Exploring LLM Annotation for Adaptation of Clinical Information Extraction Models under Data-sharing Restrictions

Seiji Shimizu[1], **Shohei Hisada**[1], **Yutaka Uno**[2],
Shuntaro Yada[1,3], **Shoko Wakamiya**[1], **Eiji Aramaki**[1]
[1]Nara Institute of Science and Technology (NAIST)
[2]Biometrics Research Laboratories, NEC
[3]Institute of Library, Information and Media Science, University of Tsukuba
shimizu.seiji.so8@is.naist.jp

Abstract

In-hospital text data contains valuable clinical information, yet deploying fine-tuned small language models (SLMs) for information extraction remains challenging due to differences in formatting and vocabulary across institutions. Since access to the original in-hospital data (source domain) is often restricted, annotated data from the target hospital (target domain) is crucial for domain adaptation. However, clinical annotation is notoriously expensive and time-consuming, as it demands clinical and linguistic expertise. To address this issue, we leverage large language models (LLMs) to annotate the target-domain data for the adaptation. We conduct experiments on four clinical information extraction tasks, including eight target-domain datasets. Experimental results show that LLM-annotated data consistently enhances SLM performance and, with a larger number of annotated data, outperforms manual annotation in three out of four tasks[1].

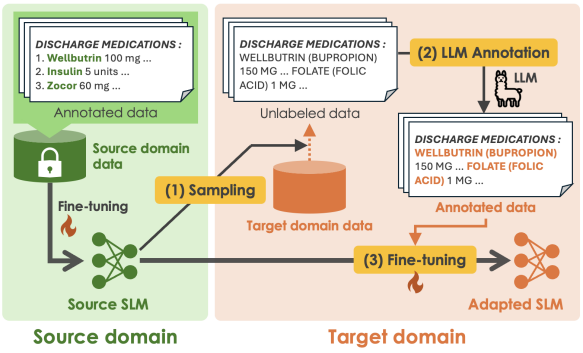

Figure 1: Overview of our SFDA approach. The goal of SFDA is to adapt the SLM fine-tuned on the source-domain data to unlabeled target-domain data. For the adaptation, only the SLM is available from the source domain due to data-sharing restrictions. The source SLM struggles with target-specific formats, such as "<brand name> (<generic name>)" (e.g., "WELLBUTRIN (BUPROPION)"). Our approach adapts the source SLMs by (1) sampling target data based on SLM's high uncertainty, (2) annotating them with an LLM as an alternative to manual annotation, and (3) fine-tuning the source SLM with the newly annotated data.

1 Introduction

In-hospital text data often contains valuable clinical information not captured by structured fields in electronic health records (Zweigenbaum et al., 2007; Escudié et al., 2017; Wang et al., 2018). Fine-tuned small language models (SLMs) offer computationally efficient inference for extracting such information and have been shown to outperform prompt-based large language models (LLMs) (Naguib et al., 2024). However, SLMs fine-tuned on in-hospital data (i.e., the **source domain**) often experience performance degradation when applied to data from a different hospital (i.e., the **target domain**) due to domain-specific vocabulary and formatting (Wu et al., 2014; Bethard et al., 2017; Miller et al., 2017). Additionally, patient privacy regulations often restrict access to source-domain data, posing further challenges for domain adaptation (Laparra et al., 2020). These challenges are addressed by source-free domain adaptation (SFDA), where adaptation must be performed using only a fine-tuned source model without direct access to the source data (Laparra et al., 2021a).

Su et al. (2022) previously compared various formulations of two major SFDA approaches in clinical NLP. **Self-training** (Kumar et al., 2010; Li and Zhang, 2019), which leverages the source SLM's own predictions as supervisions, failed to consistently improve model performance, whereas **active learning** (Settles, 2009), which relies on minimal manual annotation, proved to be a reliable alternative. Nonetheless, clinical annotation demands sufficient expertise and time from annotators (Luo et al., 2020; Su et al., 2021), which can pose a bar-

[1]Our code is available at: https://github.com/seiji-shimizu/LLM-Annotation-Clinical-SFDA

Findings of the Association for Computational Linguistics: ACL 2025, pages 14678–14694

rier to its application in real-world scenarios. This highlights the critical need for robust, human-free annotation methods in SFDA, particularly within the clinical domain.

To address this gap, we explore **active learning with LLM annotation** in an SFDA setting, inspired by recent studies (Liang et al., 2024; Xiao et al., 2023; Zhang et al., 2023; Liu et al., 2024). Fig. 1 shows an overview of our approach. The objective of SFDA is to adapt fine-tuned source-domain SLMs to unlabeled target-domain data. We formulate SFDA as a three-step process: **(1)** sampling target-domain data with target-specific formats and vocabularies based on the SLM's uncertainty, **(2)** annotating the selected samples using an LLM as an alternative to manual annotation, and **(3)** fine-tuning the SLM with the newly annotated data.

To evaluate the effectiveness of our approach, we conduct experiments on four clinical information extraction tasks encompassing eight source-target dataset pairs (summarized in Tables 1 and 2). In a preliminary experiment, we apply LLM zero-shot annotation in step **(2)** and observe a performance decline in clinical named entity recognition (NER). Upon analyzing the quality of the resulting annotations, we attribute this decline to the performance gap between the LLM and the fine-tuned source SLM, aligning with recent findings (Hu et al., 2024; Naguib et al., 2024).

Motivated by this, we introduce a novel LLM annotation method termed SLM-Assisted LLM Annotation (**SALA**). In this method, we guide an LLM to correct the SLM's prediction on the sampled data in step **(1)** instead of generating annotation from scratch. By doing so, we aim to maintain annotation quality that matches or exceeds the source SLM performance across various tasks. We evaluate our approach against unadapted SLMs, as well as the best-performing formulations of self-training and active learning derived from Su et al. (2022). Our results demonstrate that active learning with LLM annotation consistently enhances SLM performance across all tasks and, with a larger number of annotated samples, outperforms active learning with human annotation in three out of four tasks. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to explore LLM annotation in the SFDA setting, evaluating its effectiveness for adapting the source SLMs fine-tuned on clinical data.
- We propose a novel LLM annotation method that leverages SLM's prediction to maintain improvements across various clinical information extraction tasks.
- Through experiments on four tasks and eight target datasets, we demonstrate that active learning with LLM annotation consistently improves SLM performance and, with a larger number of annotations, even outperforms active learning with human annotation in three out of four tasks.

2 Related Work

Source-free Domain Adaptation: Unlike unsupervised domain adaptation, source-free domain adaptation (SFDA) adapts a fine-tuned model to target-domain data without access to the source-domain data (Laparra et al., 2020). While SFDA (Liang et al., 2020) has recently gained traction in computer vision (see survey of Yu et al., 2023), it remains underexplored in NLP, with only a few existing studies (Zhang et al., 2021; Yin et al., 2022; Shimizu et al., 2024; Zhao et al., 2024). In the clinical domain, Su et al. (2022) compared various formulations of self-training and active learning using in-hospital data, finding that self-training struggled to consistently improve model performance. This suggests that manual annotation remains the de facto standard for clinical SFDA. To achieve reliable human-free adaptation, we explore SFDA with LLM annotation, conducting an extensive evaluation across multiple clinical tasks.

LLM as Active Annotator: Recent advancements have shown that large language models (LLMs) are effective annotators for active learning (Liang et al., 2024; Xiao et al., 2023; Zhang et al., 2023; Liu et al., 2024). These methods typically generate annotations with an LLM and improve the performance of yet-to-be-fine-tuned SLMs. On the other hand, some research has shown that prompt-based LLMs often underperform compared to fine-tuned SLMs on specialized tasks such as clinical named entity recognition (NER) (Hu et al., 2024; Naguib et al., 2024). It remains unknown whether LLM annotation can enhance SLMs in clinical SFDA, particularly considering the potential negative impact on already fine-tuned clinical SLMs. We answer this by evaluating LLM annotation in the SFDA setting and proposing a novel LLM annotation method that integrates the source SLM's prediction.

Task	Description	Example
Named Entity Recognition (**NER**)	Given a text, predict spans for clinical entities and their types.	**Input:** *The patient seemed subdued.* **Answer**: {*subdued*: Problem}
Relation Extraction (**RE**)	Given a sentence with two clinical entities marked, classify their relation.	**Input:** *<e>Penicillin</e> causes <e>rash</e>.* **Answer**: Treatment improves
Negation Detection (**ND**)	Given a text with a clinical entity marked, classify if it is negated or not.	**Input:** *She did not complain of <e> any fever </e>.* **Answer**: Negated
Time Expression Recognition (**TER**)	Given a text, predict spans for time expressions and their types.	**Input:** *The patient underwent surgery on July.* **Answer:** {*July*: Month-Of-Year}

Table 1: Overview of the four clinical information extraction tasks.

Task	Data Source	Source Dataset	Size	Target Dataset: *Denotation*	Size
NER	i2b2 2010	Beth Clinical Notes	74 documents	Partners Clinical Notes: *Part*	97 documents
		Partners Clinical Notes	97 documents	Beth Clinical Notes: *Beth*	74 documents
RE		Beth Clinical Notes	2,037 sentences	Partners Clinical Notes: *Part*	1,264 sentences
		Partners Clinical Notes	1,264 sentences	Beth Clinical Notes: *Beth*	2,037 sentences
ND	SemEval 2021 Task 10	Mayo Clinical Notes	10,259 instances	i2b2 2010: *i2b2*	5,545 instances
				MIMIC-III: *mimic*	9,580 sentences
TER		Mayo Clinical Notes	278 documents	Food Security Reports: *Food*	17 documents
				News Reports: *News*	99 documents

Table 2: Source and target-domain datasets used in this study. The source datasets are used for fine-tuning SLMs, while the target datasets are used for adaptation and evaluation.

3 Data

We base our experiments on four clinical information extraction tasks summarized in Table 1, and each task is associated with pairs of source and target datasets summarized in Table 2. **Named Entity Recognition (NER)** and **Relation Extraction (RE)** tasks are derived from the i2b2 2010 (Uzuner et al., 2011), which provides clinical notes from two hospitals: Beth Israel Deaconess Medical Center and Partners Healthcare. We fine-tune SLMs on data from one hospital and treat the other hospital's data as the target-domain data, yielding four target datasets. **Negation Detection (ND)** and **Time Expression Recognition (TER)** tasks are based on SemEval 2021 Task 10 (Laparra et al., 2021b), which provides one source-domain SLM and two target datasets for each task. Using the provided source models, we adapt them to the respective target datasets, adding four more target datasets. Although the two target datasets for TER are non-clinical (Laparra et al., 2018), we demonstrate the performance degradation of the source SLMs on these datasets in Sect. 4. Thus, we include these target datasets to evaluate SFDA methods intended for clinical information extraction. In total, we adapt the source SLMs to eight target datasets.

Task	Source	Target	Δ	Source	Target	Δ
NER	*Part* 85.3	*Beth* 81.3	**-4.0**	*Beth* 89.4	*Part* 80.9	**-8.5**
RE	*Part* 70.2	*Beth* 56.1	**-14.1**	*Beth* 66.6	*Part* 57.3	**-9.3**
ND	*Mayo* 82.0	*i2b2* 84.6	+2.6	*Mayo* 82.0	*mimic* 63.5	**-18.5**
TER	*Mayo* 96.8	*News* 78.1	**-18.7**	*Mayo* 96.8	*Food* 78.4	**-18.4**

Table 3: The source SLM performance on source and target-domain data in $F_1(\%)$ scores and their difference. Results are averaged over three runs with different seeds. Scores on the source domain for ND and TER are cited from Laparra et al. (2021a).

4 Motivation: Performance Degradation

Our motivation is to address the performance degradation of source-domain fine-tuned SLMs when applied to target-domain data. In this section, we showcase the performance degradation on the target-domain datasets summarized in Table 2. For each source dataset, RoBERTa[2] (Liu, 2019) is fine-tuned on the training set and evaluated on the evaluation set. Additionally, we evaluate the fine-tuned source SLMs on the evaluation sets of the corresponding target datasets.

[2] https://huggingface.co/FacebookAI/roberta-base

Table 3 shows the results in $F_1(\%)$ scores. To summarize, there are noticeable performance declines from the source to the target datasets in nearly all source-target pairs. The degradation ranges from 4.0% to 18.7%, with 11.1% on average. This confirms previous findings on the performance degradation of clinical information extraction models (Wu et al., 2014; Bethard et al., 2017; Miller et al., 2017) and underscores the importance of domain adaptation. To further illustrate the performance degradation of the source SLMs, we provide specific examples of prediction errors arising from differences in formats and vocabularies.

NER: In NER, target-specific formats often lead to errors. Consider the following example from *Part*:

> **Input:** *LANTUS (INSULIN GLARGINE) 35 UNITS...*

Here, "*LANTUS (INSULIN GLARGINE)*" is a single medication, with the format "*<brand name> (<generic name>)*." However, since such formatting is barely present in the source domain (*Beth*), the source SLM incorrectly identifies "*LANTUS*" and "*INSULIN GLARGINE*" as separate medications.

RE: Errors in RE are often caused by variations in the vocabulary used to describe relationships between concepts. Below is an example from *Beth*:

> **Input:** *<e>diabetic ulcer</e> s/p <e>surgery</e> but never healed.*

In this example, "*s/p*" stands for "*status post*", indicating that the "*diabetic ulcer*" did not improve after "*surgery*". However, since this abbreviation is not used in the source domain (*Part*), the SLM fails to correctly interpret the relationship between the concepts and predicts "*Treatment Improves*".

ND: In ND, formatting differences between the source and target-domains often lead to errors. Below is an example from *mimic*:

> **Input:** *Tobacco: no, <e>Alcohol</e>: no ...*

The input lists the patient's social history, indicating the absence of alcohol use. The source SLM fails in detecting the negation for "*Alcohol*" since the source-domain data rarely includes this type of formatting.

TER: The target datasets for TER are derived from non-clinical data, making vocabulary differences a frequent source of errors.

> **Input:** *AP-NY-12-05-98 0942EST ...*

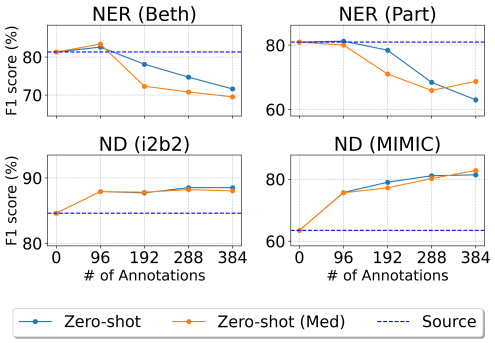

Figure 2: Performance of the source SLMs adapted with zero-shot annotations in $F_1(\%)$ scores. Results are averaged over three runs with different seeds. "Zero-shot" refers to the performance of the general-domain LLM, while "Zero-shot (Med)" refers to the performance of the medical-domain LLM. The dotted lines represent the performance of the unadapted source SLMs. While the performance improves in ND with the increasing annotations, it declines in NER.

In this example, "*AP*" and "*NY*" represent "*Associate Press*" and "*New York*", respectively. However, since such vocabularies are not present in the Mayo clinical notes, the source SLM fails to properly recognize the temporal expression "*12-05-98*" as a standard Month-Date-Year format. The examples above highlight the challenges of applying the source SLMs to target domains.

5 Preliminary: Zero-shot Annotation

To mitigate performance degradation, we leverage LLMs to annotate target-domain data within our approach (Fig. 1). As a preliminary experiment, we evaluate the effectiveness of zero-shot annotation with the following specifications.

Active Learning with Zero-shot Annotation: In step **(1)**, we select target samples with the highest predictive entropy, following Su et al. (2022). In step **(2)**, we prompt LLMs with one of the selected samples and a detailed task description at a time to generate an annotation. In step **(3)**, we fine-tune the source SLM with the generated annotations, mapping them to the SLM's label space with a rule-based resolver. Throughout this study, we use downloaded open-source LLMs to safeguard clinical data and prevent third-party sharing in real-world applications. Specifically, we employ Llama-3.3-70B[3] (Dubey et al., 2024), representing a general-domain LLM, and Med42-70B[4] (Christophe et al., 2024), represent-

[3] https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct

[4] https://huggingface.co/m42-health/med42-70b

Method	#	NER		ND	
		Beth	*Part*	*i2b2*	*mimic*
Source	96	51.5	39.2	85.1	80.9
Zero-Shot	96	52.4	**67.5**	**89.2**	82.3
Zero-Shot$_{\text{Med}}$	96	**54.6**	46.1	**89.2**	**89.7**
Source	384	**64.1**	**66.8**	88.8	65.2
Zero-Shot	384	49.3	58.5	94.3	90.5
Zero-Shot$_{\text{Med}}$	384	49.9	44.2	**94.5**	**92.8**

Table 4: Performance comparison (F_1% scores) between the source SLM and zero-shot LLM annotation on the annotation targets. The scores are averaged over three runs. The "#" column indicates the number of annotations, and "Med" denotes the medical LLM. Full results for all target datasets are available in Appendix A.2.

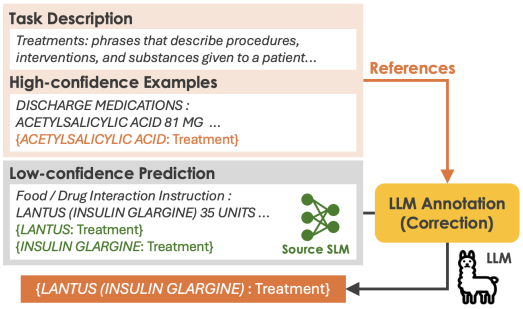

Figure 3: Overview of SALA. Instead of generating LLM annotation from scratch, we prompt LLM to correct SLM's low-confidence prediction, referencing a detailed task description and high-confidence examples.

ing a medical-domain LLM. Implementation details and example prompts are provided in the Appendix A.1.

Fig. 2 illustrates the performance of the source SLMs adapted with zero-shot annotation for **NER** and **ND**. For the evaluation, we experimented with the number of selected samples from 96 to 384 instances. As the number of annotations increases, performance in **NER** decreases, whereas it improves in **ND**. This result suggests that the scalability of LLM annotation can not be fully exploited with zero-shot annotation in tasks like **NER**.

We hypothesize that the performance decline stems from a gap between the source SLM's performance and zero-shot annotation. Table 4 shows the source SLM's performance and the quality of zero-shot annotation on the sampled annotation targets. Since the targets are chosen from samples with lower uncertainty, the source SLM's performance generally improves as the number of annotations increases. However, the quality of zero-shot annotation fails to meet the source SLM's performance in **NER**. The performance gap between fine-tuned SLMs and LLMs aligns with previous findings (Hu et al., 2024; Naguib et al., 2024) and negatively affects the source SLM, resulting in a decline in performance after fine-tuning. Notably, the medical LLM underperforms the general domain LLM on *Part*. While the exact reasons for the lower performance remain an open question, our results align with prior findings (Dorfner et al., 2024), which suggest that biomedical LLMs do not always outperform generalist models on unseen clinical data. The above results emphasizes the importance of ensuring that annotation quality matches or exceeds the source SLM performance for the scalability of LLM annotation.

6 Proposed: SALA

To improve the LLM annotation in target domains, we introduce SALA, SLM-assisted LLM annotation. Inspired by recent studies (Yang et al., 2024; Xu et al., 2023), we incorporate the source SLM prediction into the LLM's in-context learning. As depicted in Fig. 3, we guide the LLM to generate annotations based on three inputs:

- **Task Description:** A detailed explanation based on the annotation guidelines, providing the LLM with a clear reference for annotation.
- **High-confidence Examples:** Examples retrieved from a demonstration pool of high-confidence SLM predictions and LLM annotations.
- **Low-confidence Prediction:** The source SLM prediction on the low-confidence annotation target.

For the annotation, we prompt LLMs to **correct errors in the low-confidence prediction** while referencing the task description and high-confidence examples. This correction-based approach helps maintain annotation quality that matches or exceeds the source SLM's performance across various tasks by constraining LLM annotation to improve upon the SLM prediction. In this section, we elaborate on how high-confidence examples are retrieved (Sect. 6.1) and how low-confidence predictions are corrected (Sect. 6.2). Finally, we present the overall algorithm of SALA to iteratively improve the annotation quality (Sect. 6.3).

Notations: We denote the source SLM as $\mathcal{S}$, the LLM as $\mathcal{P}$, a target-domain sample as $x_i \in D$, and an annotation target as $\bar{x}_i \in \bar{D}$.

6.1 Retrieving High-confidence Examples

To retrieve high-confidence examples, we initialize a class-wise demonstration pool with high-confidence SLM predictions. To do so, we first construct a class-wise target data D^c for each class $c \in C$ by pseudo-labeling the target-domain data D with the source SLM. Then, a class-wise demonstration pool D^c_{demo} is initialized via:

$$D^c_{\text{demo}} = \{(x_i, c)) \mid rank(h_i) \leq R\%\} \quad (1)$$

where h_i is the SLM's predictive entropy for a sample x_i. Following Su et al. (2022), we calculated the entropy based on the model's output softmax probability distribution. $rank(h_i) \leq R\%$ selects the bottom $R\%$ of the class-wise target-domain data D^c with the lowest entropy. From each of this class-wise demonstration pool, we retrieve a single high-confidence example $e^c \in D^c_{\text{demo}}$ based on the highest embedding similarity with the annotation target $\bar{x}_i$. The full set of high-confidence examples for $\bar{x}_i$ is denoted as $E = \{e^c \mid c \in C\}$.

6.2 Correcting Low-confidence Prediction

To maintain annotation quality across various tasks, we adopt a correction-based approach rather than generating annotations from scratch. Specifically, we query the LLM with a prompt t to correct the SLM prediction $\mathcal{S}(\bar{x}_i)$, using a task description d and high-confidence examples E as references:

$$t = T(d, E, \bar{x}_i, \mathcal{S}(\bar{x}_i)) \quad (2)$$

where T is a manually-designed template that guides the baseline correction process. The LLM-annotated data $\tilde{D}_{\text{llm}}$ are then generated as follows:

$$\tilde{D}_{\text{llm}} = \{(\bar{x}_i, \tilde{y}^{\text{llm}}_i) \mid \bar{x}_i \in \bar{D}, \tilde{y}^{\text{llm}} = \mathcal{R}(\mathcal{P}(t))\}, \quad (3)$$

where $\mathcal{P}(t)$ represents the LLM's response to the prompt t, and $\mathcal{R}$ is a rule-based resolver that maps the LLM-generated output to the source SLM's label space. Specifically, LLM-corrected predictions are converted into PyTorch[5] tensors with $\mathcal{R}$ and then used as labels for fine-tuning the source SLMs.

6.3 Overall Algorithm

The overall pipeline of SALA is detailed in Alg. 1. The algorithm builds upon the work of Xiao et al. (2023), with the key distinction being the incorporation of task-specific knowledge from the source SLM. As described in Sect. 2, we first select the top K samples with the highest entropy as the annotation targets $\bar{D}$ (line 2). These annotation targets are then annotated to obtain LLM-annotated data $\tilde{D}_{\text{llm}}$ (line 8). We iteratively apply LLM annotation, **fine-tuning**, and **annotation filtering** to progressively enhance the quality of the final LLM-annotated data $\tilde{D}_{\text{llm}}$.

Algorithm 1:

Input : D: The unlabeled target dataset
$\mathcal{P}$: The LLM
$\mathcal{S}$: The source SLM
$\mathcal{R}$: The resolver
K: The number of annotations
N_{iter}: The maximum iteration number
τ: The filtering threshold

Output : $\tilde{D}_{\text{llm}}$: The LLM-annotated data

1 # Select uncertain samples
2 $\bar{D} = [x_i$ for $x_i \in$ top K entropy in $D]$
3 # Initialize the class-balanced demonstration pool
4 $D_{\text{demo}} = \cup_{c \in C} D^c_{\text{demo}}$
5 **for** $iter \leftarrow 0$ **to** N_{iter} **do**
6 # LLM annotation
7 Construct a prompt t as Eq.(2)
8 $\tilde{D}_{\text{llm}} = \{(\bar{x}_i, \tilde{y}^{\text{llm}}_i) \mid \bar{x}_i \in \bar{D}, \tilde{y}^{\text{llm}} = \mathcal{R}(\mathcal{P}(t))\}$
9 # Fine-tune SLM
10 Fine tune $\mathcal{S}$ with $\tilde{D}_{\text{llm}}$
11 # Annotation filtering with cross entropy loss l_i
12 $\tilde{D}_{\text{llm}} = \{(\bar{x}_i, \tilde{y}^{\text{llm}}_i) \mid l_i < \tau\}$
13 $D_{\text{demo}} = \tilde{D}_{\text{llm}}$
14 # Update $\bar{D}$
15 $\bar{D} = \bar{D} \setminus \{\bar{x}_i \mid (\bar{x}_i, \tilde{y}^{\text{llm}}_i) \in \tilde{D}_{\text{llm}}\}$
16 **end**

Fine-tuning: Once $\tilde{D}_{\text{llm}}$ is generated, the source SLM $\mathcal{S}$ is fine-tuned on $\tilde{D}_{\text{llm}}$ at each iteration (line 10). The fine-tuning serves two purposes: (1) to improve the low-confidence predictions by updating $\mathcal{S}$, and (2) to distill recurring patterns from $\tilde{D}_{\text{llm}}$. While (1) directly improves the annotations, (2) is used to filter outliers in $\tilde{D}_{\text{llm}}$.

Annotation Filtering: We filter $\tilde{D}_{\text{llm}}$ based on a cross-entropy loss l_i between output logits of $\mathcal{S}$ and the LLM annotation $\tilde{y}^{\text{llm}}_i$ for each $(\bar{x}_i, \tilde{y}^{\text{llm}}_i) \in \tilde{D}_{\text{llm}}$. LLM-annotated data with a loss l_i below a threshold τ are considered clean annotations and used in the demonstration pool for the next iteration (lines 12 and 13), while the rest are re-annotated (line 15). By using the updated SLM, annotation targets that are inconsistent with other LLM-annotated data and incompatible with the SLM's knowledge are filtered out.

[5] https://pytorch.org/

Method	Human-free	NER		RE		ND		TER		Avg
		Beth	*Part*	*Beth*	*Part*	*i2b2*	*mimic*	*Food*	*News*	
Full-supervision (skyline)	×	88.7	86.7	75.4	65.7	89.8	88.7	85.8	83.0	83.0
Source	✓	81.3	80.9	56.1	57.3	84.6	63.5	78.1	78.4	72.5
Self-training	✓	81.9⋆	72.7	39.5	49.0	86.1⋆	70.7⋆	78.7⋆	78.0	69.6
Active learning	×	84.1⋆	82.9⋆	66.2⋆	57.1	86.5⋆	76.9⋆	**84.0**⋆	**82.4**⋆	77.5⋆
Active + SALA_{Med}	✓	83.2⋆	81.3⋆	72.2⋆	57.6⋆	**88.3**⋆	**79.6**⋆	79.6⋆	78.7⋆	77.6⋆
Active + SALA	✓	**84.2**⋆	**83.2**⋆	**72.3**⋆	**66.2**⋆	88.0 ⋆	78.6⋆	79.9⋆	75.1	**78.4**⋆

Table 5: $F_1(\%)$ scores for the SLM performance on the evaluation sets of target datasets. Results are averaged over three runs with different seeds. The last column shows the average score across all datasets. "Source" refers to the performance of unadapted source SLMs, while "Full-supervision (skyline)" represents the performance of fully fine-tuned models using all labeled target data. "Med" indicates the medical-domain LLM. The "Human-free" column indicates whether human annotations are used. ⋆ indicates improvement from the source SLMs. "Active + SALA_{Med}" outperforms the source SLM in all tasks, while "Active + SALA" outperforms active learning with human annotation in three out of four tasks.

7 Experiment

We conduct an experiment comparing our approach against existing SFDA baselines. With SALA used for the LLM annotation, the proposed method is denoted as "Active + SALA". Our focus is on the performance of the SLMs, as computationally efficient inference is crucial for processing in-hospital text data. A comparison with LLM zero-shot and few-shot inferences is provided in Appendix A.3.

7.1 Setup

Baselines: We consider the most robust SFDA formulations from Su et al. (2022), both with and without human annotation, alongside the unadapted source SLM. For the human-free approach, we include self-training, which uses an iterative training and dataset construction strategy. For the approach with human annotation, we include active learning, which also follows an iterative training and dataset construction strategy. As an upper-bound reference (skyline), we include a fully supervised setting, where the source models are fine-tuned on the labeled development set of target domain data.
Datasets: Experiments are conducted on all source-target pairs listed in Table 2. For each pair, the source SLMs are adapted using the **unlabeled development set** of the target datasets and evaluated on the evaluation sets.
Implementation Details: For self-training and active learning, we adopt the implementation from Su et al. (2022), setting the annotation budget to 12 per iteration over eight iterations for active learning. The LLMs used are the same as those in Sect. 5. As for the hyper-parameters of SALA, the maximum iteration number N_{iter} and filtering threshold τ in Alg. 1 are set to 5 and 5e-3, respectively. The number of annotations K is set to 384, which accounts for nearly all instances in the development sets of the TER target domain. Additional implementation details and example prompts are provided in Appendix A.1.

7.2 Results

Table 5 show results in $F_1(\%)$ scores across different target datasets. Self-training fails to consistently improve the performance of the source SLM, confirming previous findings (Su et al., 2022). In contrast, active learning provides a strong baseline, improving the source SLMs across nearly all target datasets with a limited annotation budget. Our approach consistently improves the source SLMs, with "Active + SALA_{med}" outperforming the unadapted source SLMs across all target datasets. Notably, "Active + SALA" outperforms even active learning with human annotation in three out of four tasks. In summary, LLM annotation offers a reliable alternative to human annotation in SFDA settings by consistently enhancing the performance of source SLMs.

8 Discussion

In this work, we introduced active learning with LLM annotation in an SFDA setting, and demonstrated that the proposed LLM annotation method (SALA) can enhance the performance of source SLMs across various tasks. In this section, we address remaining questions regarding the effectiveness of our approach, particularly in the utilization of SLM assistance for LLM annotation (Sect. 8.1) and the sampling method for annotation targets

Method	Iteration	NER		RE		ND		TER		Avg
		Beth	*Part*	*Beth*	*Part*	*i2b2*	*mimic*	*Food*	*News*	
Zero-Shot	-	49.3	58.5	56.2	**67.8**	94.3	90.5	55.1	37.4	63.6
LLM-active	First	51.5	57.4	52.6	62.7	93.7	**93.3**	46.7	36.8	61.8
	Last	53.0	59.0	56.3	64.8	**94.8**	89.9	49.4	45.0	64.0
SALA	First	63.1	74.3	53.0	67.0	92.4	85.6	78.4	68.0	72.7
	Last	**65.8**	**74.6**	**56.9**	66.3	93.7	86.7	**78.6**	**70.2**	**74.1**

Table 6: Annotation quality of different LLM annotation methods in $F_1(\%)$ scores. Results are averaged over three runs with different seeds. The "Iteration" column indicates the results from either the first or last iteration. SALA achieves high-quality annotation across various tasks and the highest score on average.

	NER	RE	ND	TER	Avg
Random	82.9	69.1	81.6	76.9	77.6
Entropy	**83.7**	**69.3**	**83.3**	**77.5**	**78.4**

Table 7: Performance comparison between random and entropy-based sampling used in "Active + SALA". The results are averaged over three runs with different seeds.

(Sect. 8.2). We also demonstrate that SALA effectively mitigates the performance decline observed in the zero-shot annotation (Sect. 8.3).

8.1 LLM Annotation Quality

To ensure consistent annotation quality across diverse tasks, we incorporate source SLM predictions into the LLM annotation process. To assess the effectiveness of this SLM-assisted approach, we compare SALA with LLM annotation performed without SLM assistance.

Baselines: For LLM annotation without SLM assistance, we consider zero-shot annotation (**Zero-shot**) from Sect. 5. Additionally, we include an LLM-only active annotation setting (**LLM-active**), adjusting the method proposed by Xiao et al. (2023) to our experimental settings. Specifically, we employ Algorithm 1 with two modifications. First, following Xiao et al. (2023), the demonstration pool (line 4) is initialized with LLM-generated examples and annotations based on randomly selected unlabeled target samples. Second, the prompting process uses the same template as SALA (Eq. 2), but source SLM predictions are omitted. Further implementation details for this baseline are provided in the Appendix A.1.

Table 6 presents the comparison of annotation quality in $F_1(\%)$ scores on the target development sets. Full results, including the medical-domain LLM annotation, are presented in the Appendix A.5. While all three LLM annotation methods perform similarly in RE and ND, both Zero-shot and LLM-active suffer from lower annotation quality in NER and TER. SALA successfully mitigates this issue, maintaining relatively high-quality annotations in both NER and TER, indicating the effectiveness of the SLM assistance. Additionally, annotation quality at the final iteration generally surpasses that of the initial iteration, indicating that the iterative approach in Alg. 1 contributes to performance improvement. In summary, SALA achieves high-quality annotations across various tasks, demonstrating the effectiveness of SLM assistance for LLM annotation in the SFDA setting.

8.2 Entropy-based vs. Random Sampling

To select annotation targets where the SLM is likely to produce prediction errors, we use the source SLM's predictive entropy as a selection criterion. To evaluate the effectiveness of this sampling strategy, we compare the performance of "Active + SALA" using both random and entropy-based sampling methods.

Table 7 presents the performance of adapted SLMs in $F_1\%$ scores for each sampling method. The results are averaged for each task. In all tasks, entropy-based sampling outperforms random sampling. This result indicates that the source SLM's predictive entropy is effective in selecting annotation targets that benefit from LLM correction, confirming the existing comparison of sampling methods (Zhang et al., 2023).

8.3 Performance with Increasing Annotations

Using zero-shot annotation, we observed a performance decline of the source SLMs in NER (see Sect. 2). Fig. 4 shows the performance of "Active + SALA" with the number of annotations increasing from 96 to 384. In contrast to Fig. 2, the SLM performance consistently improves with increasing annotations in NER task, showing the scalability

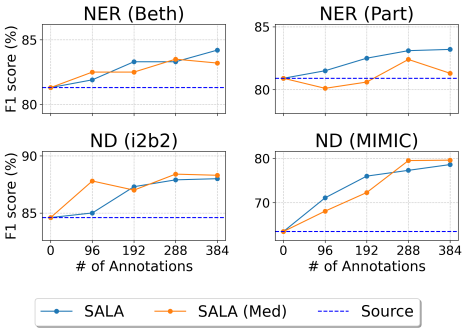

Figure 4: Performance of "Active + SALA" on NER and ND in $F_1(\%)$ scores. Results are averaged over three runs with different seeds. The dotted horizontal line is the performance of the unadapted source SLM. The performance improves with increasing annotations for both tasks.

of SALA in various clinical information extraction tasks.

9 Conclusion

In this paper, we explore active learning with LLM annotation, adapting SLMs fine-tuned on the source domain to the target domain's specific formats and vocabularies. To improve annotation quality, we introduce a novel LLM annotation method, SALA, which incorporates the source SLM prediction into the LLM annotation through in-context learning. Through experiments across four clinical information extraction tasks and eight target datasets, we demonstrate that the proposed approach consistently enhances the performance of source SLMs and outperforms active learning with human annotation in three out of four tasks with a larger annotation number. Furthermore, we show that SLM assistance improves the quality of LLM-generated annotations in challenging tasks such as clinical NER. These results highlight the potential of LLM annotation as a scalable and effective alternative to human annotation in clinical SFDA, where clinical and linguistic expertise is typically required.

10 Limitations

Due to the limited availability of publicly accessible clinical corpora, our study utilizes data from three clinical institutions. Our qualitative error analysis of the source SLMs (see Sect. 4) suggests that the primary differences among these institutions are limited to variations in formatting and vocabulary. While our experimental results demonstrate performance improvements, residual performance degradation persists when adapting across different clinical specialties and languages. Future work should explore more extensive evaluations across diverse institutions and language settings to better facilitate the deployment of clinical information extraction models in real-world scenarios.

Another limitation of this study is the use of fixed hyperparameters in LLM annotation, which may result in suboptimal annotation quality. For instance, on the *mimic* dataset in the ND task, SALA shows lower annotation quality compared to other methods, potentially due to biases introduced by SLM predictions. This issue could be mitigated by searching the optimal number of annotations. Notably, the performance of the SLM continues to improve even at an annotation number of 384 (see Fig. 4), indicating that further exploration of the number of annotations could yield better results. Another example is the filtering threshold τ. Although annotation quality improved in the last iteration of Alg. 1 compared to the first iteration (see Table 6), identifying the optimal number of iterations could further enhance annotation quality. We leave the search for such hyperparameters for future work.

Lastly, the performance gains from "Active + SALA" are limited in the TER task. TER is particularly challenging as it involves token classification across 56 classes. This limitation could potentially be addressed through more robust fine-tuning (Xiao et al., 2023; Zhang et al., 2023) or by leveraging a more capable LLMs (Bi et al., 2024) to enhance annotation quality.

11 Ethics Statement

While SALA offers a scalable and effective approach to annotation in clinical SFDA, the use of large language models (LLMs) for annotation may inherit biases present in the LLMs and the fine-tuned source SLMs. These biases could potentially affect annotations and predictions related to sensitive characteristics such as race, gender, disabilities, and other protected attributes. Moreover, the clinical context amplifies the importance of ethical considerations, as inaccurate or biased annotations could adversely impact downstream applications in healthcare. To mitigate these risks, we recommend that users apply rigorous bias evaluation and mitigation strategies, including techniques for bias reduction in LLM outputs and thorough post-hoc analysis of model predictions.

Acknowledgments

This work was supported by Cross-ministerial Strategic Innovation Promotion Program (SIP) on "Integrated Health Care System" Grant Number JPJ012425 and CREST Grant Number JPMJCR22N1, Japan.

References

Steven Bethard, Guergana Savova, Martha Palmer, and James Pustejovsky. 2017. SemEval-2017 task 12: Clinical TempEval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572, Vancouver, Canada. Association for Computational Linguistics.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Clément Christophe, Praveen K Kanithi, Tathagata Raha, Shadab Khan, and Marco AF Pimentel. 2024. Med42-v2: A suite of clinical llms. *arXiv preprint arXiv:2408.06142*.

Felix J Dorfner, Amin Dada, Felix Busch, Marcus R Makowski, Tianyu Han, Daniel Truhn, Jens Kleesiek, Madhumita Sushil, Jacqueline Lammert, Lisa C Adams, et al. 2024. Biomedical large languages models seem not to be superior to generalist models on unseen medical data, 2024. *URL https://arxiv.org/abs/2408.13833*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jean-Baptiste Escudié, Bastien Rance, Georgia Malamut, Sherine Khater, Anita Burgun, Christophe Cellier, and Anne-Sophie Jannot. 2017. A novel data-driven workflow combining literature and electronic health records to estimate comorbidities burden for a specific disease: a case study on autoimmune comorbidities in patients with celiac disease. *BMC medical informatics and decision making*, 17:1–10.

Yan Hu, Qingyu Chen, Jingcheng Du, Xueqing Peng, Vipina Kuttichi Keloth, Xu Zuo, Yujia Zhou, Zehan Li, Xiaoqian Jiang, Zhiyong Lu, Kirk Roberts, and Hua Xu. 2024. Improving large language models for clinical named entity recognition via prompt engineering. *Journal of the American Medical Informatics Association*, 31(9):1812–1820.

Hongjin Kim, Jai-Eun Kim, and Harksoo Kim. 2024. Exploring nested named entity recognition with large language models: Methods, challenges, and insights. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8653–8670, Miami, Florida, USA. Association for Computational Linguistics.

Abhishek Kumar, Avishek Saha, and Hal Daume. 2010. Co-regularization based semi-supervised domain adaptation. *Advances in neural information processing systems*, 23.

Egoitz Laparra, Steven Bethard, and Timothy A Miller. 2020. Rethinking domain adaptation for machine learning over clinical language. *JAMIA Open*, 3(2):146–150.

Egoitz Laparra, Xin Su, Yiyun Zhao, Özlem Uzuner, Timothy Miller, and Steven Bethard. 2021a. SemEval-2021 task 10: Source-free domain adaptation for semantic processing. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 348–356, Online. Association for Computational Linguistics.

Egoitz Laparra, Xin Su, Yiyun Zhao, Özlem Uzuner, Timothy Miller, and Steven Bethard. 2021b. SemEval-2021 task 10: Source-free domain adaptation for semantic processing. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 348–356, Online. Association for Computational Linguistics.

Egoitz Laparra, Dongfang Xu, Ahmed Elsayed, Steven Bethard, and Martha Palmer. 2018. SemEval 2018 task 6: Parsing time normalizations. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 88–96, New Orleans, Louisiana. Association for Computational Linguistics.

Limin Li and Zhenyue Zhang. 2019. Semi-supervised domain adaptation by covariance matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(11):2724–2739.

Jian Liang, Dapeng Hu, and Jiashi Feng. 2020. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6028–6039. PMLR.

Jinggui Liang, Lizi Liao, Hao Fei, Bobo Li, and Jing Jiang. 2024. Actively learn from LLMs with uncertainty propagation for generalized category discovery. 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 7845–7858, Mexico City, Mexico. Association for Computational Linguistics.

Chengyuan Liu, Fubang Zhao, Kun Kuang, Yangyang Kang, Zhuoren Jiang, Changlong Sun, and Fei Wu. 2024. Evolving knowledge distillation with large language models and active learning. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6717–6731, Torino, Italia. ELRA and ICCL.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Yen-Fu Luo, Sam Henry, Yanshan Wang, Feichen Shen, Ozlem Uzuner, and Anna Rumshisky. 2020. The 2019 n2c2/umass lowell shared task on clinical concept normalization. *Journal of the American Medical Informatics Association*, 27(10):1529–e1.

Timothy Miller, Dmitriy Dligach, Steven Bethard, Chen Lin, and Guergana Savova. 2017. Towards generalizable entity-centric clinical coreference resolution. *Journal of Biomedical Informatics*, 69:251–258.

Marco Naguib, Xavier Tannier, and Aurélie Névéol. 2024. Few-shot clinical entity recognition in English, French and Spanish: masked language models outperform generative model prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6829–6852, Miami, Florida, USA. Association for Computational Linguistics.

Burr Settles. 2009. Active learning literature survey.

Seiji Shimizu, Shuntaro Yada, Lisa Raithel, and Eiji Aramaki. 2024. Improving self-training with prototypical learning for source-free domain adaptation on clinical text. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 1–13, Bangkok, Thailand. Association for Computational Linguistics.

Xin Su, Yiyun Zhao, and Steven Bethard. 2021. The University of Arizona at SemEval-2021 task 10: Applying self-training, active learning and data augmentation to source-free domain adaptation. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 458–466, Online. Association for Computational Linguistics.

Xin Su, Yiyun Zhao, and Steven Bethard. 2022. A comparison of strategies for source-free domain adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8352–8367, Dublin, Ireland. Association for Computational Linguistics.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. 2020. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272.

Yanshan Wang, Liwei Wang, Majid Rastegar-Mojarad, Sungrim Moon, Feichen Shen, Naveed Afzal, Sijia Liu, Yuqun Zeng, Saeed Mehrabi, Sunghwan Sohn, et al. 2018. Clinical information extraction applications: a literature review. *Journal of biomedical informatics*, 77:34–49.

Stephen Wu, Timothy Miller, James Masanz, Matt Coarr, Scott Halgrim, David Carrell, and Cheryl Clark. 2014. Negation's not solved: Generalizability versus optimizability in clinical natural language processing. *PLOS ONE*, 9(11):1–11.

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, Singapore. Association for Computational Linguistics.

Canwen Xu, Yichong Xu, Shuohang Wang, Yang Liu, Chenguang Zhu, and Julian McAuley. 2023. Small models are valuable plug-ins for large language models. *Preprint*, arXiv:2305.08848.

Linyi Yang, Shuibai Zhang, Zhuohao Yu, Guangsheng Bao, Yidong Wang, Jindong Wang, Ruochen Xu, Wei Ye, Xing Xie, Weizhu Chen, and Yue Zhang. 2024. Supervised knowledge makes large language models better in-context learners. In *The Twelfth International Conference on Learning Representations*.

M. Yin, B. Wang, Y. Dong, and C. Ling. 2022. Source-free domain adaptation for question answering with masked self-training. *Preprint*, arXiv:2212.09563.

Zhiqi Yu, Jingjing Li, Zhekai Du, Lei Zhu, and Heng Tao Shen. 2023. A comprehensive survey on source-free domain adaptation. *Preprint*, arXiv:2302.11803.

Bo Zhang, Xiaoming Zhang, Yun Liu, Lei Cheng, and Zhoujun Li. 2021. Matching distributions between model and data: Cross-domain knowledge distillation for unsupervised domain adaptation. 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 5423–5433, Online. Association for Computational Linguistics.

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, Singapore. Association for Computational Linguistics.

Zishuo Zhao, Ziyang Ma, Zhenzhou Lin, Jingyou Xie, Yinghui Li, and Ying Shen. 2024. Source-free domain adaptation for aspect-based sentiment analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15076–15086, Torino, Italia. ELRA and ICCL.

Pierre Zweigenbaum, Dina Demner-Fushman, Hong Yu, and Kevin B Cohen. 2007. Frontiers of biomedical text mining: current progress. *Briefings in bioinformatics*, 8(5):358–375.

A Appendix

A.1 Implementation Details

A.1.1 Source SLMs Training

For NER and RE tasks, we fine-tune the source SLMs on the respective source datasets. Specifically, we fine-tune the RoBERTa-base model using Huggingface Trainer[6]. The key hyperparameters are listed in Table 8, while default values are used for the remaining hyperparameters.

Parameter	Value
Learning rate	2e-5
Number of training epochs	10
Weight Decay	0.01
Max length (RE)	340
Max length (NER)	500
Training batch size	4

Table 8: Training hyperparameters for the source SLMs.

A.1.2 Entropy-based Sampling

We calculate per-sample entropy using Scipy[7] (Virtanen et al., 2020) based on the softmax probabilities of target samples. For text classification tasks (ND and RE), entropy is computed directly from the softmax probabilities of each instance. For token classification tasks, per-token entropy is first calculated using the softmax probabilities of individual tokens, and the average entropy across tokens is used as the instance-level entropy.

A.1.3 LLM Annotation

For annotation with LLMs, we used the Huggingface Transformers pipeline[8] with default generation hyperparameters and set the temperature to 0.1. LLMs were queried using templates exemplified in the Fig 5, 6, 7 and 8. We present templates for NER and RE as examples of token and text classification tasks, respectively. For NER and TER tasks, we generated annotations in JSON format, inspired by (Kim et al., 2024). In the LLM-active method, we first generated example texts and their annotations using prompts in the figures. Specifically, 100 texts were randomly sampled from the development set of the target datasets for the example texts. Example prompts are presented in Fig. 9 and 10.

A.1.4 Fine-tuning of Source SLMs

The implementation of fine-tuning with LLM-annotated data follows the same procedure as the source SLM training for NER and RE tasks. For ND and TER tasks, we adopted the hyperparameters from Su et al. (2022).

A.2 Full Comparison of Zero-shot Annotation Quality and Source SLM performance

Table 9 presents the full comparison of zero-shot annotation quality and source SLM performance, as discussed in Sect 2. Zero-shot annotation surpasses the source SLM performance in text classification tasks (RE and ND) at an annotation budget of 384 but falls short in token classification tasks (NER and TER). This observation motivated us to improve LLM annotation with SLM assistance.

A.3 LLM Inference on Evaluation Sets

Table 10 compares "Active + SALA" with LLM inference on the evaluation sets of the target datasets. For few-shot inference, we retrieve class-wise examples from a manually annotated development set using the same retrieval method as SALA. While LLM inference achieves high F_1 scores in simpler tasks like ND, it struggles with more complex tasks such as NER and TER, even with few-shot examples. This highlights the advantage of using fine-tuned SLMs for clinical information extraction, particularly given their computationally efficient inference when processing large-scale clinical data.

A.4 Additional Evaluation of SLM Performance

Table 11 presents the additional results of SLM performance for active learning using various LLM annotation methods. With the number of annotations set to 96, SALA has no clear advantage over other LLM annotation methods. However, with an increased number of annotations, SALA demonstrates an overall advantage, as all methods perform comparably in text classification tasks, while SALA exhibits a prominently better performance in token classification tasks.

[6] https://huggingface.co/docs/transformers/en/main_classes/trainer

[7] https://scipy.org/

[8] https://huggingface.co/docs/transformers/en/main_classes/text_generation

A.5 Additional Evaluation of Annotation Quality

Table 12 presents the additional results of annotation quality for various LLM annotation methods on annotation targets, with the number of annotations set to 384 instances. The results include comparisons between medical domain and general domain LLMs for each annotation method. A similar trend is observed as in the SLM performance, providing additional evidence of the effectiveness of SALA.

Method	#	NER		RE		ND		TER		Avg
		Beth	*Part*	*Beth*	*Part*	*i2b2*	*mimic*	*Food*	*News*	
Source	96	51.5	39.2	23.2	39.2	85.1	80.9	**75.1**	**70.1**	58.0
Zero-Shot	96	52.4	**67.5**	50.0	**72.3**	**89.2**	82.3	44.5	31.7	**61.2**
Zero-Shot$_{\text{Med}}$	96	**54.6**	46.1	**56.4**	66.1	**89.2**	**89.7**	31.9	20.4	56.8
Source	384	**64.1**	**66.8**	32.8	51.8	88.8	65.2	**79.3**	**74.5**	**65.4**
Zero-Shot	384	49.3	58.5	56.2	67.8	94.3	90.5	55.1	37.4	63.6
Zero-Shot$_{\text{Med}}$	384	49.9	44.2	**60.9**	**68.6**	**94.5**	**92.8**	39.1	24.4	59.3

Table 9: Performance comparison of zero-shot annotation and the source SLMs.

Method	Human-free	NER		RE		ND		TER		Avg
		Beth	*Part*	*Beth*	*Part*	*i2b2*	*mimic*	*Food*	*News*	
Full-supervision (skyline)	×	88.7	86.7	75.4	65.7	89.8	88.7	85.8	83.0	83.0
Source	✓	81.3	80.9	56.1	57.3	84.6	63.5	78.1	78.4	72.5
Zero-shot	✓	57.8	50.6	59.3	**71.5**	91.1	**81.2**	53.4	52.0	64.6
Few-shot	×	61.2	55.6	62.2	66.7	**92.9**	80.3	62.5	69.9	68.9
Active + SALA$_{\text{Med}}$	✓	83.2	81.3	72.2	57.6	88.3	79.6	79.6	**78.7**	77.6
Active + SALA	✓	**84.2**	**83.2**	**72.3**	66.2	88.0	78.6	**79.9**	75.1	**78.4**

Table 10: LLM inference performance on the evaluation set of the target datasets.

Method	#	NER		RE		ND		TER		Avg
		Beth	*Part*	*Beth*	*Part*	*i2b2*	*mimic*	*Food*	*News*	
Full-supervision (skyline)	Full	88.7	86.7	75.4	65.7	89.8	88.7	85.8	83.0	83.0
Source	-	81.3	80.9	56.1	57.3	84.6	63.5	78.1	78.4	72.5
Active + Zero-shot$_{\text{Med}}$	96	82.6	81.2	63.9	61.5	87.9	75.7	75.0	72.5	**75.0**
Active + Zero-shot	96	**83.4**	80.0	61.0	59.3	87.9	75.6	76.9	59.0	72.9
Active + LLM-active$_{\text{Med}}$	96	83.6	79.1	63.1	61.6	87.9	75.7	75.2	67.2	74.2
Active + LLM-active	96	83.1	**82.2**	61.4	59.5	**88.2**	**76.0**	74.6	74.6	**75.0**
Active + SALA$_{\text{Med}}$	96	82.5	80.1	**63.9**	59.0	87.8	68.1	**80.3**	**77.5**	74.6
Active + SALA	96	81.9	81.5	60.2	**61.3**	85.0	71.1	79.3	76.8	74.6
Active + Zero-shot$_{\text{Med}}$	384	71.6	63.0	63.3	**67.2**	**88.5**	81.4	74.7	64.1	71.7
Active + Zero-shot	384	69.5	68.7	69.5	66.5	88.0	**82.8**	75.2	55.8	72.0
Active + LLM-active$_{\text{Med}}$	384	66.9	62.6	64.2	58.8	88.3	80.2	70.0	37.5	66.1
Active + LLM-active	384	70.7	62.8	66.4	56.5	88.1	80.6	71.9	63.0	70.0
Active + SALA$_{\text{Med}}$	384	83.2	81.3	72.2	57.6	88.3	79.6	79.6	**78.7**	77.6
Active + SALA	384	**84.2**	**83.2**	**72.3**	66.2	88.0	78.6	**79.9**	75.1	**78.4**

Table 11: Performance comparison of the adapted SLMs with varous LLM annotation methods.

Method	NER		RE		ND		TER		Avg
	Beth	*Part*	*Beth*	*Part*	*i2b2*	*mimic*	*Food*	*News*	
Few-shot	56.3	61.7	57.6	65.0	93.7	92.3	52.4	47.4	65.8
Zero-shot	49.3	58.5	56.2	67.8	94.3	90.5	55.1	37.4	63.6
Zero-shot$_{\text{Med}}$	49.9	44.2	**60.9**	68.6	94.5	**92.8**	39.1	24.4	59.3
LLM-active	53.0	59.0	56.3	64.8	**94.8**	89.9	49.4	45.0	64.0
LLM-active$_{\text{Med}}$	50.9	47.5	57.6	68.3	**94.8**	90.6	48.0	22.9	60.1
SALA	**65.8**	**74.6**	56.9	66.3	93.7	86.7	78.6	70.2	74.1
SALA$_{\text{Med}}$	65.2	69.5	56.8	**70.4**	93.7	90.6	**81.6**	**73.8**	**75.2**

Table 12: Annotation quality comparison of various LLM annotation methods.

Instructions: You are an intelligent clinical language model.
Given the entity label set: [<label set>], please recognize the named entities in the given clinical text.
<task description>
Here are examples of the annotation.
<high-confidence examples>
Provide the answer in the following lines of JSON format:
{entity name1: entity type1}
{entity name2: entity type2}
Extract the entity name from the text exactly.
Be sure to choose the entity type from [<label set>].
Be sure to keep the order of the entities as they appear in the text.
If there are no entities in the entire text, return the empty JSON: {}.
Now, please recognize the named entities in the following clinical text
Text: "<input>"
Here is the base answer
Answer:
<SLM prediction>
If the base answer needs modification, please return the modified answer based on the annotation task description and the examples.
If the base answer is correct, please return the answer as it is.
Answer:

Figure 5: The template of SALA prompt for NER.

Instructions: You are an intelligent clinical language model.
Given the entity label set: [<label set>], please recognize the named entities in the given clinical text.
<task description>
Provide the answer in the following lines of JSON format:
{entity name1: entity type1}
{entity name2: entity type2}
Extract the entity name from the text exactly.
Be sure to choose the entity type from [<label set>].
Be sure to keep the order of the entities as they appear in the text.
If there are no entities in the entire text, return the empty JSON: {}.
Now, please recognize the named entities in the following clinical text
Text: "<input>"
Answer:

Figure 6: The template of zero-shot prompt for NER.

Instructions: You are an intelligent clinical language model.
<task description>
Here are examples of the annotation.
<high-confidence examples>
Classify the relation of the two concepts marked with <e1> </e1> and <e2> </e2>.
Provide the answer by choosing one word from the following categories: [<label set>]
Do not include anything other than the category in the answer.
Now, please classify the following text
Text: "<input>"
Here is the base answer
Answer:
<SLM prediction>
If the base answer needs modification, please return the modified answer based on the annotation task description and the examples.
If the base answer is correct, please return the answer as it is.
Answer:

Figure 7: The template of SALA prompt for RE.

Instructions: You are an intelligent clinical language model.
<task description>
Classify the relation of the two concepts marked with <e1> </e1> and <e2> </e2>.
Provide the answer by choosing one word from the following categories: [<label set>]
Do not include anything other than the category in the answer.
Now, please classify the following text
Text: "<input>"
Answer:

Figure 8: The template of zero-shot prompt for RE.

Instructions: You are an intelligent clinical language model.
Given an annotation guideline and example texts below, please generate an example text with annotations.
<task description>
Here are the example texts.
<examples>
Provide the answer in the following lines of JSON format:
{entity name1: entity type1}
{entity name2: entity type2}
Extract the entity name from the text exactly.
Be sure to choose the entity type from [<label set>].
Be sure to keep the order of the entities as they appear in the text.
If there are no entities in the entire text, return the empty JSON: {}.
The generated text and answer should be in the format:
Text: "Example text here"
Answer:
{entity name1: entity type1}
{entity name2: entity type2}
Now, please generate an example text with annotations.
Text:"

Figure 9: The template of prompt for generating example texts and their annotations in NER.

Instructions: You are an intelligent clinical language model.
Given an annotation guideline and example texts below, please generate an example text with an annotation.
<task description>
Here are the example texts.
<examples>
Classify the relation of the two concepts marked with <e1> </e1> and <e2> </e2>.
Provide the answer by choosing one word from the following categories: [<label set>]
Do not include anything other than the category in the answer.
The generated text and answer should be in the format:
Text: "Example text here"
Answer: <answer>
Now, please generate an example text with an annotation.
Text:"

Figure 10: The template of prompt for generating example texts and their annotations in RE.