Mind Your Steps in Biomedical Named Entity Recognition: First Extract, Tag Afterwards

Darya Shlyk[1], Stefano Montanelli[1], Marco Mesiti[1], Lawrence Hunter[2]
[1]Università degli Studi di Milano, Via Giovanni Celoria 19, 20133 Milan, Italy,
[2]The University of Chicago, 5801 South Ellis Avenue, Chicago, IL, 60637, USA
Correspondence: darya.shlyk@unimi.it

Abstract

Few-shot prompting with Large Language Models (LLMs) has emerged as a promising paradigm for advancing information extraction, particularly in data-scarce domains like biomedicine, where high annotation costs constrain the availability of training data. However, challenges persist in biomedical Named Entity Recognition (NER), where LLMs fail to achieve necessary accuracy and lag behind supervised fine-tuned models. In this study, we introduce FETA (*First Extract, Tag Afterwards*), a two-stage approach for entity recognition that combines instruction-guided prompting and a novel self-verification strategy to improve accuracy and reliability of LLM predictions in domain-specific NER tasks. FETA achieves state-of-the-art results on multiple established biomedical datasets. Our experiments demonstrate that carefully designed prompts, using self-verification and instruction guidance, can steer general-purpose LLMs to outperform fine-tuned models in knowledge-intensive NER tasks, unlocking their potential for more reliable and accurate information extraction in resource-constrained settings.

1 Introduction

Biomedical named entity recognition (BioNER) is a challenging and important real-world task, presenting unique difficulties compared to general-domain NER. First, the sheer number of named entities is vast, encompassing hundreds of thousands of gene and protein names, millions of species names, and numerous cell types, metabolites, biological processes, diseases, drugs and so on. Second, many terms are ambiguous with respect to class (e.g., *adenomatous polyposis coli* is a gene, a protein, and a disease name), are indicated by different parts of speech (e.g. *nucleus and nuclear*), and otherwise exhibit unusual variability and ambiguity. Furthermore, scientific publications make extensive use of locally defined abbreviations that can be generally quite ambiguous (e.g., *AS* may stand for *Angelman syndrome*, *ankylosing spondylitis*, or *aortic stenosis* among others).

In many tasks, including named entity recognition, few-shot prompting with Large Language Models (LLMs) can achieve state-of-the-art performance (Brown et al., 2020). However, in the biomedical domain, these methods still struggle with accuracy and lag behind conventional approaches that require extensive supervision with expert-annotated training data. (Gutierrez et al., 2022; Moradi et al., 2021; Keloth et al., 2024).

In this work, we present FETA (*First Extract, Tag Afterwards*), a two-staged prompting approach that integrates instructional guidance with an innovative context-aware self-verification mechanism to guide off-the-shelf LLMs to perform accurate and verifiable biomedical information extraction without domain-specific training. FETA is designed to work in real-world biomedical applications, where labeled data is often scarce, expensive to obtain, and requires expert domain knowledge.

Building on recent advances in prompt engineering (Gero et al., 2023; Munnangi et al., 2024), we leverage prompt chaining to decompose BioNER into manageable sub-tasks, where the first handles span detection of target biomedical entities, while the second aims to disambiguate candidate mentions for accurate and reliable NER results. Inspired by the way humans use annotation guidelines to perform complex labeling tasks, FETA leverages natural language instructions designed by domain experts to guide the span detection process. Unlike the prevailing example-driven paradigm that relies on few-shot demonstrations for in-context domain adaptation, guideline-based prompting aims to capitalize on the instruction-following abilities of generative LLMs. Additionally, using explicit guidelines helps mitigate the risk of annotation bias, where a few misleading examples in the prompt can often result in unstable or incorrect model behavior.

Proceedings of the 1st Workshop on Linguistic Analysis for Health (HeaLing 2026), pages 127–141
March 28, 2026 ©2026 Association for Computational Linguistics

Furthermore, unlike existing high-cost multi-stage verification pipelines (Kim et al., 2024; Munnangi et al., 2024), FETA eliminates the need for external knowledge bases or entity linking tools for post-hoc verification, and introduces a lightweight and scalable self-verification mechanism, that grounds entity disambiguation in the original context, allowing the LLM to validate and correct its own predictions in a single pass. We empirically demonstrate the effectiveness of the proposed framework through an extensive experimental analysis, evaluating multiple established biomedical benchmarks, comparing against SOTA BioNER methods, and testing across a range of commercial and open-source LLMs of varying size.

Our contributions can be summarized as follows:

- We propose FETA, a prompt-based approach that combines instructional guidance and self-verification to unlock the LLM potential for knowledge-intensive biomedical information extraction without domain-specific training.
- In addition to surpassing fine-tuned, domain-adapted BioNER methods, FETA significantly improves computational efficiency compared to similar LLM-based frameworks.
- Through extensive experiments evaluating off-the-shelf LLMs of varying scale, we show that FETA can achieve competitive results even with moderately-sized language models.

2 Related Work

2.1 Prompt Engineering for NER

Prompt engineering has been instrumental in adapting LLMs for NER, leveraging their zero- and few-shot learning capabilities (Gong, 2024; Xie et al., 2023). Several studies have explored different prompting strategies to align generalist text generation models with the structured nature of entity extraction. A common approach involves structured output generation, where models extract entity spans in predefined formats, such as lists or JSON-like schemas (Wei et al., 2023; Ashok and Lipton, 2023; Agrawal et al., 2022). These methods often require additional post-processing to validate outputs and lack inherent positional information for extracted entities. Recent work explored alternative NER-specific task formulations that better align with LLM pre-training objectives. Li et al. (2023) leverage LLMs specialized in code generation framing NER as a code generation task. An alternative strategy treats NER as a sequence tagging task, prompting LLMs to annotate entities inline within the original text using delimiters (Wang et al., 2023; Hu et al., 2024b). This method aligns with traditional BIO-tagging schemes used in supervised NER (Luo et al., 2023), retaining positional information and facilitating post-processing. Recent studies (Cheng et al., 2024; Monajatipoor et al., 2024; Min et al., 2022) explored different example selection schemes to improve LLM's in-context learning abilities. However, selecting optimal examples remains inherently challenging and can lead to unstable or suboptimal model performance. To mitigate this, **rather than relying on few-shot demonstrations, we propose to leverage the instruction-following capabilities of LLMs by enhancing prompts with domain-specific guidelines to aid generalization**.

2.2 Biomedical NER with LLMs

General-purpose LLMs often struggle with biomedical NER due to a lack of domain-specific training and a tendency to hallucinate erroneous predictions (Hu et al., 2024a). To mitigate hallucinations, recent studies explored self-verification via prompt chaining, where follow-up prompts refine initial extractions (Averly and Ning, 2024; Gero et al., 2023; Bian et al., 2023, 2024). Other approaches enhance LLM with external knowledge for more accurate information extraction (Nagar et al., 2024; Fu et al., 2023). Multi-step verification frameworks, such as VerifiNER (Kim et al., 2024), use LLMs to validate predictions by reasoning over entity-level information retrieved from external knowledge bases. Similarly, (Munnangi et al., 2024) prompt LLMs to refine extractions using structured entity definitions from external sources. While knowledge-augmented verification improves accuracy, these methods depend on the availability of entity-level information in structured knowledge sources and an effective entity-linking mechanism. Additionally, entity-level revision with knowledge-augmented prompts has an added cost in terms of multiple inference steps that can significantly increase computational costs, particularly when using proprietary LLMs. In contrast, we propose a **more efficient approach that grounds self-verification in context, reducing the reliance on external knowledge sources and addressing computational inefficiencies with prior multi-stage frameworks**.

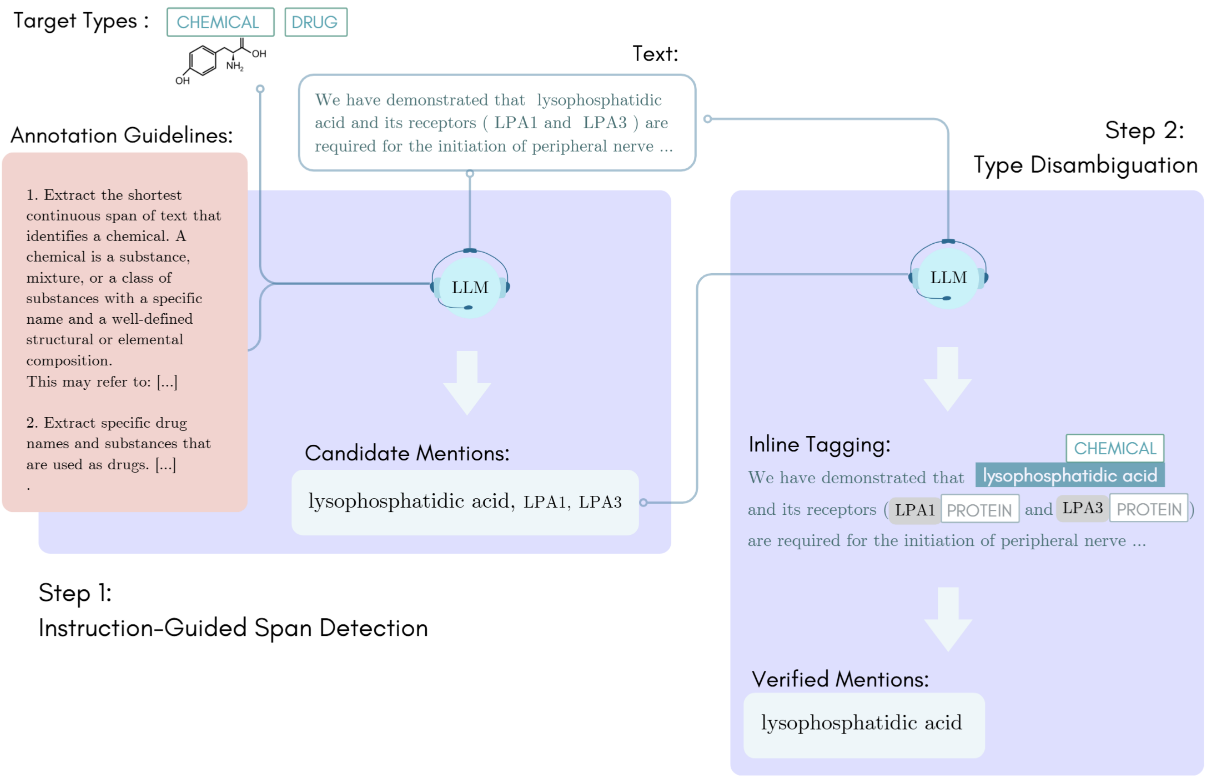
Target Types :
CHEMICAL
DRUG
OH
NH2
OH
Text:
We have demonstrated that lysophosphatidic acid and its receptors (LPA1 and LPA3) are required for the initiation of peripheral nerve ...
Annotation Guidelines:
1. Extract the shortest continuous span of text that identifies a chemical. A chemical is a substance, mixture, or a class of substances with a specific name and a well-defined structural or elemental composition.
This may refer to: [...]
2. Extract specific drug names and substances that are used as drugs. [...]
LLM
Candidate Mentions:
lysophosphatidic acid, LPA1, LPA3
Step 1:
Instruction-Guided Span Detection
Step 2:
Type Disambiguation
LLM
Inline Tagging:
CHEMICAL
We have demonstrated that lysophosphatidic acid
and its receptors (LPA1 PROTEIN and LPA3 PROTEIN)
are required for the initiation of peripheral nerve ...
Verified Mentions:
lysophosphatidic acid

Figure 1: Example of the FETA approach for Biomedical Named Entity Recognition. Given a text passage and a pre-defined set of target biomedical enity types, (1) we extract candidate entity mentions from the text using an LLM prompt enriched with expert-designed domain-specific annotation guidelines; (2) then, using a follow-up prompt, we task the LLM to tag the extracted mentions within the original text, specifying their semantic types (*inline tagging*). Finally, we parse the resulting annotations and filter out candidate mentions based on their predicted type to produce the final set of verified mentions of the target types.

3 Method Overview

The FETA approach is designed to address the task of biomedical entity recognition, which aims to identify a list of entity mentions $M = [m_1, \ldots, m_n]$, that are instances of predefined biomedical types T in a given input text I. Formally, an entity mention m_j is a tuple $\langle s_j, e_j, t_j \rangle$, where s_j and e_j denote the start and end character offsets of the entity span in I, and $t_j \in T$ is the entity type of that span (e.g., `disease`, `gene`, `chemical`, etc.), drawn from a set of target biomedical types of interest $T = \{t_1, \ldots, t_k\}$. FETA uses a prompt-chaining technique to solve the task in two subsequent steps, *Instruction-Guided Span Detection* and *Type Disambiguation*, each executed via a separate LLM call.

1. **Instruction-Guided Span Detection**:

   In this step, we prompt the LLM to extract a list of candidate entity spans C from text I given a set of target entity types T. The LLM prompt is enriched with expert knowledge in the form of natural language annotation guidelines. Unlike traditional few-shot prompting, explicit guidelines provide an unambiguous description of target entity types and clarify span detection criteria for correct span extraction. Incorporating these instructions within the prompt, has the potential to enhance the model's ability to correctly recognize mentions of specific biomedical types in the absence of domain-specific training. The resulting list of candidate spans C contains spans of I that the LLM identifies to be instances of the target types in T. Due to the LLM's tendency toward false positives (Brown et al., 2020), the candidate list C may include spurious entity spans, i.e., mentions of biomedical entities wrongly extracted as instances of T. These false positives often arise from strong semantic bonds reflecting biological interactions between different classes of biomedical entities. For instance, proteins engage in numerous chemical interactions and are frequently mentioned together with chemicals in the literature, which can lead to an over-representation of protein mentions as false positives when the target biomedical type is `chemical`. An example of such behavior is illustrated in Figure 1, where the LLM extracts two protein mentions, *LPA1* and *LPA3*, along with *lysophosphatidic acid* as instances of type `chemical` or `drug`. Similar error patterns can be observed with other biomedical types, posing the need for a type disambiguation strategy to refine the initial list of candidate mentions.

2. **Type Disambiguation**: The goal of the second step is to produce a list of entity mentions M by detecting and removing false positives from the candidate list C. To this end, we prompt the LLM to disambiguate the type of each candidate in C via inline tagging. Namely, we instruct the LLM to annotate all candidate spans, extracted in the previous step, within the original text passage I using HTML-like tags that (i) mark the exact boundaries of the candidate span in I, and (ii) specify the entity type of a candidate span based on its surrounding context. During post-processing of the tagging results, any candidate span assigned a type $t \notin T$ is discarded.

   This strategy offers two key advantages. Firstly, it elicits contextual reasoning in the LLM using the input text alone, without the need for external domain-specific knowledge bases. By requiring the LLM to tag candidate spans in the source text, the prompt forces the model to reassess the type of each candidate in a context-aware manner, leveraging the context surrounding the span to disambiguate its type. Secondly, in contrast to similar prompt-based methods that incur high costs by invoking LLMs for each mention individually, inline-tagging enables to disambiguate all candidate mentions at once with a single LLM call, leading to improved efficiency.

3.1 Instruction-Guided Span Detection

For the span detection step, we define a prompt template (illustrated in Appendix, Figure 4) consisting of several components, including: (i) *Task Description*: it specifies the entity extraction objective, listing the target entity types T for recognition (e.g., diseases and symptoms, proteins, chemicals); (ii) *Output Format*: this component describes the expected structure of the output formatting to facilitate reliable parsing of the LLM-generated response. Following prior work (Wang et al., 2023; Munnangi et al., 2024), we adopt a JSON-style output format; (iii) *Example*: it illustrates the desired model behavior and the expected output structure; (iv) *Annotation Guidelines*: a list of natural language instructions derived from expert annotation guidelines. These guidelines define the scope of each target biomedical entity type (e.g., *a chemical is a substance, mixture, or a class of substances with a specific name and a well-defined structural or elemental composition.*) and outline criteria for consistent span boundary identification (e.g., *if a chemical's name is followed by its abbreviation, extract both separately*). The complete set of guidelines used in our experiments is included in the Appendix, Table 6 and 7.

Consider the example in Figure 1. Given an input text passage I, a target set of entity types T = `{CHEMICAL, DRUG}`, and the corresponding expert-defined annotation guidelines, we construct the prompt using the template in Figure 4 in Appendix and call the LLM to generate a JSON formatted output containing a list of candidate mentions C = *[lysophosphatidic acid, LPA1, LPA3].*

3.2 Type Disambiguation

The prompt template for the second step includes the task description and a one-shot annotated example that illustrates the task (see Appendix, Figure 5). The model is provided with a list of candidate entity spans C extracted in the first step along with the original text passage I. The LLM's task is to return the passage with all candidate spans marked inline using `<entity>...</entity>` tags. For each marked span, the model must specify the appropriate type using the tag attribute. The task description clearly defines the type labels, that should be used for the target entity types. For example, it instructs: *"Use* `<entity type="CHEMICAL"></entity>` *to tag chemical entities"*. Spans annotated with one of the target types will be retained during post-processing. Conversely, any span assigned to a type $t \notin T$ is considered a false positive and is discarded from the final set of entity mentions. Most of discarded spans correspond to distant or unrelated biomedical entity types outside the scope of the target category. For instance, when the target entity type is `CHEMICAL` or `DRUG`, entities such as proteins, genes, or cells are frequently filtered as false positives. To ensure consistency and efficiency, our filtering strategy also removes entity spans whose predicted labels are syntactic variants of the target type (e.g., “DRUGS” instead of “DRUG”,” or “THE DRUG”). However, such instances are exceedingly rare, which discourages the need for a more sophisticated normalization strategy.

Considering the example in Figure 1, the second step generates a markup copy of the input text, where only *lysophosphatidic acid* is tagged as an instance of the target type in T, while *LPA2* and *LPA3* are correctly disambiguated as instances of type protein, and are thus excluded from the final list of entity mentions M.

4 Experimental Setup

4.1 Benchmark Corpora

To evaluate the effectiveness of the FETA approach, we used several well-established biomedical corpora for NER, spanning both abstract-level texts and full-text articles to ensure a comprehensive evaluation across different application scenarios. These include, **BC5CDR** (Li et al., 2016), a corpus of 1500 PubMed abstracts annotated for disease, symptom, and chemical entity types; **NLM-Chem** corpus (Islamaj et al., 2022), a collection of 150 full-text PubMed Central articles, annotated for chemical and drug mentions; **NLM-Gene** (Islamaj et al., 2021) a corpus of 550 PubMed abstracts manually annotated for gene and protein mentions; **GENIA** (Kim et al., 2003), a collection of 2000 MEDLINE abstracts annotated for several entity types, including cell_line, cell_type, DNA, RNA, and protein. We adhere to the original train-validation-test splits, using validation sets for prompt optimization and test sets for final evaluation. As part of preprocessing, we split documents into sentences, presenting each sentence individually in the prompt for annotation. The dataset statistics are detailed in Table 5 in Appendix.

4.2 Models

We test FETA using a selection of off-the-shelf general-purpose LLMs, balancing efficiency and performance by including both moderate-sized and large-scale models from leading open and commercial providers. For compact yet competitive models, we consider OpenAI’s GPT-4o Mini (Hurst et al., 2024), Google’s Gemini Flash 1.5 (8B) (Reid et al., 2024), and Google’s Gemma 2 (9B) (Team et al., 2024), which have demonstrated strong performance relative to larger counterparts. As a large-scale representative, we include Meta’s LLaMA 3.3 (70B) (Dubey et al., 2024). For all models, we access the instruction-tuned version via API calls.

4.3 Evaluation Metrics

We evaluate all models using standard evaluation metrics, such as precision (P), recall (R), and F1 score. Detailed definitions and formulas are provided in Appendix 7 Since strict (exact) matching penalizes even minor misalignment in entity boundaries, we also include relaxed (partial) match metrics to account for approximate but valid LLM-based span predictions. Unlike exact matching, partial matching allows for slight boundary mismatches between the predicted entity span and the gold standard. For example, given the ground truth "*heterogeneous neurological disease*", the prediction "*neurological disease*" would be considered correct under partial matching but incorrect under exact matching.

5 Experimental Results

The following experiments are designed to: (i) probe the staged prompt design, testing the effectiveness of task decomposition on LLM performance for biomedical NER; (ii) assess the potential of the prompt-based FETA approach in steering off-the-shelf generative LLMs to compete with state-of-the art BioNER systems; (iii) isolate the contributions of key design choices of FETA with focus on instruction guidance in span detection and effectiveness of type disambiguation for post-hoc verification, and (iv) compare it against alternative prompt-based verification methods.

5.1 Evaluating the effectiveness of the staged FETA approach for biomedical NER

To quantify the contribution of the prompt chaining technique, i.e., the sequential decomposition of span detection and type disambiguation, on the overall performance of LLMs in the context of biomedical NER, we compare FETA against the following single-stage prompting baselines:

- **Extract-Only**: This strategy corresponds to the first stage of the FETA pipeline. It prompts the LLM to extract mentions of the target biomedical types without performing any subsequent validation of the extracted entity spans (see Figure 6 in Appendix). This baseline helps to elucidate the value of the additional revision step for improving the accuracy of initial LLM predictions.

		Exact Match			Partial Match		
Model	Strategy	BC5CDR P / R / F1	NLM-Chem P / R / F1	NLM-Gene P / R / F1	BC5CDR P / R / F1	NLM-Chem P / R / F1	NLM-Gene P / R / F1
Proprietary Language Models							
Gpt4o-mini	*All-in-One*	35.9 / 62.6 / 46.7	48.3 / 75.4 / 58.9	63.5 / 81.2 / 71.3	51.5 / 89.7 / 65.4	54.2 / 84.1 / 65.9	70.0 / 89.0 / 78.4
	Extract-Only	51.8 / 66.4 / 58.2	58.1 / 74.8 / 65.4	66.4 / 80.8 / 72.9	68.4 / 88.0 / 77.0	64.2 / 82.2 / 72.1	74.0 / 89.5 / 81.1
	FETA	61.9 / 67.6 / 64.6	79.1 / 73.1 / 76.0	82.2 / 80.3 / 81.3	81.4 / 89.2 / 85.1	86.0 / 79.0 / 82.4	90.1 / 88.7 / 89.4
Gemini Flash 1.5	*All-in-One*	62.2 / 49.7 / 55.3	59.9 / 64.1 / 61.9	69.0 / 77.1 / 72.8	80.6 / 65.0 / 72.0	67.6 / 71.9 / 69.7	77.3 / 85.9 / 81.3
	Extract-Only	65.0 / 60.0 / 62.3	73.9 / 58.8 / 65.5	77.2 / 73.5 / 75.3	81.0 / 74.9 / 77.8	80.7 / 63.9 / 71.3	85.6 / 81.1 / 83.3
	FETA	69.8 / 69.3 / 69.5	82.7 / 63.4 / 71.8	83.4 / 73.5 / 78.1	85.8 / 85.6 / 85.7	89.8 / 68.5 / 77.7	92.3 / 81.2 / 86.4
Open-Source Language Models							
Gemma 2	*All-in-One*	65.9 / 53.3 / 58.9	66.0 / 65.2 / 65.6	82.4 / 72.2 / 77.0	82.5 / 67.3 / 74.1	73.4 / 72.4 / 72.9	89.6 / 79.4 / 84.2
	Extract-Only	72.6 / 63.5 / 67.8	70.1 / 67.1/ 68.5	86.0 / 77.1 / 81.2	86.5 / 76.1 / 80.9	76.2 / 72.9 / 74.5	90.8 / 82.2 / 86.3
	FETA	71.5 / 71.6 / 71.6	79.0 / 71.3 / 74.9	86.4 / 80.2 / **83.2**	85.2 / 85.8 / 85.5	86.4 / 77.9 / 82.0	92.9 / 86.1 / 89.4
LLaMa 3.3	*All-in-One*	56.0 / 70.2 / 62.3	50.8 / 81.7 / 62.6	67.7/ 80.0/ 73.36	72.0 / 91.8 / 80.7	55.5 / 88.6 / 68.3	75.1 / 89.5 / 81.7
	Extract-Only	74.3 / 75.4 / 74.8	76.2 / 74.5 / 75.4	83.7 / 79.3 / 81.4	86.2 / 87.7 / 86.9	81.0 / 78.6 / 79.8	90.3 / 85.7 / 87.9
	FETA	73.1 / 79.3 / **76.1**	84.9 / 80.1 / **82.4**	87.2 / 79.0 / 82.9	85.1 / 92.6 / **88.7**	90.3 / 84.5 / **87.3**	94.2 / 85.8 / 89.8
Supervised Baselines							
BioBERT		69.6 / 65.6 / 67.5	80.3 / 68.2 / 73.7	80.9 / 83.4 / 81.6	86.7 / 81.7 / 84.2	84.1 / 72.5 / 77.9	92.6 / 95.2 / **93.9**
PubMedBERT		74.8 / 55.7 / 63.9	86.2 / 65.2 / 74.2	83.0 / 79.5 / 81.2	89.6 / 67.7 / 77.1	90.1 / 68.0 / 77.5	92.6 / 88.1 / 90.2
BioNER-LLaMA2		74.9 / 61.0 / 66.9	85.3 / 59.8 / 70.3	88.1 / 77.0 / 82.2	90.3 / 73.6 / 81.1	88.2 / 61.7 / 72.6	96.9 / 84.6 / 90.3

Table 1: Evaluation of the FETA approach. Best F1 scores for each dataset are reported in bold.

- **All-in-One**: In this strategy, the LLM is prompted to simultaneously perform both span detection and type disambiguation in a single step (see Figure 7 in Appendix). Unlike FETA, this approach does not rely on an intermediate set of candidate mentions. Instead, the model is directly asked to identify and annotate entities belonging to the target biomedical types. This setup serves to evaluate the LLM's multi-tasking ability and helps assess whether decomposing the NER task into distinct stages yields better results than a unified prompting approach.

Table 1 presents the results of the comparative evaluation of prompting strategies with different LLMs on three biomedical benchmarks using exact and partial matching criteria. Among the evaluated strategies, *All-in-One* shows the lowest performance, particularly in terms of precision. The substantial performance gap between *All-in-One* and the *FETA* approach underscores the difficulty LLMs face when required to simultaneously identify and disambiguate entity mentions in a single step. In contrast, the staged design used in *FETA*, which decomposes the task into two subtasks, allows more focused use of LLMs, and achieves the best results, consistently surpassing other strategies across all datasets and models. Additionally, when comparing *FETA* to the *Extract-Only* strategy, the benefit of the second validation step becomes evident. The disambiguation stage, implemented via a subsequent LLM call, improves the quality of the final predictions by filtering out false positives. These findings demonstrate the clear advantage of prompt chaining and task decomposition, supporting the hypothesis that breaking complex tasks into simpler, sequential steps enables more effective use of LLMs in complex biomedical information extraction tasks.

5.2 Comparison with supervised BioNER approaches

To understand how well our prompt-based approach, that uses off-the-shelf generative LLMs, performs relative to fully fine-tuned, domain-adapted systems, we compare FETA against several state-of-the-art BioNER methods: (i) **BioBERT** (Lee et al., 2019), a domain-specific BERT-base model pretrained on biomedical corpora and fine-tuned for biomedical NER task; (ii) **PubMedBERT** (Gu et al., 2020), another BERT-based BioNER model pretrained from scratch on PubMed articles. (iii) **BioNER-LLaMA 2** (Keloth et al., 2024), an instruction-tuned generative LLM, using LLaMA (7B) as a backbone, specifically trained to recognize and tag biomedical entities within the input text. For training details see Appendix 7. The evaluation results in Table 1 demonstrate the strong performance of the FETA approach across a range of LLMs compared to supervised baselines. Notably, the best-performing model, LLaMA 3.3, almost consistently outperforms fine-tuned BioNER systems across all benchmarks, often by a significant margin. These findings highlight

		Exact Match			Partial Match		
Model	Strategy	BC5CDR	NLM-Chem	NLM-Gene	BC5CDR	NLM-Chem	NLM-Gene
Proprietary Language Models							
GPT4o-mini	Base	58.2	65.4	72.9	77.0	72.1	81.1
	+ TD	63.6 (+ 5.4)	74.0 (+ 8.6)	81.2 (+ 8.3)	83.4 (+ 6.4)	81.4 (+ 9.3)	89.2 (+ 8.1)
	+ AG	60.2 (+ 2.0)	70.5 (+ 5.1)	74.4 (+ 1.5)	79.6 (+ 2.6)	76.7 (+ 4.6)	82.8 (+ 1.7)
	+ AG + TD	64.6 (+ 6.4)	76.0 (+ 10.6)	81.3 (+ 8.4)	85.1 (+ 8.1)	82.4 (+ 10.3)	89.4 (+ 8.3)
Gemini-Flash 1.5	Base	62.3	65.5	75.3	77.8	71.3	83.3
	+ TD	65.0 (+ 2.7)	67.9 (+ 2.4)	77.8 (+ 2.5)	80.3 (+ 2.5)	73.9 (+ 2.6)	85.7 (+ 2.4)
	+ AG	62.4 (+ 0.1)	71.0 (+ 5.5)	75.8 (+ 0.5)	79.7 (+ 1.9)	76.8 (+ 5.5)	84.1 (+ 0.8)
	+ AG & TD	69.5 (+ 7.2)	71.8 (+ 6.3)	78.1 (+ 2.8)	85.7 (+ 7.9)	77.7 (+ 6.4)	86.4 (+ 3.1)
Open-Source Language Models							
Gemma 2	Base	67.8	68.5	81.2	80.9	74.5	86.3
	+ TD	69.0 (+ 1.2)	72.4 (+ 3.9)	82.3 (+ 1.1)	82.2 (+ 1.3)	78.8 (+ 4.3)	88.0 (+ 1.7)
	+ AG	68.6 (+ 0.8)	70.7 (+ 2.2)	80.6 (- 0.6)	82.3 (+ 1.4)	77.5 (+ 3.0)	87.0 (+ 0.7)
	+ AG & TD	71.6 (+ 3.8)	74.9 (+ 6.4)	**83.2** (+ 2.0)	85.5 (+ 4.6)	82.0 (+ 7.5)	89.4 (+ 3.1)
LLaMa 3.3	Base	74.8	75.4	81.4	86.9	79.8	87.9
	+ TD	75.6 (+ 0.8)	79.2 (+ 3.8)	83.3 (+ 1.9)	87.7 (+ 0.8)	83.8 (+ 4.0)	89.6 (+ 1.7)
	+ AG	75.5 (+ 0.7)	79.5 (+ 4.1)	80.8 (- 0.6)	88.1 (+ 1.2)	84.3 (+ 4.5)	87.7 (- 0.2)
	+ AG & TD	**76.1** (+ 1.3)	**82.4** (+ 7.0)	82.9 (+ 1.5)	**88.7** (+ 1.8)	**87.3** (+ 7.5)	89.8 (+ 1.9)

Table 2: Ablating key components of the FETA approach: Base (no Type Disambiguation or Annotation Guidelines), TD (Span Detection w/o Annotation Guidelines + Type Disambiguation), AG (Span Detection w/ Annotation Guidelines, no Type Disambiguation), and AG & TD (Span Detection using Annotation Guidelines + Type Disambiguation). Reported values are F1 scores, with the delta with respect to the base case in parentheses. Best results for each dataset are reported in bold.

the promise of prompt engineering as a viable alternative to supervised learning for biomedical NER. By effectively guiding general-purpose LLMs through carefully designed instructions, FETA enables accurate biomedical information extraction without requiring domain-specific supervision with extensive training data. Moreover, open-source, moderately-sized models, like Gemma 2, perform on par with their commercial counterparts of similar size and can even rival larger models like LLaMA 3.3. This results point to untapped potential for the application of open-source LLMs in complex biomedical text mining tasks, suggesting a cost-effective alternative to commercial language models.

5.3 Ablation study: Isolating the contributions of key design choices in FETA

Table 2 presents ablation results that quantify the contributions of two key components of the FETA approach: (i) instructional guidance in the span detection step, and (ii) type disambiguation as a self-verification mechanism. When examined in isolation, each component independently improves NER performance, with disambiguation showing a more pronounced impact. On average, type disambiguation improves F1 scores by 3.5 (exact match) and 3.8 (relaxed match) percentage points, while instructional guidance provides smaller but consistent gains of 1.8 and 2.3 points, respectively. Combined, these components yield an average F1 improvement of 5.3 (exact) and 5.9 (relaxed), demonstrating their complementary roles. Error analysis in Figure 2 shows that guidelines help increase recall, though sometimes at the expense of precision, while the disambiguation step improves precision by effectively filtering out false positives.

To evaluate the impact of explicit annotation instructions versus few-shot examples on LLM performance, we systematically varied the number of in-context examples (0, 1, or 5) in the span detection step, with and without annotation guidelines. We report the evaluation results for different numbers of N-shots in Figure 3 in Appendix. Our findings reveal that prompts incorporating explicit expert-designed annotation guidelines consistently outperform few-shot prompting alone regardless of the number of annotated examples. Remarkably, a prompt with explicit instructions and just one example outperforms prompts with five annotated examples but no guidelines, underscoring the superior effectiveness of instruction-guided prompting for in-context domain adaptation in biomedical NER.

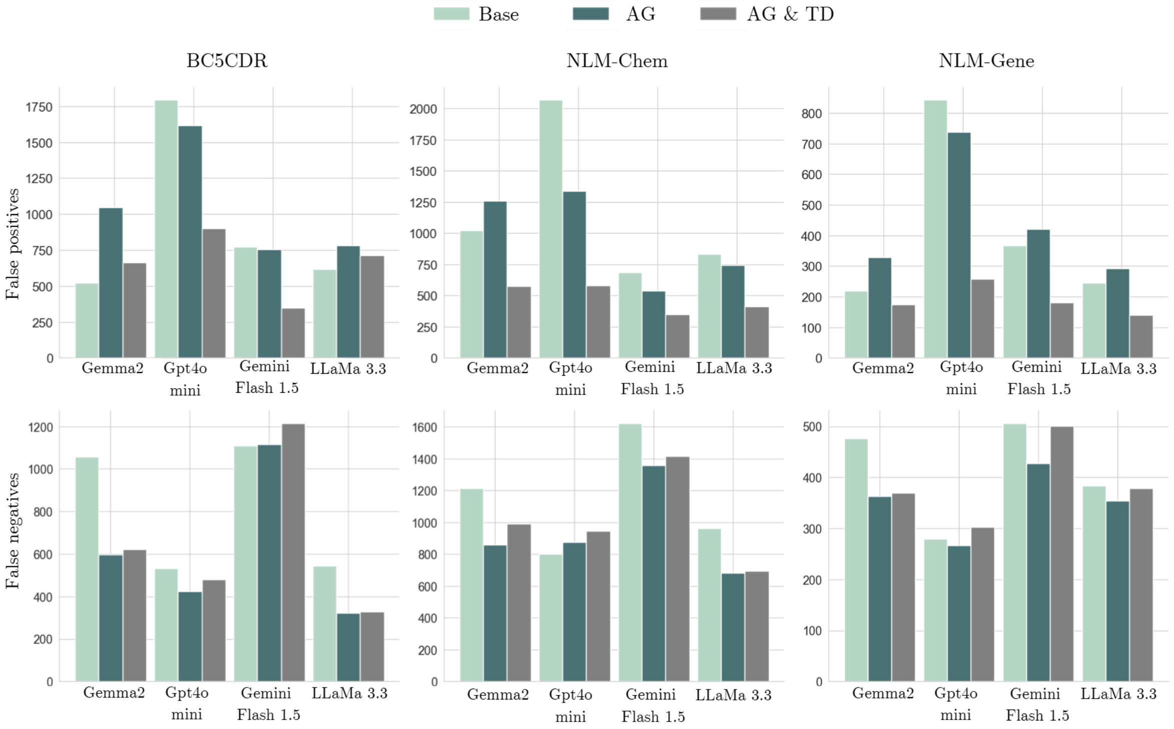

Figure 2: Distribution of false positives (top row) and false negatives (bottom row) in case of span detection alone without guidelines (Base), span detection incorporating annotation guidelines (AG); span detection with guidelines followed by type disambiguation (AG+TD).

5.4 Comparison of LLM-based verification methods

To illustrate the advantages of FETA's self-verification strategy, that employs in-line tagging for type disambiguation, we compare it against two existing prompt-based approaches that revise NER outputs in post hoc manner:

- **VerifiNER** (Kim et al., 2024) a multi-staged verification pipeline combining LLM prompting with external biomedical knowledge bases (e.g., UMLS) to refine NER predictions through knowledge-grounded reasoning.

- **GPT-NER** (Wang et al., 2023) employs few-shot prompting in a question-answering format to independently verify and filter predicted entity mentions.

Table 3 reports the performance of these methods on the BC5CDR and GENIA using exact match metrics. On both benchmarks, FETA achieves superior F1 scores while relying solely on the LLM's contextual reasoning, without additional dependance on external knowledge sources.

Beyond accuracy, a key contribution of FETA is its computational efficiency. As shown in Table 4, our verification strategy requires only a single LLM call per passage, verifying all candidate mentions jointly via in-line tagging. In contrast, both VerifiNER and GPT-NER verify each entity independently, resulting in prompt complexity that grows linearly with the number of predicted entities ($\mathcal{O}(n)$). A detailed analysis of computational efficiency is provided in Appendix C.

This efficiency gain becomes more pronounced as the number of predicted mentions per passage increases, highlighting the scalability advantages of our method. In practical scenarios involving entity-dense passages, FETA offers substantial reductions in inference cost, without sacrifying performance.

In summary, FETA delivers two key advantages over prior work:

1. It grounds verification in the original context through in-line tagging, enabling the LLM to disambiguate entity types without relying on external knowledge.

2. It verifies all predicted entities in a single prompt, resulting in constant-time inference with respect to the number of entity mentions.

These results highlight FETA as a simple, effective, and scalable alternative to expensive and complex multi-step verification pipelines.

	BC5CDR			GENIA		
Model	Precision	Recall	F1	Precision	Recall	F1
GPT-NER	79.8	47.5	59.5	56.4	42.1	48.3
VerifiNER	91.0	46.9	61.9	72.4	44.9	55.5
FETA	76.8	81.9	**79.2**	57.2	58.8	**58.0**

Table 3: Performance comparison of LLM-based verification methods on GENIA and BC5CDR datasets.

Model	LLM Calls per passage	Big-O notation
GPT-NER	n	$\mathcal{O}(n)$
VerifiNER	$[2n,\ 2\alpha n + 2n]$	$\mathcal{O}(n)$
FETA	1	$\mathcal{O}(1)$

Table 4: Comparison of efficiency across verification methods in terms of LLM call counts per passage.

6 Conclusions

In this work, we introduced FETA, a two-stage prompting framework that enables off-the-shelf LLMs to perform biomedical named entity recognition without domain-specific training. By decomposing the NER task into two subtasks, candidate span detection followed by type disambiguation via inline tagging, FETA effectively leverages the reasoning and generalization capabilities of generative LLMs through carefully crafted prompts.

Our experiments demonstrate that FETA outperforms existing LLM-based verification strategies and achieves superior performance compared to state-of-the-art supervised BioNER methods across multiple benchmarks. These findings highlight the potential of prompt-based methods for biomedical information extraction, even in the absence of dedicated training or external knowledge sources.

By requiring two prompt calls per text passage, FETA offers a scalable alternative to traditional fine-tuning or high-cost, multi-stage verification pipelines. This makes it especially appealing for real-world biomedical applications where labeled data is scarce, and rapid domain adaptation is crucial. Overall, this work provides strong evidence that carefully engineered prompt design can unlock the potential of general-purpose LLMs for complex biomedical NLP tasks, opening new avenues for low-resource applications in biomedical text mining.

7 Limitations

While FETA achieves strong performance without task-specific fine-tuning or reliance on biomedical knowledge bases, its effectiveness depends on the clarity and structure of prompts, which may require manual tuning, including the definition and structuring of annotation guidelines. Additionally, as any prompt-based approach, it remains sensitive to LLM output variability which may limit reproducibility in some settings.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David A. Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 1998–2022.

Dhananjay Ashok and Zachary C Lipton. 2023. Prompt-NER: Prompting for named entity recognition. *arXiv preprint arXiv:2305.15444*.

Reza Averly and Xia Ning. 2024. Entity Decomposition with Filtering: A Zero-Shot Clinical Named Entity Recognition Framework. *arXiv preprint arXiv:2407.04629*.

Junyi Bian, Jiaxuan Zheng, Yuyi Zhang, Hong Zhou, and Shanfeng Zhu. 2024. One-shot Biomedical Named Entity Recognition via Knowledge-Inspired Large Language Model. In *Proceedings of the 15th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, BCB 2024, Shenzhen, China, November 22-25, 2024*, pages 26:1–26:10. ACM.

Junyi Bian, Jiaxuan Zheng, Yuyi Zhang, and Shanfeng Zhu. 2023. Inspire the large language model by external knowledge on biomedical named entity recognition. *arXiv preprint arXiv:2309.12278*.

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

Qi Cheng, Liqiong Chen, Zhixing Hu, Juan Tang, Qiang Xu, and Binbin Ning. 2024. A novel prompting method for few-shot ner via llms. *Natural Language Processing Journal*, 8:100099.

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*.

Zihao Fu, Yixuan Su, Zaiqiao Meng, and Nigel Collier. 2023. Biomedical Named Entity Recognition via Dictionary-based Synonym Generalization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 14621–14635.

Z Gero, C Singh, H Cheng, T Naumann, M Galley, J Gao, and H Poon. 2023. Self-Verification Improves Few-Shot Clinical Information Extraction. arXiv 2023. *arXiv preprint arXiv:2306.00024*.

Jiaying Gong. 2024. *Few-Shot and Zero-Shot Learning for Information Extraction*. Ph.D. thesis, Virginia Tech, Blacksburg, VA, USA.

Yu Gu, Robert Tinn, Hao Cheng, Michael R. Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3:1 – 23.

Bernal Jimenez Gutierrez, 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, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 4497–4512.

Danqing Hu, Bing Liu, Xiaofeng Zhu, Xudong Lu, and Nan Wu. 2024a. Zero-shot information extraction from radiological reports using ChatGPT. *International Journal of Medical Informatics*, 183:105321.

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. 2024b. Improving large language models for clinical named entity recognition via prompt engineering. *J. Am. Medical Informatics Assoc.*, 31(9):1812–1820.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Rezarta Islamaj, Robert Leaman, David Cissel, Cathleen Coss, Joseph Denicola, Carol Fisher, Rob Guzman, Preeti Gokal Kochar, Nicholas Miliaras, Zoe Punske, et al. 2022. NLM-Chem-BC7: manually annotated full-text resources for chemical entity annotation and indexing in biomedical articles. *Database*, 2022(2022):baac102.

Rezarta Islamaj, Chih-Hsuan Wei, David Cissel, Nicholas Miliaras, Olga Printseva, Oleg Rodionov, Keiko Sekiya, Janice Ward, and Zhiyong Lu. 2021. NLM-Gene, a richly annotated gold standard dataset for gene entities that addresses ambiguity and multi-species gene recognition. *Journal of biomedical informatics*, 118:103779.

Vipina K Keloth, Yan Hu, Qianqian Xie, Xueqing Peng, Yan Wang, Andrew Zheng, Melih Selek, Kalpana Raja, Chih Hsuan Wei, Qiao Jin, et al. 2024. Advancing entity recognition in biomedicine via instruction tuning of large language models. *Bioinformatics*, 40(4):btae163.

Jin-Dong Kim, Tomoko Ohta, Yuka Tateisi, and Junichi Tsujii. 2003. Genia corpus - a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19 Suppl 1:i180–2.

Seoyeon Kim, Kwangwook Seo, Hyungjoo Chae, Jinyoung Yeo, and Dongha Lee. 2024. VerifiNER: Verification-augmented NER via Knowledge-grounded Reasoning with Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 2441–2461.

Jinhyuk Lee, WonJin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36:1234 – 1240.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Peng Li, Tianxiang Sun, Qiong Tang, Hang Yan, Yuanbin Wu, Xuanjing Huang, and Xipeng Qiu. 2023. CodeIE: Large Code Generation Models are Better Few-Shot Information Extractors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 15339–15353.

Ling Luo, Chih-Hsuan Wei, Po-Ting Lai, Robert Leaman, Qingyu Chen, and Zhiyong Lu. 2023. AIONER: all-in-one scheme-based biomedical named entity recognition using deep learning. *Bioinform.*, 39(5).

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, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11048–11064.

Masoud Monajatipoor, Jiaxin Yang, Joel Stremmel, Melika Emami, Fazlolah Mohaghegh, Mozhdeh Rouhsedaghat, and Kai-Wei Chang. 2024. LLMs in Biomedicine: A study on clinical Named Entity Recognition. *arXiv preprint arXiv:2404.07376*.

Milad Moradi, Kathrin Blagec, Florian Haberl, and Matthias Samwald. 2021. Gpt-3 models are poor few-shot learners in the biomedical domain. *arXiv preprint arXiv:2109.02555*.

Monica Munnangi, Sergey Feldman, Byron C. Wallace, Silvio Amir, Tom Hope, and Aakanksha Naik. 2024. On-the-fly Definition Augmentation of LLMs for Biomedical NER. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 3833–3854.

Aishik Nagar, Viktor Schlegel, Thanh-Tung Nguyen, Hao Li, Yuping Wu, Kuluhan Binici, and Stefan Winkler. 2024. LLMs are not Zero-Shot Reasoners for Biomedical Information Extraction. *arXiv preprint arXiv:2408.12249*.

Reid et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. GPT-NER:Named Entity Recognition via Large Language Models. *CoRR*, abs/2304.10428.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. ChatIE: Zero-shot information extraction via chatting with ChatGPT. arXiv. *arXiv preprint arXiv:2302.10205*.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical Study of Zero-Shot NER with ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 7935–7956.

Appendix

A Evaluation metrics

To ensure consistency with prior research, both exact and partial match scores were computed using publicly available evaluation scripts. Exact match requires that predicted entity spans exactly match the gold-standard spans in both boundary and type.

For partial match evaluation, a prediction is considered correct if it shares the same entity type as the gold span and has at least one overlapping token. This relaxed criterion accounts for minor boundary mismatches that are often acceptable in practice, particularly with LLM-generated outputs.

The partial match metrics are computed using the following formulas:

$$\text{Recall} = \frac{|\text{Predicted} \cap \text{Gold}|}{|\text{Gold}|}$$

$$\text{Precision} = \frac{|\text{Predicted} \cap \text{Gold}|}{|\text{Predicted}|}$$

Here, $|\text{Predicted} \cap \text{Gold}|$ denotes the number of predicted spans that have any boundary overlap with gold entity spans and match in type. The F1 score is computed as the harmonic mean of precision and recall.

B Implementation details

Following prior work, we adopt the same evaluation setup as in Keloth et al. (2024), where models are fine-tuned on a source dataset and evaluated on a target dataset with an identical set of biomedical types. This approach provides an accurate estimate of the generalization ability of a BioNER system and enables a fair comparison between off-the-shelf and fine-tuned models. Specifically, we use NCBI-Disease, BC5CDR-Chem, and BC2GM for training and BC5CDR-Disease, NLM-Chem, NLM-Gene for evaluation. BERT-based models are fine-tuned for 10 epochs with a batch size of 32, selecting the best-performing checkpoint based on validation set performance.

C Analysis of efficiency across prompt-based Verification methods

Quantifying efficiency across prompt-based verification methods is non-trivial due to the complexity of prompt strategies, especially for multi-step methods like VerifiNER. Below, we provide an approximation of the number of LLM calls per input passage for each approach discussed in Section 5.4, offering a comparative view of computational cost, outlined in Table 4.

VerifiNER. The number of LLM calls varies based on the number of predicted entity mention n and a hyperparameter α, which controls the number of substrings generated from the local context around each predicted mention. The factor of 2

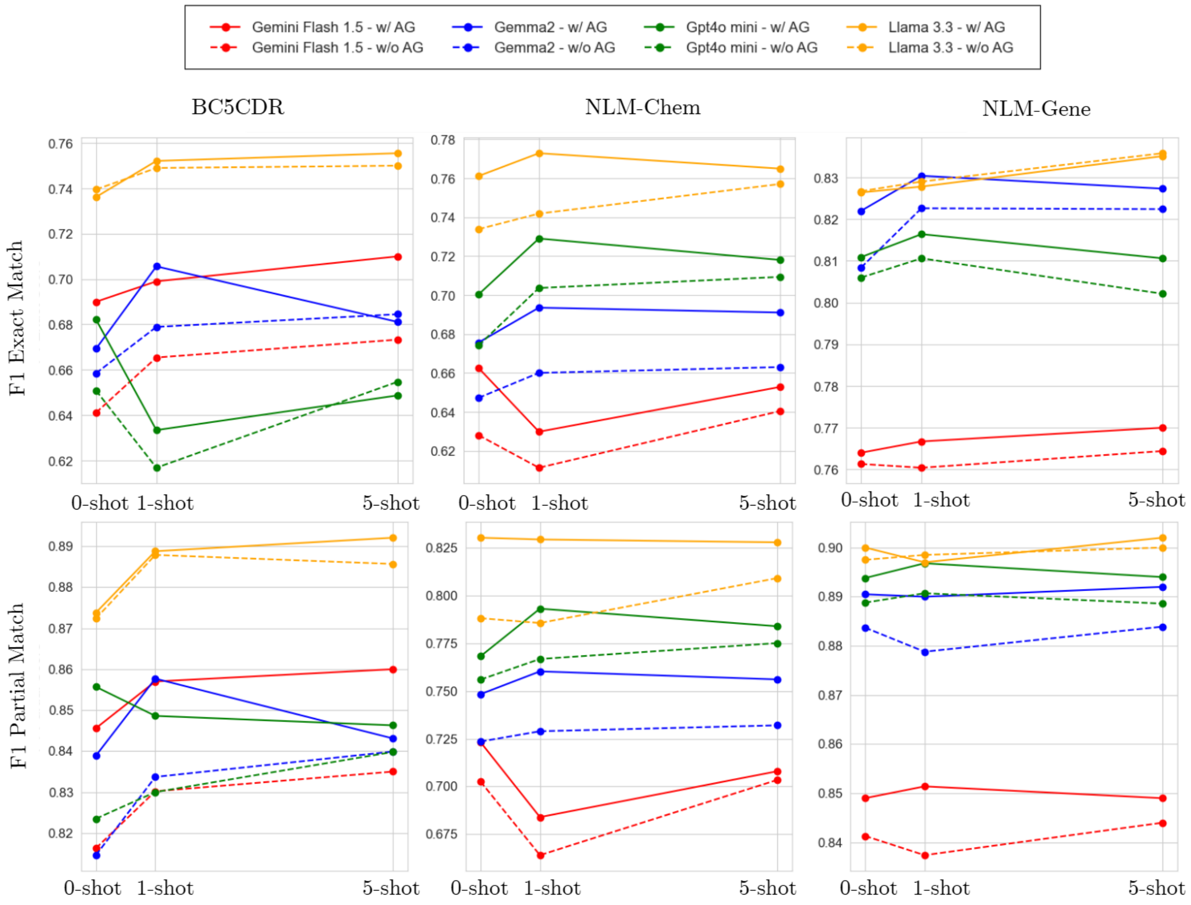

Figure 3: Performance comparison with N-shot examples with (solid line) and without (dashed line) annotation guidelines (AG) using exact (top row) and partial (bottom row) evaluation criteria.

accounts for symmetric expansion to the left and right of each mention. In addition to verifying the expanded mention set, an extra LLM call is issued for final consistency verification of each predicted entity. Thus, the *upper bound* can be estimated as:

$$\text{LLM Calls} = 2 \times \alpha \times n + 2n$$

And, the *lower bound*, assuming no neighborhood expansion (i.e., $\alpha = 0$), is:

$$\text{LLM Calls} = 2n$$

GPT-NER. GPT-NER requires one LLM call per each predicted entity span:

$$\text{LLM Calls} = n$$

FETA. Our method verifies all predicted spans jointly in a single prompt, using a constant number of LLM calls per passage.

$$\text{LLM Calls} = 1$$

Dataset	**Types**	**Guidelines**	**Documents**	**Mentions**
BC5CDR (Li et al., 2016)	Disease, Symptom, Chemical	yes	500	4363
NLM-Chem (Islamaj et al., 2022)	Chemical, Drug	yes	50	11514
NLM-Gene (Islamaj et al., 2021)	Gene, Protein	yes	100	2729
GENIA (Kim et al., 2003)	Cell line, Cell type, DNA, RNA, Protein	no	213	1854

Table 5: Dataset Statistics. The number of documents and mentions refers to the test partition used for evaluation.

Annotation Guidelines for NLM-Chem

1. Extract the shortest continuous span of text that identifies a chemical. A chemical is a substance, mixture, or a class of substances with a specific name and a well-defined structural or elemental composition. This may refer to:

- Chemical elements and atomic symbols (e.g., Sodium (Na), Hydrogen (H), Oxygen (O));
- Organic compounds (e.g., Acetone, Benzene, Vitamine C); Inorganic compounds (e.g., Water, Sodium Chloride, Sulfuric Acid);
- Lipids and Fatty acids (e.g., Polyunsaturated fatty acids, glycerolipid);
- Amino Acids (e.g., Serine (Ser), Arginine (Arg), Cysteine (Cys));
- Nucleotides (e.g., Adenosine Triphosphate, Cytidine Triphosphate);
- Nucleosides (e.g., Adenosine, Thymidine, Uridine); Carbohydrates (e.g., Glucose, Oligosaccharides); Minerals (e.g., Tals, Mica, Quartz);
- Synthetic polymers and manufactured materials with a well-defined compositions (e.g., Polyethylene glycol, Nylon, Polystyrene Sulfonate, Polyvinyl chloride, Polyamides, Nafion);
- Named laboratory reagents that have a defined chemical composition (e.g., fast blue conjugate dye, Laemmli buffer, amplex red reagent).

2. Extract specific drug names and substances that are used as drugs.

3. Extract systematic chemical names (e.g.m IUPAC-like names) and chemical formulas representing the molecular structure or composition of a compound.

4. Extract abbreviations and acronyms that refer to a chemical. If a chemical's name is followed by its abbreviation, extract both the chemical name and the abbreviation, treating each as a standalone chemical entity.

5. Extract chemical names of amino acids from residues and from mutations, omitting any numerical references to their position within proteins (e.g., "Ser238" -> "Ser", "Arg363Cys" -> "Arg; Cys"). Ignore the numerical part when extracting amino acid names.

6. Ignore the chemical term if it is mentioned as part of a protein, enzyme, or gene name.

7. Avoid extracting chemical bonds written as symbols or shorthand (i.e. H-bond, C-H, C=O, O-H, N-H etc.).

8. Ignore mentions of chemical functions or pharmaceutical actions like pesticides, analgesics, contraceptives, hypertensives, enzyme inhibitors.

9. Ignore the following general words: Atoms, Matter, Element, Molecule, Gene, Protein, DNA (as well as cDNA, ssDNA, etc.), Alkali, Moiety, Solutions, RNA (as well as mRNA, sniRNA, snoRNA, etc).

Table 6: Annotation Guidelines for NLM-Chem

Annotation Guidelines for BC5CDR-Disease

1. Extract the shortest continuous span of text that identifies a specific disease name or a disease class. This may refer to:

- Physical abnormality (e.g., Hernia, Fistula of thoracic duct);
- Mental or Behavioral condition (e.g., Schizophrenia, anxiety disorder, dementia);
- Disease or Syndrome (e.g., Toxicity, Acute pancreatitis, Rheumatoid Arthritis);
- Injury or poisoning (e.g., Contusion, laceration of cerebrum);
- Dysfunction and Deficiency (e.g., Uniparental disomy, Intestinal metaplasia);
- Symptom (e.g., Back Pain, Seizures, Skeletal muscle paralysis);
- Pathologic Function (e.g., Myocardial degeneration, Adipose Tissue Atrophy).

2. Extract abbreviations that refer to a disease name. If a disease name is followed by its abbreviation, extract both the disease name and the abbreviation, treating each as a standalone entity.

3. If multiple disease mentions appear in the same noun phrase, separated by a conjunction or punctuation, extract each as a separate entity.

4. Extract disease mentions used as syntactic modifiers of other concepts.

5. Avoid extracting the following very general words as standalone entities: Disease, Syndrome, Deficiency, Complications, Abnormalities. These may be extracted only within a specific disease mention.

6. Avoid extracting biological processes such as Tumorigenesis, Carcinogenesis. However, do extract all mentions of the following: Tumor, Cancer, Toxicity, Pain, Death.

Annotation Guidelines for NLM-Gene

1. Extract every span of text that identifies a gene or protein name, its synonym, gene or protein family, and other gene products, such as enzymes, receptors, transcription factors, cytokines, kinases, and protein-based hormones.

2. If multiple synonymous forms of the same gene/protein mention appear in the text (such as the full gene/protein name and its abbreviation), extract each as a separate span.

3. If multiple genes/proteins are mentioned in the same noun phrase, extract each one individually.

4. Avoid extracting the following very general words: Protein, Enzyme, Protein Isoforms, Isoenzymes, Micro RNA, Non-coding RNA, Organism Specific Proteins (e.g. Drosophila Proteins, Arabidopsis Proteins), and Location-Specific Proteins (e.g. Membrane Proteins, Nuclear Proteins, Blood Proteins). However, do extract all mentions of the following: Cytokines, Chemokines, Kinases, Caspases.

Table 7: Annotation Guidelines for BC5CDR-Disease and NLM-Gene

Task Description
Your task is to identify and extract all mentions of {types} in the text provided within triple backticks.

Output Format
Format the output as the following JSON object:
{"entities": "a semicolon-separated list of specified biomedical entities mentioned in the text." }

Example:
Text:
*```In the present study we explored whether environmental exposures to 1-naphthol (1N), a metabolite of carbaryl and naphthalene, is associated with decreased semen quality in humans.```*
Output:
{"entities": "1-naphthol; 1N; carbaryl; naphthalene"}

#Annotation Guidelines:
Please refer to the following annotation guidelines to correctly extract all mentions of {types} from the input text: {Annotation Guidelines}

#Text:
{Text}

Figure 4: Prompt Template for Instruction-based Span Detection

Task Description
Given a semicolon-separated list of entities, your task is to identify and tag all mentions of these entities in the text provided within triples backticks. For each provided entity, choose an HTML tag to reflect its high-level semancti category (e.g., disease, symptom, chemical, protein, cell, etc). For instance:

- Use <entity type="chemical"></entity> to tag chemicals.

Example:
Entities: 1-naphthol; 1N; carbaryl; humans
Text:
*```In the present study we explored whether environmental exposures to 1-naphthol (1N), a metabolite of carbaryl and naphthalene, is associated with decreased semen quality in humans.```*

Output:
*```In the present study we explored whether environmental exposures to <entity type="chemical">1-naphthol </entity> (<entity type="chemical">1N</entity>), a metabolite of <entity type="chemical">carbaryl</entity> and naphthalene, is associated with decreased semen quality in <entity type="organism">humans</entity>.```*

#Entities:
{Entities}

#Text:
{Text}

Figure 5: Prompt Template for Type Disambiguation

Task Description
Your task is to identify and extract all mentions of {types} in the text provided within triple backticks.

Output Format
Format the output as the following JSON object:
{"entities": "a semicolon-separated list of specified biomedical entities mentioned in the text." }

Example:
Text:
*```In the present study we explored whether environmental exposures to 1-naphthol (1N), a metabolite of carbaryl and naphthalene, is associated with decreased semen quality in humans.```*
Output:
{"entities": "1-naphthol; 1N; carbaryl; naphthalene"}

#Text:
{Text}

Figure 6: Prompt Template for Extract-Only Strategy

Task Description
Your task is to identify and tag all mentions of {types} entities in the text provided within triples backticks. For each provided entity, choose an HTML tag to reflect its high-level semancti category (e.g., disease, symptom, chemical, protein, cell, etc). For instance:

- Use <entity type="chemical"></entity> to tag chemicals.

Example:
Text:
*```In the present study we explored whether environmental exposures to 1-naphthol (1N), a metabolite of carbaryl and naphthalene, is associated with decreased semen quality in humans.```*

Output:
*```In the present study we explored whether environmental exposures to <entity type="chemical">1-naphthol </entity> (<entity type="chemical">1N</entity>), a metabolite of <entity type="chemical">carbaryl</entity> and naphthalene, is associated with decreased semen quality in <entity type="organism">humans</entity>.```*

#Text:
{Text}

Figure 7: Prompt Template for All-in-One Strategy