document set), thus giving more weight to terms that are significant.

The scoring function of BM25 between a paragraph with n terms and a document in a pool of length N is defined as:

$$\text{Score}(p,d) = \sum_{i=1}^{n} \text{IDF}(p_i) \cdot \frac{f(p_i,d)\cdot(k_1+1)}{f(p_i,d)+k_1\left(1-b+b\cdot\frac{|d|}{\text{avg\_dl}}\right)} \tag{3}$$

where $f(p_i, d)$ is the term frequency of p_i in document d, $|d|$ is the length of document d, avg_dl is the average length of all documents in the demonstration pool, IDF(p_i) is the inverse document frequency of term p_i, and k_1 and b are hyperparameters of the model.

To demonstrate the generalizability of ALLabel across different text similarity measurement methods, we also perform complementary experiment using Sentence-BERT as the retriever and observe consistent trends from Table 6, which suggests that ALLabel is robust to variations of similarity measurement methods. We observe that using Sentence-BERT as the retriever yields slightly lower performance compared to BM25, which is why we use BM25 as the retriever in the main experiments shown in Table 2.

B.2 Calculation Rules of F1-score

In our experiments, we primarily use the F1-score to evaluate the extraction performance of LLMs.

Section	**Content**
Role Description	You are a chemical expert with 20 years of experience in reviewing literature and extracting key information. Your expertise lies in systematically and accurately extracting synthesis parameters from chemical literature, focusing on MOFs (Metal-Organic Frameworks) synthesis sectpions ...
Task Description	Your task is to summarize the following details for a JSON format table from some input: "Compound_Name", "Metal_Source", "Organic_Linker", "Solvent", "Modulator", "Reaction_Time", "Reaction_Temperature". Among them, "Metal_Source", "Organic_Linker", "Solvent", and "Modulator" should also contain their amounts ...
Background Information	Background Information and Detailed Instructions: Compound_Name of MOFs (Metal-Organic Frameworks): MOFs are porous materials formed by the coordination of metal ions or clusters with organic ligands. They exhibit a high surface area and ...
Format	The detailed format descriptions and requirement for each class are below: The output should be a JSON table list. Each JSON format table represents a MOF ...
Example	**Input 1:** 2.2. Preparation of [Ag(H 3 bptc)(bpe)] .2H 2 O n (1) Compound 1 was obtained by reaction of AgNO 3 (0.2 mmol), bpe (0.05 mmol) and H 4 bptc (0.05 mmol) ... **Output 1:** [{"Metal_Source": [{"precursor_name": "AgNO3","amount": "0.2 mmol" }], ...}] **Input 2:** ... **Output 2:** ... ...

Table 5: Prompt template used for LLM entity extraction tasks, exemplified by CSD-MOFs dataset.

Method	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**	**55**	**60**
Random (SBERT)	$81.8_{1.1}$	$83.3_{1.2}$	$84.6_{0.7}$	$84.9_{0.5}$	$85.2_{0.7}$	$85.5_{0.6}$	$86.3_{1.2}$	$84.9_{0.7}$	$85.4_{1.0}$	$88.0_{0.9}$	$88.1_{0.5}$
Core-set (SBERT)	$85.0_{1.0}$	$85.8_{0.8}$	$85.3_{1.2}$	$84.7_{0.9}$	$87.1_{0.8}$	$87.6_{0.7}$	$87.0_{0.5}$	$87.5_{0.9}$	$88.9_{1.0}$	$88.8_{0.5}$	$89.0_{0.4}$
Perplexity (SBERT)	83.6	83.6	83.6	82.6	84.1	85.5	85.8	86.7	86.8	88.0	90.3
ALLabel (SBERT)	**86.7**	**87.2**	**89.2**	**89.7**	**90.8**	**90.8**	**91.7**	**90.9**	**91.0**	**91.5**	**92.4**

Table 6: Comparison of F1-scores on the CSD-MOFs dataset using Sentence-BERT similarity (mean$_{\text{std}}$ for stochastic methods). We report the mean and standard deviation of five separate runs for Random and Core-set. The highest F1-scores are highlighted in **bold**.

The calculation follows the standard definitions of precision, recall, and F1-score, based on the counts of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

Due to the structured nature of JSON-formatted data, it is crucial to clearly define TP, FP, TN, and FN in our work:

- TP: The ground-truth label and the LLM-generated output are both non-empty and match exactly in both precursor name and amount.
- FP: The LLM-generated output is non-empty but does not exactly match the ground-truth label.
- TN: Both the ground-truth label and the LLM-generated output are empty.
- FN: The ground-truth label is non-empty, but the LLM-generated output is empty.

Following these calculation rules, we compute F1-scores for each entity type across all samples in the three datasets mentioned in Section 5.1. The final dataset-level F1-score is obtained by averaging the sample-level F1, which are themselves calculated by averaging the F1 of all entity types. It should be noted that a DOI may link to multiple target products (i.e., one paper reporting more than one target product). For the convenience of follow-up processing, we perform deduplication on the three datasets, allowing us to focus on the DOIs associated with publications that report the information of only one target product.

C Relationship between Similarity and Uncertainty

We conduct additional experiments to indicate the negative correlation between similarity score and LLM prediction uncertainty (measured by perplexity). Perplexity is a widely used metric to quantify the uncertainty of probabilistic language models. It measures how well a model predicts a sequence of tokens by evaluating the inverse probability of the test set, normalized by the number of tokens. In other words, the model has insufficient confidence in test queries with high perplexity, so perplexity can serve as an indicator of the extraction difficulty of a test query. The formula for perplexity $PP(W)$ over a sequence of tokens $W = (w_1, w_2, \ldots, w_N)$ is defined as:

$$PP(W) = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log P(w_i|w_1,\ldots,w_{i-1})\right) \quad (4)$$

To illustrate the rationale behind the uncertainty-similarity strategy, we randomly select 50 samples as test queries from each of the three datasets, while the remaining samples automatically form the demonstration pool. The experimental result is shown in Table 7, where **Sim.** is the average similarity scores between 3-shot demonstration and test query. From the experimental results, we observe that as the similarity between the demonstration and the test query increases, the perplexity of the LLM's prediction on the test query **Perp.** consistently decreases.

CSD-MOFs		NC 2024 General		USPTO	
Sim.	**Perp.** ↓	**Sim.**	**Perp.** ↓	**Sim.**	**Perp.** ↓
0.35	28.6	0.31	32.1	0.29	30.9
0.48	22.3	0.47	25.7	0.46	24.2
0.62	17.5	0.60	19.8	0.59	18.5
0.76	13.2	0.73	14.6	0.72	13.7
0.88	10.1	0.85	10.9	0.84	9.8

Table 7: The negative correlation between similarity (**Sim.**) and perplexity (**Perp.**).

D Additional Experimental Results

D.1 Improvement on Core-set Algorithm

The cold start problem in core-set algorithm refers to the challenges that arise from the random initialization of seed data. This random selection can introduce significant variability in the performance of the algorithm, as the quality of the resulting core-set heavily depends on the seed data points. In some cases, random initialization may lead to poor representations of the underlying data distribution. To improve the robustness and stability of the traditional core-set algorithm, we adjust the initialization process by selecting the sample with the lowest average similarity to others, thus enhancing the diversity of the core-set distribution. To verify the effect of the warm start, we conduct comparative experiments on the three datasets based on whether the core-set method randomly selects seed data. The experimental results are displayed in Table 8, which illustrate that our improved core-set algorithm with a warm start can stably achieve performance close to or even exceeding the upper bound of the traditional core-set algorithm.

Method	**CSD-MOFs**	**NC 2024 General**	**USPTO**
Cold Start	$88.3_{0.8}$	$79.1_{0.7}$	$83.7_{0.8}$
Warm Start	**88.7**	**80.0**	**84.4**

Table 8: Comparisons of core-set algorithm with a cold start versus a warm start. We repeat five separate runs for cold start and report the mean and standard deviation.

D.2 Extraction Performance with More LLMs

We also perform entity extraction experiments on the three datasets using DeepSeek-V3 (DeepSeek-AI, 2024) as the LLM annotator. As shown in Table 9, ALLabel's extraction performance still outperforms all baselines at each pool size, further validating the effectiveness of our proposed framework across different LLMs.

D.3 Performance on More NLP Tasks

Our work is based on few-shot in-context learning and is therefore broadly applicable to a wide range of downstream NLP tasks. In this work, we primarily focus on the application of ALLabel to entity recognition tasks. To further validate the generalizability of our method, we conduct supplementary experiments on four benchmark datasets across two tasks: (1) Paraphrase Identification: MRPC (Dolan and Brockett, 2005), and PAWS (Zhang et al., 2019); (2) Natural Language Inference: MNLI-m/mm (Williams et al., 2018). We continue to use a 3-shot ICL setting with GPT-4o and vary the pool size from 30 to 100 in steps of 5. As shown in Table 10, we report the average F1 or accuracy across the four datasets. The results demonstrate that ALLabel consistently and

Method	10	15	20	25	30	35	40	45	50	55	60
					CSD-MOFs						
Random	$81.8_{1.8}$	$83.1_{1.6}$	$83.5_{1.7}$	$84.2_{1.1}$	$84.7_{1.5}$	$85.4_{1.0}$	$85.6_{2.0}$	$85.8_{1.2}$	$86.2_{1.9}$	$86.7_{1.7}$	$87.3_{1.8}$
Core-set	$84.5_{0.6}$	$85.1_{0.7}$	$85.7_{0.8}$	$85.5_{1.1}$	$86.2_{0.9}$	$86.4_{1.0}$	$86.7_{0.9}$	$87.3_{0.6}$	$87.9_{0.5}$	$88.6_{0.6}$	$88.4_{0.6}$
Perplexity	83.2	83.4	83.5	84.0	84.5	85.8	86.4	86.9	87.7	88.4	90.0
ALLabel	**87.9**	**89.4**	**90.8**	**90.2**	**91.0**	**91.5**	**91.8**	**91.6**	**91.8**	**92.4**	**92.6**
					NC 2024 General						
Random	$74.2_{2.0}$	$75.1_{1.6}$	$76.3_{1.8}$	$75.8_{1.0}$	$75.9_{0.9}$	$77.1_{2.2}$	$78.2_{1.5}$	$78.0_{0.9}$	$78.9_{1.8}$	$79.1_{1.3}$	$79.9_{1.7}$
Core-set	$76.5_{0.8}$	$77.8_{0.9}$	$78.1_{0.6}$	$77.5_{1.0}$	$78.3_{0.5}$	$79.1_{0.6}$	$78.7_{0.6}$	$79.5_{0.9}$	$79.1_{1.0}$	$80.0_{0.7}$	$79.8_{0.6}$
Perplexity	74.9	76.1	75.8	76.4	76.7	77.1	78.0	78.7	80.2	81.0	81.5
ALLabel	**77.7**	**78.5**	**79.2**	**80.6**	**81.7**	**82.2**	**82.5**	**82.9**	**82.8**	**83.1**	**83.4**
					USPTO						
Random	$79.5_{1.8}$	$81.0_{1.3}$	$81.6_{1.9}$	$82.4_{0.9}$	$81.8_{1.1}$	$83.0_{1.6}$	$83.2_{1.2}$	$83.3_{1.5}$	$83.9_{1.9}$	$84.1_{1.8}$	$84.5_{1.5}$
Core-set	$81.2_{1.0}$	$82.0_{0.5}$	$82.1_{0.9}$	$82.5_{0.8}$	$82.2_{0.6}$	$82.4_{0.7}$	$82.8_{1.0}$	$83.0_{0.5}$	$83.8_{0.7}$	$83.5_{0.8}$	$84.5_{0.7}$
Perplexity	79.9	81.0	81.2	81.7	82.5	83.6	83.0	83.3	84.2	84.8	85.1
ALLabel	**84.9**	**85.5**	**85.8**	**86.5**	**86.2**	**87.5**	**88.2**	**88.6**	**88.7**	**88.5**	**88.9**

Table 9: Extraction performance of ALLabel and other baselines across three datasets using DeepSeek-V3 as the LLM annotator, where the demonstration pool size spans from 10 to 60. We report the mean and standard deviation of five separate runs for Random and Core-set.

significantly outperforms all baselines, proving its potential for application in more NLP tasks.

D.4 Analysis of Sampling Order

ALLabel is a three-stage framework, corresponding to three AL sampling strategies. To determine the optimal sequence of these stages, we conduct ablation experiments on sampling order. Notably, the uncertainty sampling stage in our framework relies on the sampling results obtained through similarity sampling, necessitating that the former follows the latter. The experimental results are shown in Table 11, which demonstrate that the sampling order employed by ALLabel (i.e., D-S-U) outperforms the other two sampling orders across most pool sizes. We analyze this result and conclude that the diversity sampling stage should select the most representative samples to effectively cover the sample space of the entire dataset. Therefore, this sampling stage should be executed as early as possible to avoid the repeated selection of a set of data points that are too similar.

D.5 Analysis of Sampling Proportion

We also conduct experiments to determine the optimal sampling proportion of the three stages. As shown in Table 12, we compare different sampling proportions, adopting the D-S-U sequence consistently. It should be noted that since the popular setting of ICL is to adaptively select the most similar examples for each test query, we set the sampling proportion of the similarity sampling stage to be the highest among the three stages. The results of the ablation study shown in Table 4 also explain the rationality behind this treatment. Experimental results show that the 1:3:1 division proportion achieves the best performance across most pool sizes, which is adopted by ALLabel.

Method	**MRPC acc**	**MRPC F1**	**PAWS acc**	**PAWS F1**	**MNLI-m acc**	**MNLI-mm acc**
Random	$51.0_{2.2}$	$61.5_{1.7}$	$43.7_{2.3}$	$40.5_{1.4}$	$29.9_{1.7}$	$30.6_{0.9}$
Core-set	$51.8_{1.0}$	$62.3_{0.8}$	$44.3_{1.3}$	$46.2_{0.4}$	$31.0_{1.4}$	$31.3_{1.5}$
Perplexity	53.2	64.5	45.1	46.6	33.0	33.7
ALLabel	**67.5**	**72.4**	**61.7**	**69.0**	**65.3**	**67.8**

Table 10: Evaluation of ALLabel and baselines on other NLP tasks, using GPT-4o as the LLM annotator. We can observe that ALLabel still outperforms baseline methods on the four datasets.

Sequence	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**	**55**	**60**
S-U-D	**87.8**	89.3	89.4	90.2	90.5	91.2	91.2	90.9	91.1	91.4	91.9
S-D-U	87.6	89.0	90.7	90.9	91.2	92.0	91.9	**91.9**	91.7	92.1	92.4
D-S-U (**ALLabel**)	**87.8**	**89.4**	**91.0**	**91.2**	**91.5**	**92.5**	**92.2**	91.8	**91.9**	**92.8**	**93.3**

Table 11: Ablation study of sampling order on CSD-MOFs dataset. Assuming the annotation budget is M, we test three sampling sequences in the experiment: (1) S-U-D: selecting $3M/5$ through similarity sampling, followed by $M/5$ through uncertainty sampling, and finally $M/5$ through diversity sampling; (2) S-D-U: selecting $3M/5$ through similarity sampling, followed by $M/5$ through diversity sampling, and finally $M/5$ through uncertainty sampling; (3) D-S-U: selecting $M/5$ through diversity sampling, followed by $3M/5$ through similarity sampling, and finally $M/5$ through uncertainty sampling, which is also the sampling order adopted by ALLabel.

Proportion	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**	**55**	**60**
1:1:1	86.9	88.8	89.2	89.7	89.5	89.2	89.4	89.9	90.0	90.4	90.7
1:2:1	87.2	88.5	90.0	90.4	90.3	91.0	91.6	91.6	**92.1**	92.0	92.5
1:3:1 (**ALLabel**)	**87.8**	**89.4**	**91.0**	**91.2**	**91.5**	**92.5**	**92.2**	**91.8**	91.9	**92.8**	**93.3**
1:4:1	86.2	87.9	89.2	89.5	90.1	91.0	91.0	91.3	91.7	92.4	92.8
1:5:1	86.8	88.3	88.6	89.1	89.5	89.8	90.1	90.7	90.5	91.1	91.5

Table 12: Ablation study of sampling proportion on CSD-MOFs dataset. The 1:1:1 proportion refers to selecting $M/3$ samples through diversity sampling, followed by $M/3$ samples through similarity sampling, and concluding with $M/3$ samples through uncertainty sampling, and so on for other division ratios. During the three-stage division process, the number of samples is consistently rounded to the nearest integer if the division proportion is not divisible by a given pool size.