OpenBioNER: Lightweight Open-Domain Biomedical Named Entity Recognition Through Entity Type Description

Alessio Cocchieri[♠*] Giacomo Frisoni[♠*] Marcos Martínez Galindo[♢]
Gianluca Moro[♠*] Giuseppe Tagliavini[♠] Francesco Candoli[♠]
[♠]Department of Computer Science and Engineering, University of Bologna
[♢]IBM Research Europe, Dublin
[♠]{a.cocchieri, giacomo.frisoni, gianluca.moro, giuseppe.tagliavini}@unibo.it
[♠]francesco.candoli@studio.unibo.it [♢]marcos.martinez.galindo@ibm.com

Abstract

Biomedical Named Entity Recognition (BioNER) faces significant challenges in real-world applications due to limited annotated data and the constant emergence of new entity types, making zero-shot learning capabilities crucial. While Large Language Models (LLMs) possess extensive domain knowledge necessary for specialized fields like biomedicine, their computational costs often make them impractical. To address these challenges, we introduce OpenBioNER, a lightweight BERT-based cross-encoder architecture that can identify any biomedical entity using only its description, eliminating the need for retraining on new, unseen entity types. Through comprehensive evaluation on established biomedical benchmarks, we demonstrate that OpenBioNER surpasses state-of-the-art baselines, including specialized 7B NER LLMs and GPT-4o, achieving up to 10% higher F1 scores while using 110M parameters only. Moreover, OpenBioNER outperforms existing small-scale models that match textual spans with entity types rather than descriptions, both in terms of accuracy and computational efficiency.[1]

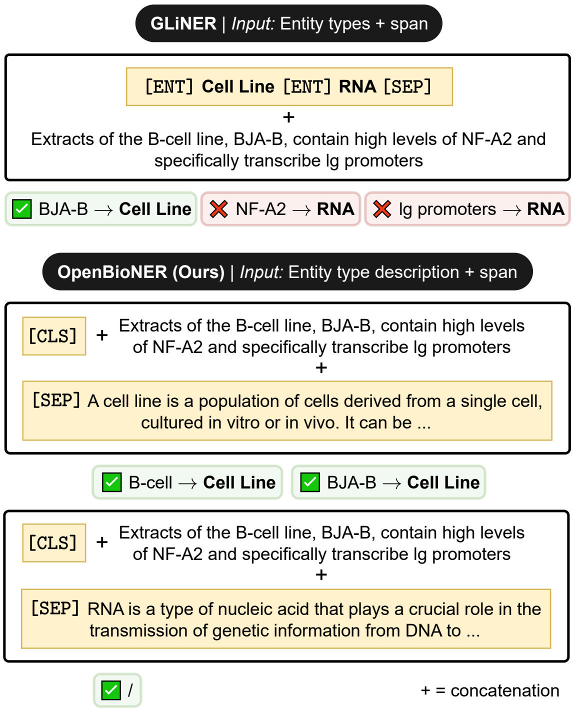
GLiNER | Input: Entity types + span
[ENT] Cell Line [ENT] RNA [SEP]
+
Extracts of the B-cell line, BJA-B, contain high levels of NF-A2 and specifically transcribe Ig promoters
✓ BJA-B → Cell Line
✗ NF-A2 → RNA
✗ Ig promoters → RNA
OpenBioNER (Ours) | Input: Entity type description + span
[CLS] + Extracts of the B-cell line, BJA-B, contain high levels of NF-A2 and specifically transcribe Ig promoters
+
[SEP] A cell line is a population of cells derived from a single cell, cultured in vitro or in vivo. It can be ...
✓ B-cell → Cell Line
✓ BJA-B → Cell Line
[CLS] + Extracts of the B-cell line, BJA-B, contain high levels of NF-A2 and specifically transcribe Ig promoters
+
[SEP] RNA is a type of nucleic acid that plays a crucial role in the transmission of genetic information from DNA to ...
✓ /
+ = concatenation

Figure 1: **Comparison between small language models for open-domain biomedical Named Entity Recognition.** GLiNER (Zaratiana et al., 2024) matches entity types with text spans in a latent space. Our proposed OpenBioNER classifies tokens by injecting the description of one target entity at a time via cross-attention. In the example, the broader and less ambiguous context provided by class descriptions allows OpenBioNER to prevent RNA false positives.

1 Introduction

Biomedical Named Entity Recognition (BioNER) focuses on identifying and extracting biomedical entities from text. These entities often include the names of proteins, genes, and their respective locations of activity, such as specific cell types or organism names. The extraction of these entities is fundamental to the advancement of gene-disease association studies, drug discovery, personalized medicine, document classification, and the construction of knowledge graphs (Wei et al., 2012; Domeniconi et al., 2014, 2016; Alshahrani et al., 2017; Frisoni et al., 2022b, 2023).

The development of effective BioNER systems poses significant challenges due to the syntactic and semantic complexities of the domain (Nayel et al., 2019). Biomedicine is characterized by specialized jargon, synonyms, hierarchical relationships between entities, ambiguous abbreviations, and polysemy (e.g., "EGFR" can refer to both the epidermal growth factor receptor and the estimated glomerular filtration rate). Creating annotated examples

[*] Equal contribution (co-first authorship).
[1] Our code, data, and fine-tuned models are publicly available at disi-unibo-nlp/openbioner

Findings of the Association for Computational Linguistics: NAACL 2025, pages 818–837
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

for these entities is labor-intensive and requires expert medical annotators. The task becomes even more complicated due to the constant emergence of new entity types. Since 2011, more than 1 million new records have been added to the PubMed database yearly, translating to approximately three articles published every minute (Frisoni et al., 2021). This rapid expansion highlights the necessity of zero-shot learning (Xian et al., 2019; Wang et al., 2019a), where models must adapt to entities that were not seen during training. Recent studies have investigated zero-shot information extraction systems (Wang et al., 2023; Zhou et al., 2024; Zaratiana et al., 2024). However, the existing methodologies often prioritize general scenarios, rely on costly large language models (LLMs), and do not fully exploit target entity type descriptions for context-based classification.

To overcome these limitations, we introduce OPENBIONER (Figure 1), a lightweight BERT-based model tailored for *open-domain* BioNER. This model can find unseen target entity types based solely on their natural language descriptions, eliminating the need for retraining. OPENBIONER is pretrained on synthetic silver annotations generated through LLM self-supervision, drawing inspiration from recent advancements in the medical field (Agrawal et al., 2022; Gu et al., 2023). Extensive experiments demonstrate that OPENBIONER outperforms specialized LLMs, such as UniversalNER (UniNER) (Zhou et al., 2024) and GPT-4o (OpenAI, 2023), achieving an F1 score improvement of up to 10% in zero-shot settings across various biomedical benchmarks. In comparison to smaller baselines such as GLiNER (Zaratiana et al., 2024), our model achieves better performance while using up to 4× fewer parameters.

Our contributions can be summarized as follows:

- **Using descriptions for improved token classification.** We demonstrate that leveraging robust, synthetic descriptions—rather than simple class names—significantly enhances the model's generalization ability, particularly in complex biomedical scenarios where entity types can be highly diverse and nuanced.

- **Open-domain adaptation.** Our method incorporates progressive pre-training on thousands of entity type descriptions, equipping a BERT-based model to perform effectively in open-domain settings. This enables classification of any biomedical entity type through informed, context-aware descriptions, rather than relying solely on predefined class names.

- **State-of-the-art results.** OPENBIONER outperforms GLiNER, UniNER, GPT-4o, and other baselines across multiple benchmarks, highlighting its superior effectiveness and practical value in zero-shot configuration.

- **Dataset and model release.** To support research and real-world use, we release OPENBIONER under a permissive license, along with its pre-training data and the code for training and evaluation. This enables cost-effective applications in medical domains, providing a viable alternative to LLMs and API services.

2 Related Work

Biomedical Named Entity Recogniton BioNER has experienced significant advancements throughout the years. Early efforts relied heavily on rule-based approaches, such as MetaMap (Aronson, 2001), MetaMapLite (Demner-Fushman et al., 2017), and cTAKES (Savova et al., 2010). These systems utilized extensive dictionaries and handcrafted rules to identify static biomedical entities, lacking generalization and consideration of contextual nuances (Zhang et al., 2020). The research landscape shifted dramatically with the advent of neural models. Initial architectures were centred on recurrent neural networks and conditional random fields (Habibi et al., 2017; Giorgi and Bader, 2018; Wang et al., 2019b; Yoon et al., 2019). The introduction of BERT (Devlin et al., 2019) marked a turning point, leveraging transformer-based architectures, pretraining, and bidirectional contextual embeddings to enhance NER performance. Biomedical-specific adaptations, including BioBERT (Lee et al., 2020), ClinicalBERT (Huang et al., 2019), BlueBERT (Peng et al., 2019), SciBERT (Beltagy et al., 2019), and PubMedBERT (Gu et al., 2022), are the modern, supervised, encoder-only approaches. To support a broader spectrum of entity types, contemporary encoders standardize heterogeneous training datasets into a single format (Luo et al., 2023). Moving away from a sequence labeling conceptualization of the BioNER task, recent works have explored generative LLMs both in terms of few-shot prompting (Monajatipoor et al., 2024) and instruction fine-tuning (Keloth et al., 2024). Despite their potential, domain-specific small language models with a few million parameters better balance performance and efficiency, requiring less computational resources while being faster and easier to train.

Open-Domain Named Entity Recognition Our work is a biomedical-specific synthesis of two previously distinct NER research paths: (1) the provision of supplementary details about entity types via descriptions, and (2) the exploitation of LLMs' generalization capabilities.

In the first path, Aly et al. (2021) demonstrated the effectiveness of cross-encoder architectures in handling unseen targets by augmenting the input with manually-defined entity type descriptions. Concurrently, other studies validated the use of descriptions in zero-shot settings for related tasks such as entity typing (Obeidat et al., 2019), relation extraction (Chen and Li, 2021), and entity linking (Logeswaran et al., 2019; Wu et al., 2020).

The second path has been marked by the rise of generative LLMs, enabling entity identification via natural language instructions that specify types of interest. This strategy capitalizes on vast prior knowledge and introduces support for arbitrarily varied target classes. Notable contributions include InstructUIE (Wang et al., 2023), which instruction-tuned FlanT5 on over 30 information extraction datasets to capture inter-task dependencies, excelling in zero-shot NER. GoLLIE (Sainz et al., 2024) improved zero-shot results on unseen information extraction tasks by fine-tuning CodeLLaMA to comply with code-style annotation guidelines. GNER (Ding et al., 2024) enhanced performance by integrating negative instances into the instruction tuning process. Due to the lack of gold labels, inconsistencies in datasets (Yang et al., 2024), and LLM costs, open-domain NER has shifted towards knowledge distillation through self-supervision. In this paradigm, models like ChatGPT annotate raw data, creating silver training instances used to fine-tune smaller models. UniNER (Zhou et al., 2024) pioneered this approach by fine-tuning LLaMA-7B on Pile-NER, a ChatGPT-supervised dataset. Its multi-round approach crafts different inquiry prompts for each entity type, though this method results in low inference speed at billion-scale. SeqGPT (Yu et al., 2024) applied a similar strategy for more general single-round extraction and classification tasks. The authors pretrained BLOOMZ with an extremely diverse label set generated by ChatGPT, then fine-tuning on high-quality gold datasets.

Recently, lightweight models have gained traction. GLiNER (Zaratiana et al., 2024) pushed state-of-the-art zero-shot NER by training a DeBERTa cross-encoder to match entity types and text spans using Pile-NER. NuNER (Bogdanov et al., 2024) pretrained RoBERTa on ≈4 million ChatGPT annotations, equaling the performance of much larger LLMs in few-shot scenarios.

3 Preliminary

Entity type descriptions in biomedicine provide critical semantic depth, helping models distinguish and understand entities better. Recent studies have highlighted several challenges in NER tasks that can be addressed through detailed descriptions (Zhou et al., 2024; Sainz et al., 2024; Yang et al., 2024). (1) **Concept vs. Named Entity Ambiguity:** Abstract concepts such as "modeling" in "modeling nurse-patient assignments" can be classified as Research Activity. Without descriptive context, models struggle to draw associations that deviate from typical expectations of named entities. (2) **Entity Scope Variations:** Datasets frequently differ in their definitions of entities. For example, "aspirin" might be labeled as Drug in one dataset but as Chemical in another. Precise descriptions can clarify these nuances, improving model accuracy across diverse annotation schemes. (3) **Granularity Overlap:** Annotation schemes can include categories with varying levels of specificity, leading to potential ambiguity. For example, a dataset might comprise broad entity types like Medication and more specific ones like Antibiotic. The entity "amoxicillin" could be correctly labeled as either. Without guidelines, models may struggle to consistently choose between these overlapping categories, defaulting to overgeneralization (Sainz et al., 2024). (4) **Span Inconsistencies and Synonym Recognition:** How entity mentions are selected often varies across and within datasets. For instance, one annotator might label the phrase "phosphorylated p53 protein" as a Protein entity, while another may only mark "p53." Indeed, annotation could focus on official symbols, full names, alias (e.g., "tumor protein p53"), or pronominal references (e.g., "it"). Although BioNER models may readily identify common terms (e.g., "bacteria"), they can struggle with less frequent synonyms (e.g., "microbes") or strain designations (e.g., "Escherichia coli O157:H7"). Descriptions can enhance robustness to lexical variations and domain-specific terms.

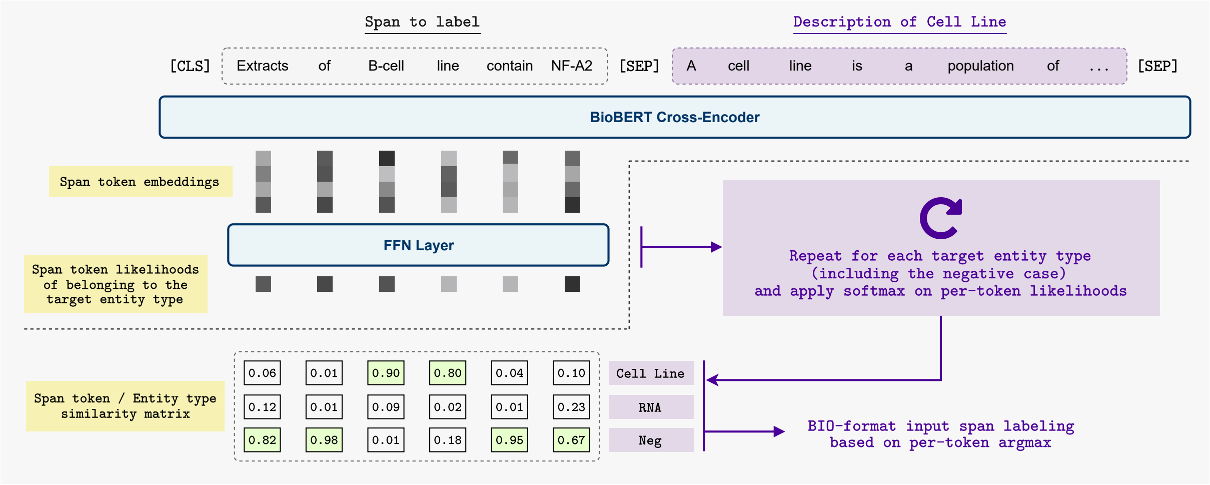
Span to label
Description of Cell Line
[CLS]
Extracts
of
B-cell
line
contain
NF-A2
[SEP]
A
cell
line
is
a
population
of
...
[SEP]
BioBERT Cross-Encoder
Span token embeddings
FFN Layer
Span token likelihoods of belonging to the target entity type
Repeat for each target entity type (including the negative case) and apply softmax on per-token likelihoods
Span token / Entity type similarity matrix
0.06 0.01 0.90 0.80 0.04 0.10
0.12 0.01 0.09 0.02 0.01 0.23
0.82 0.98 0.01 0.18 0.95 0.67
Cell Line
RNA
Neg
BIO-format input span labeling based on per-token argmax

Figure 2: **OpenBioNER architecture**. The text span and the description of a target entity type are given in input. BioBERT outputs a representation for each span token, contextualized on the description. Embeddings are fed into a linear layer, calculating a matching score between each span token and the description. This process iterates for each target entity type, substituting the input description. A softmax function is then applied to calculate the probability distribution of each token across all target entity types, including the negative case (simplified). Finally, each token is assigned to the most probable entity type.

4 Method

This section presents our model, OpenBioNER, trained to extract any type of biomedical entity from a text using its description. We discuss the underlying cross-encoder architecture and our contributions in adapting it to an open-domain setting.

Task Formalization We formulate the BioNER task as a token-level multi-class classification problem. Given a text span $\mathbf{s} = \{t_1, \ldots, t_n\}$ comprising n tokens and a description $\mathbf{d}_c$ for each target entity class $c \in \mathbb{C}$, we predict a sequence of annotations $\hat{\mathbf{y}} \in (\mathbb{C})^n$. For each token t_i, we determine its class by computing $\arg\max_{c \in \mathbb{C}} F(\mathbf{s}, t_i, \mathbf{d}_c)$, where F is a function that models the semantic affinaty between t_i and $\mathbf{d}_c$ in the context of $\mathbf{s}$.

4.1 Architecture

The overall architecture is depicted in Figure 2.

Cross-Encoder We realize F through the cross-encoder architecture introduced by Aly et al. (2021), which utilizes cross-attention to quantify span–description affinity. This design choice is motivated by the need for fine-grained token-level interactions. Compared to bi-encoders, cross-encoders allow the computation of *description-sensitive* representations for the input span tokens, potentially leading to superior annotation performance. Deviating from Aly et al. (2021), we replace the original BERT backbone with BioBERT (Lee et al., 2020). For ease of notation, we hereafter refer to the cross-encoder as X-ENC.

Input Format X-ENC receives the input tuple $(\mathbf{s}, \mathbf{d}_c)$ structured as a unified sequence:

$$\texttt{[CLS]}\ \mathbf{s}\ \texttt{[SEP]}\ \mathbf{d}_c\ \texttt{[SEP]}$$

Bidirectional attention enables rich interactions between s and $\mathbf{d_c}$. Due to context window constraints, inputs are processed into mini-batches. Specifically, for each text span to classify s, a mini-batch of size $|\mathbb{C}|$ is created by pairing $\mathbf{s}$ with all possible entity type descriptions. The maximum length for each entry in a mini-batch is set to 512.

Span Token Representation Given an entity type description d_c, X-ENC computes a vector representation $\mathbf{v}_{t_i} \in \mathbb{R}^h$ for every token t_i in $\mathbf{s}$:

$$\mathbf{v}_{t_1}, \ldots, \mathbf{v}_{t_{|\mathbf{s}|}} = \text{X-ENC}(\mathbf{s}, \mathbf{d}_c)$$

Here, $h = 768$ denotes the hidden size of the model. A learnable linear layer $\mathbf{W} \in \mathbb{R}^{h \times 1}$ projects each token vector $\mathbf{v}_{t_i}$ to a scalar score $\text{sim}(t_i, \mathbf{d}_c) \in \mathbb{R}$:

$$\mathbf{v}_{t_i, \mathbf{d}_c} \cdot \mathbf{W} = \text{sim}(t_i, \mathbf{d}_c) = \text{sim}(t_i, c)$$

This score indicates the likelihood of token t_i belonging to entity class c. To account for non-entity tokens, we introduce a negative class c_{neg}. Details on computing $\text{sim}(t_i, c_{neg})$ can be found in §A. The likelihood scores for each token are concatenated in a single vector $\mathbf{l}_{t_i} \in \mathbb{R}^{|\mathbb{C}+1|}$:

$$\mathbf{l}_{t_i} = \big(\text{sim}(t_i, c_1); \ldots; \text{sim}(t_i, c_{|\mathbb{C}|}); \text{sim}(t_i, c_{neg})\big)$$

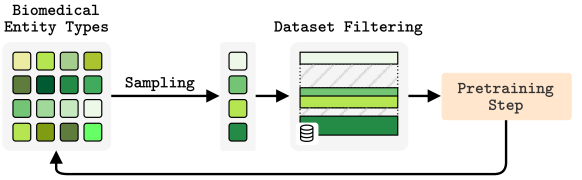

Figure 3: Iterative entity type sampling during pretraining. Each step selects a random assortment of target entity types and filters the synthetic dataset accordingly.

We then label each token (BIO format) with the most probable class after applying softmax:

$$\hat{y}_{t_i} = \arg\max_{c \in \mathbb{C}} F(\mathbf{s}, t_i, \mathbf{d}_c) = \arg\max_{c \in \mathbb{C}} \frac{\mathbf{l}_{t_i,c}}{\sum_{c' \in \mathbb{C}} \mathbf{l}_{t_i,c'}}$$

4.2 Training

The training recipe used for OPENBIONER extends the Aly et al. (2021) procedure with pretraining on an LLM-generated dataset.

Entity Masking Regularizer We use a regularization technique to enhance model robustness and prevent overfitting. This method, inspired by BERT's masked language modeling, involves probabilistic entity concealment during training. With probability p, we mask the entire target entity in the input span. This technique mitigates lexical bias and encourages the model to infer entity types from type descriptions. Moreover, it is particularly beneficial in biomedicine, where entity names can be highly variable and context-dependent, helping the model generalize beyond specific nomenclature to a broader conceptual understanding.

Class Imbalance-aware Loss We address the label imbalance caused by c_{neg} by incorporating class weights w_c into the cross-entropy loss:

$$\mathcal{L} = -\sum_{c}^{\mathbb{C}} w_c \cdot y_{i,c} \cdot \log(p(\hat{y}_{i,c}))$$

where $y_{i,c}$ represents the ground truth label and $p(\hat{y}_{i,c})$ is the softmax probability for class c. We consistently set $w_c = 1$ for all classes except the negative one, that we define as hyperparameter.

Pretraining with Progressive Entity Types Exposure Before fine-tuning, we pretrain OPENBIONER on a synthetic dataset (see 5.1). We follow the previously described training procedure but incorporate a progressive exposure strategy to manage the huge variety of labels that an LLM could generate (Figure 3). In particular, we pretrain the model in multiple iterations, with each pass focusing on a randomly selected subset of 15 to 25 targets. Dataset instances that do not contain any extracted entity types are filtered out for that step. Tokens in the input span associated with entity types not included in the current subset are labeled as 'O'. This incremental approach continues until all entity types have been covered at least once. Gradual exposure benefits biomedicine, where entity types can range from molecular structures to complex disease phenotypes. It allows the model to iteratively refine its understanding of various biomedical concepts while avoiding being overwhelmed by many targets presented simultaneously.

5 Experimental setup

5.1 Pretraining Data

The development of OPENBIONER requires a pretraining dataset that encompasses a broad spectrum of biomedical entity types and their corresponding descriptions. We used Pile-NER[2] for a fair comparison with the baselines, ensuring consistency in the biomedical entities observed. Pile-NER, built by Zhou et al. (2024), is derived from the Pile corpus (Gao et al., 2021) and comprises 50K articles segmented into 256-token passages; entity types are extracted using ChatGPT without predefined categories. We modify this dataset by (1) converting it to the BIO-tagging scheme (Huang et al., 2015) and (2) segmenting passages into sentences.[3] We filter for biomedical content using LLaMA-3.1-8B-instruct to perform binary topic classification on each sentence (see §G), a method inspired by (Xu et al., 2024). This refinement yields **Pile-NER-biomed**, a subset containing 59K instances, 193,235 entities, and 3,896 distinct entity types.

BIO Data Format Given the small proportion of consecutive entities belonging to the same class (1.22% pretraining data), we remove all the I- and B- prefixes. This approach may introduce ambiguity for consecutively named entities of the same class. However, given the size and heterogeneity of the dataset, the model is expected to learn contextual differentiation, as these entities likely appear in varied positions across multiple sentences.

[2] Universal-NER/Pile-NER-type
[3] We used NLTK's PunktSentenceTokenizer.

Dataset	Entity Types	#train	#dev	#test	Tok†	Ent‡
AnatEM (Pyysalo and Ananiadou, 2014)	Anatomy	5861	2118	3830	37	0.7
BC2GM (Smith et al., 2008)	Gene	12500	2500	5000	36	0.4
BC4CHEMD (Krallinger et al., 2015)	Chemical	30682	30639	26364	45	0.9
BC5CDR (Li et al., 2016)	Chemical, Disease	4560	4581	4797	41	2.2
NCBI (Dogan et al., 2014)	Disease	5432	923	940	39	1.0
JNLPBA (Collier and Kim, 2004)	Protein, DNA, RNA, Cell Line, Cell Type	18608	1940	4261	39	2.8
JNLPBA-*Rare*	RNA, Cell Type	-	-	465	50	1.3
MedMentions-*Rare*	Bacterium, Food, Professional, Body Substance, Body System	-	-	1048	39	1.4

† Average number of tokens per sentence (WordPiece vocabulary). ‡ Average number of entities per sentence.

Table 1: **Statistics of the datasets used for OpenBioNER evaluation.** Popular literature datasets for zero-shot and supervised comparisons (top). Custom datasets for zero-shot comparisons on rare entities (bottom).

5.2 Descriptions

Entity type descriptions are typically found in resources like Wikipedia, annotation guidelines, and domain-specific knowledge bases, including UMLS Metathesaurus. However, these descriptions may not always align with task-specific needs. To address this, we propose a two-step approach using synthetic descriptions generated by LLaMA-3.1-8B-instruct. In the first step, we *train* the model to develop a broad understanding of entities across domains, focusing on relationships between input span tokens and description tokens, even when descriptions are less specific. We prompt the LLM to generate concise, general descriptions of entity types and their applications across heterogeneous fields, then concatenated. The second step applies task-specific descriptions at *inference* time. We adopt few-shot prompting, supplying the LLM with contextual examples of entity usage. For consistency, we extract five sentence annotations per entity from each dataset's train set. See §G for prompt templates. In practical scenarios where a train set may not always be available, we assume users can manually or artificially define a small set of examples to reflect the attributes of each entity type.

5.3 Evaluation Setting

Baselines We compare our model against several baselines in two distinct configurations: **zero-shot** and **supervised**. In the zero-shot setting, we directly assess our model, pretrained on Pile-NER-biomed, on each benchmark's test set. In the supervised setting, we fine-tune it on the respective dataset train sets. For zero-shot comparisons, we include UniNER-7B[4] and GLiNER-large,[5] both trained on Pile-NER. We also evaluate GPT-4o, following Zhou et al. (2024) and using the prompt template suggested by Ye et al. (2023). For UniNER, we adhere to the template with which it was trained. For supervised comparisons, we prioritize non-description-based models. These include our backbone BioBERT-base[6] and a general-purpose BERT-base (cased). Furthermore, we incorporate GLiNER, UniNER, and InstructUIE.[7] Although these models are trained on 20 datasets across domains, our attention is on biomedicine.

Benchmarks We evaluate our model and baselines on two sets of benchmarks, as detailed in Table 1. The first set comprises standard BioNER datasets widely used in recent literature, allowing for head-to-head comparison in both supervised and zero-shot settings. The second set, denoted with the suffix "-Rare", is designated to judge BioNER zero-shot performance on infrequent entity types, mimicking real-world scenarios where annotations are scarce. Following Aly et al. (2021), we create these benchmarks by isolating the least frequent classes within each test set. See §B for more information.

Metric We apply strict entity-level micro-F1, which requires an exact match of both entity type and boundaries with the ground truth. This metric is consistent with the evaluation protocol of the baselines considered (Wang et al., 2023; Zhou et al., 2024; Zaratiana et al., 2024). Entity-level scores are computed using Seqeval.[8]

[4] Universal-NER/UniNER-7B-type

[5] urchade/gliner_large-v1

[6] dmis-lab/biobert-base-cased-v1.1

[7] BeyonderXX/InstructUIE

[8] https://github.com/chakki-works/seqeval (v1.2.2)

		Datasets								
Model	**Size**	**AnatEM**	**NCBI**	**JNLPBA**	**BC2GM**	**BC4CHEMD**	**BC5CDR**	**JNLPBA-R**	**MedMentions-R**	**AVG**
GPT-4o	-	**38.7**	50.0	41.9	37.3	36.4	66.4	26.6	49.1	43.3
UniNER†	7B	25.1	60.4	48.1	46.2	47.9	**68.0**	50.2	**53.4**	49.9
GLiNER-large†	459M	33.3	**61.9**	**57.1**	47.9	43.1	66.4	51.9	**53.4**	51.9
OpenBioNER *(Ours)*	110M	35.2	58.5	**57.1**	**49.1**	**48.0**	60.4	**63.9**	50.9	**52.9**

† Results taken from Zaratiana et al. (2024) except for JNLPBA, JNLPBA-R and MedMentions-R–which we compute using threshold = 0.5.

Table 2: **Zero-shot performance on BioNER datasets (test sets).** Sorted by average micro-F1 (ascending order).

Model	**Size**	**AnateEM**	**NCBI**	**BC2GM**	**BC4CHEMD**	**BC5CDR**	**JNLPBA**
GLiNER-large†	459M	**88.9**	**87.8**	**83.7**	87.9	88.7	-
UniNER†	7B	88.5	87.0	82.4	**89.2**	**89.3**	-
InstructUIE	11B	88.5	86.2	80.7	87.6	89.0	-
BERT-base	110M	85.3	84.0	78.5	84.0	84.7	72.7
BioBERT-base	110M	**87.5**	85.8	**82.4**	88.4	**88.5**	72.4
OpenBioNER *(Ours)*	110M	87.0	**86.1**	80.5	**88.5**	86.3	**74.3**

† Results taken from Zaratiana et al. (2024) except for JNLPBA, JNLPBA-R and MedMentions-R–which we compute using threshold = 0.5.

Table 3: **Supervised in-domain performance on BioNER datasets (test sets).** InstructUIE, UniNER, and GLiNER-large are provided as a reference only, as these models were trained on a mixture of 20 NER datasets.

Hardware All experiments are conducted on a single NVIDIA A100 GPU with 80GB of VRAM.

5.4 Implementation Details

Pretraining We pretrained OpenBioNER on the Pile-NER-biomed dataset for 4 epochs, processing entity types in subsets of 15–25 per epoch. We employed a batch size of 8 and a constant learning rate of 2e-5, the latter selected through hyperparameter tuning. Following Aly et al. (2021), we set the entity masking probability to 0.3. The negative class weight was dynamically calculated as $^{\#\text{ entities}}/_{\#\text{ non-entity words}}$ within each processed batch. We defined the maximum input sequence length to 300 tokens and the maximum description length to 150 tokens. Each epoch required approximately 10 hours, resulting in a total training time of 40 hours.

Fine-tuning For supervised evaluation, OpenBioNER, BioBERT-base and BERT-base were trained on each benchmark for 8 epochs, retaining the best-performing checkpoint on the validation set. In this scenario, OpenBioNER utilized the same domain-specific entity descriptions for both training and inference. To mitigate overfitting, we increased the entity masking probability to 0.5 and set a fixed negative class weight of 0.5 or 1.0 (see §D). Other hyperparameters remained invariant with the Pile-NER-biomed training configuration. For BioBERT and BERT, we applied a learning rate of 2e-4, a cosine decay scheduler, and a warmup ratio of 0.1. The batch size was set to 64 for all datasets except JNLPBA, where we used a batch size of 32 due to memory constraints.

6 Results

6.1 Zero-shot Evaluation

In this section, we discuss the performance of our model in a zero-shot setting, i.e., by only training on Pile-NER-biomed without fine-tuning on target datasets. The results are reported in Table 2. OpenBioNER demonstrates remarkable generalization capabilities, outperforming all baselines on the JNLPBA, BC2GM, and BC4CHEMD datasets, despite its compact size. Notably, it exhibits strong performance in recognizing rare classes, as evidenced by its results on JNLPBA-*Rare*. In addition, it achieves the highest overall score when considering average performance on all benchmarks. Similar to what Zhou et al. (2024) observed for ChatGPT, GPT-4o struggles to compete with smaller, specialized NER models, achieving success only on AnatEM. UniNER, despite its large parameter count, leads on BC5CDR and MedMentions-*Rare* but falls behind smaller encoder models on other benchmarks. Its performance is particularly challenged on AnatEM and JNLPBA, possibly due to ambiguities in entity type names when descriptions are not provided. GLiNER-large shows robust domain-specific performance across both standard and rare class identification tasks, securing the second-highest overall score. However, its effectiveness is limited in JNLPBA-*Rare*, suggesting sensitivity to class name ambiguity (see §C).

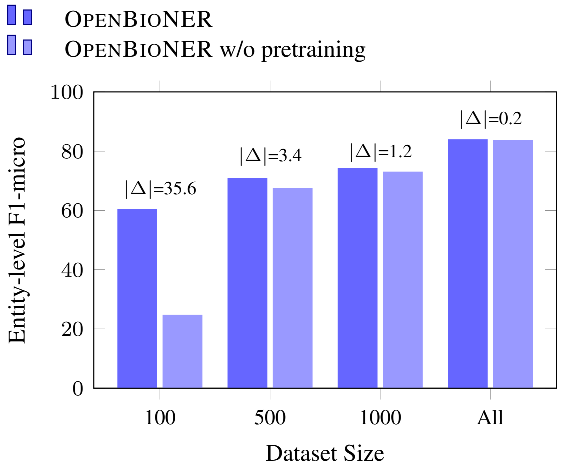

Figure 4: **Average supervised performance across varying dataset sizes**. Evaluated on AnatEM, BC2GM, BC4CHEMD, BC5CDR, and NCBI.

MedMentions-ZS	SMXM		OpenBioNER	
Class	UMLS	Ours	UMLS	Ours
Bacterium	0.29	**0.31**	0.30	**0.59**
Body Substance	**0.33**	0.10	0.19	**0.45**
Body System	0.10	**0.19**	0.16	**0.30**
Food	0.19	**0.41**	0.38	**0.47**
Prof. or Occ. Group	0.26	**0.42**	0.23	**0.52**

Table 4: **Class-based F1 scores of SMXM and OpenBioNER zero-shot performance with different description sources.** Comparison between UMLS-based descriptions and task-specific LLM-generated descriptions (Ours) on MedMentions-*Rare*.

6.2 Supervised Fine-tuning

In this section, we examine the utility of descriptions in a supervised setting, where OpenBioNER is fine-tuned on the training set of each dataset. The results, presented in Table 3, indicate that description-based models surpass non-description-based baselines in half of the benchmarks. However, when the NER task reduces to binary classification, the added complexity of descriptions appears unnecessary. While descriptions provide valuable context, they may also introduce biases and constrain the model's ability to learn generalizable patterns from the training data. Interestingly, the more complex JNLPBA benchmark with five target entity types demonstrates a pronounced advantage for OpenBioNER over traditional token classification baselines. In this multi-class scenario, descriptions facilitate better type disambiguation, leading to improved performance.

7 Ablation Studies

7.1 Effect of Pretraining on In-domain Performance

To quantify the impact of pretraining on Pile-NER-biomed, we evaluate OpenBioNER's performance in full-data and low-data scenarios. We simulate the latter conditions by running a few-shot training on 100, 500, and 1000 samples. We run a separate training run for each data setting and compare our model results with and without pretraining.[9] Figure 4 illustrates that OpenBioNER consistently outperforms its non-pretrained counterpart across all dataset sizes. The performance gap is substantial with scarse training data; e.g., with 100 samples per dataset, we register an average difference of 35.6 F1-micro points. These results quantitatively prove that pretraining on Pile-NER-biomed with descriptions promotes effective knowledge transfer and rapid adaptation.

7.2 Effect of Description Quality

The quality of entity type descriptions can significantly impact the performance of description-based models in BioNER tasks. Using overly broad or task-unspecific descriptions can lead to performance underestimation. To this end, Aly et al. (2021) operated with UMLS Metathesaurus descriptions both for training and testing. To quantitatively evaluate the quality of our LLM-based descriptions, we conduct comparative experiments against UMLS-based alternatives in zero-shot settings. Precisely, we measure how performance changes depending on the description source in MedMentions-*Rare*, the benchmark considered by Aly et al. (2021). In analyzing the source effect, we run test set inference with SMXM,[10] the BERT-based X-ENC model from Aly et al. (2021), and our OpenBioNER, pretrained on Pile-NER-biomed. Table 4 illustrates the results. OpenBioNER achieves consistently higher per-class F1 scores across all entity types when using our *task-specific* descriptions compared to UMLS descriptions. Remarkably, SMXM also exhibits improved performance with our descriptions for most entity types, with the sole exception of the Body Substance class. These results provide strong evidence

[9] We openly release our samples. A random seed of 42 was set for reproducibility.

[10] Checkpoint officially released by the authors. The model is pretrained on MedMentions instances labeled with frequent entity types, i.e., no overlapping with MedMentions-*Rare*.

Dataset	# Types	# Samples	Model	Total Time (s)	Samples/s	Latency (s)
NCBI	1	940	OpenBioNER	15.56	60.41	0.0166
			GLiNER	52.34	17.96	0.0557
BC5CDR	2	4797	OpenBioNER	152.04	31.55	0.0317
			GLiNER	276.72	17.33	0.0577
MedMentions-ZS	5	1048	OpenBioNER	72.03	14.55	0.0687
			GLiNER	63.93	16.39	0.0610
JNLPBA	5	3856	OpenBioNER	251.11	15.36	0.0651
			GLiNER	227.63	16.94	0.0590

Table 5: **Inference speed comparison between OpenBioNER and GLiNER.** Tested across four BioNER benchmarks, measured in total time, samples per second, and latency using the Zshot library.

that our synthetic descriptions are not only effective for OpenBioNER but also generalize well to models with different architectures and training distributions. Qualitative examples of generated descriptions can be found in §F.

7.3 Inference Time

While using descriptions improves effectiveness, it inevitably incurs a computational cost, as longer descriptions take more time to process than simple class names. To quantify this overhead, we compare OpenBioNER to GLiNER-large using the IBM Zshot Library[11] (Picco et al., 2023), which supports both models for a direct and consistent evaluation. We consider a subset of benchmarks with varying numbers of entity types using a Tesla T4 GPU (14 GB VRAM). The results are summarized in Table 5. Our findings indicate a slight speed overhead as the number of entity types increases. For datasets with $\leq 2+1$ types (including the negative class), OpenBioNER is significantly faster than GLiNER, reducing inference time by 125 seconds on the BC5CDR test set. However, with $5+1$ types, we observe an overhead of 8 seconds on MedMentions and 24 seconds on JNLPBA, as shown in Figure 5. Despite this, the practical impact remains minimal, as real-world biomedical benchmarks rarely involve more than 10 entity types per task. This holds in various domains as demonstrated by (Zhou et al., 2024), making the theoretical inefficiency relevant only in atypical scenarios with a very high number of entity types.

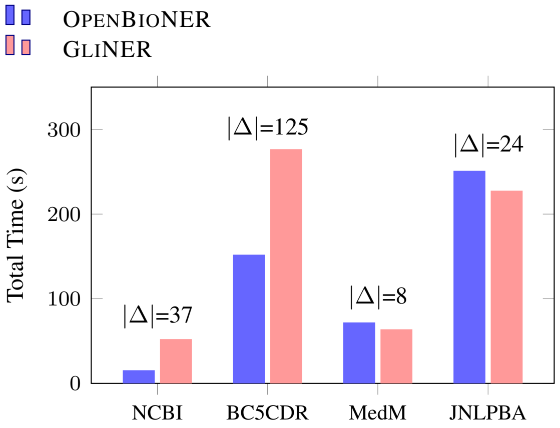

Figure 5: **Total inference time comparison between OpenBioNER and GLiNER.** Calculated by using the Zshot library. "MedM" refers to MedMentions-ZS.

8 Conclusion

We introduced OpenBioNER, a novel model for open-domain biomedical named entity recognition that leverages entity type descriptions for improved generalization in zero-shot settings. Our results demonstrate its superiority over state-of-the-art models, including GPT-4o and UniNER, with significantly fewer resources. OpenBioNER outperforms lightweight alternatives like GLiNER across multiple benchmarks, showcasing its effectiveness in challenging biomedical scenarios. Moreover, it shows easy adapatation to the target domain with limited training data. Future work will explore the application of our approach in diverse domains characterized with data scarcity, such as non-English law (Ragazzi et al., 2024), and aim to develop a more generalist model that balances flexibility and performance. Another promising direction for future research involves the retrieval (Frisoni et al., 2022a) or generation (Frisoni et al., 2024) of entity-specific context and descriptions, which can be optimized end-to-end with the NER task.

[11] https://github.com/IBM/zshot

Limitations

While OpenBioNER demonstrates strong zero-shot performance, its effectiveness is contingent upon the quality of the descriptions used during inference. The absence of a definitive formula for optimal description generation presents a challenge, and further refinement of our approach may yield improved results. Although we propose a fully unsupervised description generation method, we acknowledge that human supervision may be necessary to achieve peak performance. In this context, alternative unsupervised approaches could be explored. Recently, Picco et al. (2024) proposed UDEBO, a method that enhances zero-shot NER performance by aggregating predictions from the same model using multiple automatically generated variants of entity descriptions. This technique has demonstrated improved robustness, suggesting that leveraging diverse descriptions could further refine OpenBioNER's effectiveness. Moreover, unlike GliNER and UniNER, which can handle nested NER cases, OpenBioNER is limited by the BIO-tag scheme, which does not support the identification of entities within other entities. Addressing this limitation is an avenue for future research. Lastly, the generalizability of our model in zero-shot settings is influenced by the entities extracted through self-supervision. The reliance on Pile-NER, which was created using ChatGPT, introduces potential biases and inaccuracies due to irrelevant or incorrect annotations. A more rigorous annotation process with quality control measures could further enhance OpenBioNER's capabilities.

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. Association for Computational Linguistics.

Mona Alshahrani, Mohammad Asif Khan, Omar Maddouri, Akira R. Kinjo, Núria Queralt-Rosinach, and Robert Hoehndorf. 2017. Neuro-symbolic representation learning on biological knowledge graphs. *Bioinform.*, 33(17):2723–2730.

Rami Aly, Andreas Vlachos, and Ryan McDonald. 2021. Leveraging type descriptions for zero-shot named entity recognition and classification. 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 1516–1528, Online. Association for Computational Linguistics.

Alan R. Aronson. 2001. Effective mapping of biomedical text to the UMLS metathesaurus: the metamap program. In *AMIA 2001, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 3-7, 2001*. AMIA.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Sergei Bogdanov, Alexandre Constantin, Timothée Bernard, Benoît Crabbé, and Etienne Bernard. 2024. Nuner: Entity recognition encoder pre-training via llm-annotated data. *CoRR*, abs/2402.15343.

Chih-Yao Chen and Cheng-Te Li. 2021. ZS-BERT: Towards zero-shot relation extraction with attribute representation learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3470–3479, Online. Association for Computational Linguistics.

Nigel Collier and Jin-Dong Kim. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications, NLPBA/BioNLP 2004, Geneva, Switzerland, August 28-29, 2004*.

Dina Demner-Fushman, Willie J. Rogers, and Alan R. Aronson. 2017. Metamap lite: an evaluation of a new java implementation of metamap. *J. Am. Medical Informatics Assoc.*, 24(4):841–844.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yuyang Ding, Juntao Li, Pinzheng Wang, Zecheng Tang, Yan Bowen, and Min Zhang. 2024. Rethinking negative instances for generative named entity recognition. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 3461–3475, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *J. Biomed. Informatics*, 47:1–10.

Giacomo Domeniconi, Marco Masseroli, Gianluca Moro, and Pietro Pinoli. 2014. Discovering new gene functionalities from random perturbations of known gene ontological annotations. In *KDIR 2014 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, Rome, Italy, 21 - 24 October, 2014*, pages 107–116. SciTePress.

Giacomo Domeniconi, Marco Masseroli, Gianluca Moro, and Pietro Pinoli. 2016. Cross-organism learning method to discover new gene functionalities. *Comput. Methods Programs Biomed.*, 126:20–34.

Giacomo Frisoni, Alessio Cocchieri, Alex Presepi, Gianluca Moro, and Zaiqiao Meng. 2024. To generate or to retrieve? on the effectiveness of artificial contexts for medical open-domain question answering. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9878–9919, Bangkok, Thailand. Association for Computational Linguistics.

Giacomo Frisoni, Paolo Italiani, Stefano Salvatori, and Gianluca Moro. 2023. Cogito ergo summ: Abstractive summarization of biomedical papers via semantic parsing graphs and consistency rewards. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 12781–12789. AAAI Press.

Giacomo Frisoni, Miki Mizutani, Gianluca Moro, and Lorenzo Valgimigli. 2022a. BioReader: a retrieval-enhanced text-to-text transformer for biomedical literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5770–5793, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Giacomo Frisoni, Gianluca Moro, and Lorenzo Balzani. 2022b. Text-to-text extraction and verbalization of biomedical event graphs. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2692–2710, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Giacomo Frisoni, Gianluca Moro, and Antonella Carbonaro. 2021. A survey on event extraction for natural language understanding: Riding the biomedical literature wave. *IEEE Access*, 9:160721–160757.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

John M. Giorgi and Gary D. Bader. 2018. Transfer learning for biomedical named entity recognition with neural networks. *Bioinform.*, 34(23):4087–4094.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Heal.*, 3(1):2:1–2:23.

Yu Gu, Sheng Zhang, Naoto Usuyama, Yonas Woldesenbet, Cliff Wong, Praneeth Sanapathi, Mu Wei, Naveen Valluri, Erika Strandberg, Tristan Naumann, and Hoifung Poon. 2023. Distilling large language models for biomedical knowledge extraction: A case study on adverse drug events. *CoRR*, abs/2307.06439.

Maryam Habibi, Leon Weber, Mariana L. Neves, David Luis Wiegandt, and Ulf Leser. 2017. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinform.*, 33(14):i37–i48.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *CoRR*, abs/1904.05342.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

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

Martin Krallinger, Obdulia Rabal, and Florian Leitner. 2015. The CHEMDNER corpus of chemicals and drugs and its annotation principles. *J. Cheminformatics*, 7(S-1):S2.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinform.*, 36(4):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 J. Biol. Databases Curation*, 2016.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

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).

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. *CoRR*, abs/2404.07376.

Hamada A. Nayel, H. L. Shashirekha, Hiroyuki Shindo, and Yuji Matsumoto. 2019. Improving multi-word entity recognition for biomedical texts. *CoRR*, abs/1908.05691.

Rasha Obeidat, Xiaoli Fern, Hamed Shahbazi, and Prasad Tadepalli. 2019. Description-based zero-shot fine-grained entity typing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 807–814, Minneapolis, Minnesota. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Gabriele Picco, Leopold Fuchs, Marcos Martínez Galindo, Alberto Purpura, Vanessa López, and Hoang Thanh Lam. 2024. Description boosting for zero-shot entity and relation classification. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 9441–9457, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Gabriele Picco, Marcos Martinez Galindo, Alberto Purpura, Leopold Fuchs, Vanessa Lopez, and Thanh Lam Hoang. 2023. Zshot: An open-source framework for zero-shot named entity recognition and relation extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 357–368, Toronto, Canada. Association for Computational Linguistics.

Sampo Pyysalo and Sophia Ananiadou. 2014. Anatomical entity mention recognition at literature scale. *Bioinform.*, 30(6):868–875.

Luca Ragazzi, Gianluca Moro, Stefano Guidi, and Giacomo Frisoni. 2024. Lawsuit: a large expert-written summarization dataset of italian constitutional court verdicts. *Artificial Intelligence and Law*, pages 1–37.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2024. Gollie: Annotation guidelines improve zero-shot information-extraction. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Guergana K. Savova, James J. Masanz, Philip V. Ogren, Jiaping Zheng, Sunghwan Sohn, Karin Kipper Schuler, and Christopher G. Chute. 2010. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *J. Am. Medical Informatics Assoc.*, 17(5):507–513.

Larry Smith, Lorraine K. Tanabe, Robert J. Ando, et al. 2008. Overview of biocreative ii gene mention recognition. *Genome Biology*, 9 Suppl 2(Suppl 2):S2.

Wei Wang, Vincent W. Zheng, Han Yu, and Chunyan Miao. 2019a. A survey of zero-shot learning: Settings, methods, and applications. *ACM Trans. Intell. Syst. Technol.*, 10(2):13:1–13:37.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, Jihua Kang, Jingsheng Yang, Siyuan Li, and Chunsai Du. 2023. Instructuie: Multi-task instruction tuning for unified information extraction. *CoRR*, abs/2304.08085.

Xuan Wang, Yu Zhang, Xiang Ren, Yuhao Zhang, Marinka Zitnik, Jingbo Shang, Curtis P. Langlotz, and Jiawei Han. 2019b. Cross-type biomedical named entity recognition with deep multi-task learning. *Bioinform.*, 35(10):1745–1752.

Chih-Hsuan Wei, Bethany R. Harris, Donghui Li, Tanya Z. Berardini, Eva Huala, Hung-Yu Kao, and Zhiyong Lu. 2012. Accelerating literature curation with text-mining tools: a case study of using pubtator to curate genes in pubmed abstracts. *Database J. Biol. Databases Curation*, 2012.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. 2019. Zero-shot learning - A comprehensive evaluation of the good, the bad and the ugly. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(9):2251–2265.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2024. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *CoRR*, abs/2406.08464.

Yuming Yang, Wantong Zhao, Caishuang Huang, Junjie Ye, Xiao Wang, Huiyuan Zheng, Yang Nan, Yuran Wang, Xueying Xu, Kaixin Huang, Yunke Zhang, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Beyond boundaries: Learning a universal entity taxonomy across datasets and languages for open named entity recognition. *CoRR*, abs/2406.11192.

Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, Jie Zhou, Siming Chen, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. A comprehensive capability analysis of GPT-3 and GPT-3.5 series models. *CoRR*, abs/2303.10420.

Wonjin Yoon, Chan Ho So, Jinhyuk Lee, and Jaewoo Kang. 2019. Collabonet: collaboration of deep neural networks for biomedical named entity recognition. *BMC Bioinform.*, 20-S(10):55–65.

Tianyu Yu, Chengyue Jiang, Chao Lou, Shen Huang, Xiaobin Wang, Wei Liu, Jiong Cai, Yangning Li, Yinghui Li, Kewei Tu, Hai-Tao Zheng, Ningyu Zhang, Pengjun Xie, Fei Huang, and Yong Jiang. 2024. Seqgpt: An out-of-the-box large language model for open domain sequence understanding. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 19458–19467. AAAI Press.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2024. GLiNER: Generalist model for named entity recognition using bidirectional transformer. 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 5364–5376, Mexico City, Mexico. Association for Computational Linguistics.

Yuhao Zhang, Yuhui Zhang, Peng Qi, Christopher D. Manning, and Curtis P. Langlotz. 2020. Biomedical and clinical english model packages in the stanza python NLP library. *CoRR*, abs/2007.14640.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2024. Universalner: Targeted distillation from large language models for open named entity recognition. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

A Modeling Non-Entity Tokens

A significant challenge in open-domain NER arises when tokens labeled as non-entities during training are later identified as entities of unseen types during testing. This issue is particularly pronounced in our pretraining setup using Pile-NER-biomed, where tokens of unextracted entity types are initially labeled as ’O’ (non-entity) and remain so until their corresponding types are extracted. To address this limitation, we extend the dynamic negative class modeling approach of Aly et al. (2021), originally designed for closed-domain settings. Unlike their approach, which independently encodes the input sequence without using a negative class description, we find that incorporating cross-attention with the negative description yields more effective results. Specifically, for each input sequence token, we employ a cross-attention mechanism with all possible entity type descriptions, including the negative class, to generate token representations $(\mathbf{v}_{t_{neg}}, \mathbf{v}_{t_1}, \ldots, \mathbf{v}_{t_k})$. We then apply separate linear transformations using weight matrices $\mathbf{W}_{pos}$ and $\mathbf{W}_{neg-ind}$ for the positive and negative class vectors, respectively, allowing the model to learn distinct features for the negative and positive classes. The separate linear layer $\mathbf{W}_{neg-ind}$ allows the model to focus specifically on learning a good representation for the negative class, which might be more challenging to learn than the positive class representations. Negative and positive class vectors are concatenated to form a feature map $\mathbf{m}$. Subsequently, we apply max pooling over this feature set and take the maximum value corresponding to $\mathrm{sim}(\mathbf{t_i}, \mathbf{c_{neg}})$. A detailed mathematical formulation of this process is provided in Algorithm 1.

However, in practice, defining "what is not" with respect to "what is" can be a challenging task, making it difficult for users to make informed decisions about the most suitable negative description. To mitigate this issue, we devise a strategy to reduce the model’s sensitivity to the negative description, allowing for potentially omitting it without compromising performance. In practice, during training we found it beneficial to randomly truncate the negative description for each sentence in the batch, as shown in Figure 6. To evaluate the model’s robustness, we experimented with different strategies for modeling the negative description at test time:

- Out-of-domain descriptions, not related to the biomedical field, to inject noise.
- Randomly sampling three sentences from the

training set with only negative labels associated to each token.

- Using artificially generated descriptions from a LLM.
- Omitting the negative description altogether, to evaluate the model's ability to perform without any explicit negative information.

The results, presented in Table 6, demonstrate that sampling three sentences from each benchmark's training set yields the best performance. Moreover, the results show only small oscillations across different modeling approaches, highlighting the robustness of our model to various negative description representations.

Algorithm 1 Negative Entity Type Encoding

```
Require: Description-sensitive representations $\mathbf{v}_{t_{neg-ind}}, \mathbf{v}_{t_1}, \ldots, \mathbf{v}_{t_k}$
Ensure: Final negative class output $\mathrm{sim}(t_i, \mathbf{d}_{neg})$
  $\mathbf{m} \leftarrow [W_{\text{pos}} \cdot \mathbf{v}_{t_1}, \ldots, W_{\text{pos}} \cdot \mathbf{v}_{t_k}]$  ▷ Projection of positive description-sensitive vectors
  $\mathbf{m} \leftarrow [W_{\text{neg-ind}} \cdot \mathbf{v}_{t_{neg-ind}}, \ldots, W_{\text{pos}} \cdot \mathbf{v}_{t_k}]$  ▷ Add $v_{\text{neg-ind}}$ to $\mathbf{m}$ via separate $W_{\text{neg-ind}}$
  $\mathbf{m} \leftarrow [\mathrm{sim}(t_i, c_{neg-ind}); \ldots; \mathrm{sim}(t_i, c_{|T|})]$  ▷ Obtain per-class similarity score
  $\mathrm{sim}(\mathbf{t_i}, \mathbf{c_{neg}}) \leftarrow \max(\mathbf{m})$  ▷ Max pooling over both positive and negative scores
  return $\mathrm{sim}(\mathbf{t_i}, \mathbf{c_{neg}})$
```

> To our knowledge, this is the first report of a pseudo-allergy caused by polyethylene glycol. **[SEP]** *Coal, water, oil, etc. are normally used for traditional electricity generation. However, using liquefied natural gas as fuel for joint circulatory electricity generation has advantages. The chief financial officer is the only one there taking the fall.*
>
> Five cm H2O CP during nitroprusside did not further alter any of the above-mentioned variables. **[SEP]** *Coal, water, oil, etc. are normally used for traditional electricity generation. However, using liquefied natural gas as fuel for joint circulatory electricity generation has advantages.*

Figure 6: Example of *out-of-domain* negative description (in italic font), randomly truncated for two different sentences within the same batch.

B *-Rare* Dataset Creation

We construct our *-Rare* benchmark variants by following a multi-step process inspired by the method of Aly et al. (2021). The steps involved are as follows:

1. We begin by counting the frequency of each entity type in the dataset.
2. We then allocate the most frequent entity types to the training set, the least frequent to the test set, and the remaining ones to the development set. This ensures that the development set is representative of the test set to the greatest extent possible.
3. For the MedMentions dataset, we adopt the same training, development, and test splits defined by Aly et al. (2021). In the case of the JNLPBA dataset, we reserve the RNA and Cell Type labels as the rarest categories for testing.[12]
4. We designate entities whose classes appear in different sets than their assigned set (as per step 2) as negative ('O') in each sentence.
5. We remove sentences with no annotated entities, i.e., those where every token is labeled as 'O'.
6. Finally, we discard the training and development splits, retaining only the test sets as our final benchmarks for evaluation.

C GLiNER Type Sensibility

During the evaluation of GLiNER on our newly introduced benchmarks, we observed a significant sensitivity of the model to case formats, particularly in the presence of acronyms such as RNA. As shown in Table 2, we report the highest score obtained for each dataset. However, a closer examination of the results reveals that the model's performance varies substantially across different case formats, especially in datasets with more ambiguous entity types, such as JNLPBA. In JNLPBA, entities like Cell Line, RNA, and DNA can have different interpretations, making it challenging to understand their context without guidance from a description. To investigate this phenomenon, we established three evaluation settings: (1) lower case types (e.g., "dna"), (2) title case types (e.g., "Dna"),

[12] Note that for JNLPBA, we do not consider a separate dev split due to the limited number of distinct entity types.

and (3) title case acronym-aware (e.g., "DNA"). Interestingly, using the title case acronym-aware format (DNA and RNA) in JNLPBA boosts the model's performance. However, when we shift to JNLPBA-R, the model's performance drops by approximately 10% F1 points. We further investigate this behavior and find that the model significantly increases false positive predictions when evaluated on the rarest classes.

D Negative Class Weight

During supervised training, we observe that the model benefits from a fixed class weight for negative entities in the cross-entropy loss rather than a dynamic one, as employed during pretraining. We attribute this behavior to the fact that most of the benchmarks considered have a large number of training and test instances without any positive labels (i.e., only 'O' tags). As a result, the model is prone to make false positive predictions. To mitigate this, we assign a fixed weight to the negative class, which helps the model to better balance the learning process and improve its performance on the negative classes. Table 7 shows the results for two different fixed negative weights: 0.5 and 1.0. By comparing these results, we can see the impact of the fixed negative weight on the model's performance and identify the optimal value for each benchmark.

E Hyperparameter Details

Table 8 illustrates all the hyperparameters used in our experiments. This includes hyperparameters for the pretraining and fine-tuning of OpenBioNER, fine-tuning of BERT and BioBERT, description generation with Meta-LLaMA-3.1-8B-Instruct, and inference with GPT-4o. The table provides a comprehensive overview of the hyperparameters used in each experiment, allowing for reproducibility.

F Qualitative Examples

Table 9 presents examples of multi-domain descriptions generated by the LLM. To illustrate the effect of description specificity on model performance, Table 10 compares two descriptions for the entity type *Anatomy* on the AnatEM benchmark. Notably, the first description provides a general overview of anatomy, whereas the second description explicitly mentions specific cell types, body parts, and biological substances, indicating a more detailed understanding of anatomical entities present in the dataset. Instead, Table 11 showcases examples of negative descriptions tested on the BC2GM dataset.

G Prompt Templates

In this section, we present the prompts utilized to address each of the tasks detailed in the paper. Figure 7 provides an illustration of the prompt employed to generate multi-domain descriptions. Figure 8 showcases an example of the prompt template used to generate domain-specific descriptions through few-shot learning. Figure 9 illustrates the prompt template utilized to filter biomedical instances from Pile-NER, resulting in the creation of Pile-NER-Biomed. Figure 10 demonstrates the prompt used by GPT-4o to perform inference on each benchmark.

NEG type	NCBI	AnatEM	JNLPBA	JNLPBA-ZS	BC2GM	BC4CHEMD	BC5CDR	MedMentions-ZS	AVG
Out-of-domain	**58.2**	34.4	56.8	62.2	**48.8**	47.2	**52.6**	49.4	50.2
None	57.9	32.6	**57.1**	62.1	48.2	47.2	51.5	50.6	50.9
LLM-generated	57.9	30.5	**57.1**	62.1	47.4	**48.0**	**52.6**	**50.7**	50.8
Train sample	57.7	**35.2**	**57.0**	**62.4**	47.8	47.5	52.4	50.6	**51.3**

Table 6: **Zero-shot performance changing negative (NEG) descriptions.**

#Samples	AnatEM	NCBI	JNLPBA	BC2GM	BC4CHEMD	BC5CDR	AVG
NEG weight = 0.5							
100	53.2	**67.1**	**60.4**	**53.2**	**51.7**	**74.4**	**60.0**
500	**70.8**	72.0	66.6	64.3	**69.8**	79.5	**70.5**
1000	**77.8**	**78.7**	65.8	**65.0**	75.8	**81.9**	**74.2**
supervised	86.7	**86.1**	**74.5**	80.1	88.3	86.2	**83.7**
NEG weight = 1.0							
100	**55.8**	66.3	57.9	52.4	47.9	73.4	59.0
500	68.9	**73.1**	**66.9**	**65.7**	68.7	**79.6**	**70.5**
1000	76.4	77.1	**66.1**	63.4	**76.1**	**81.9**	73.5
supervised	**87.0**	85.4	74.1	**80.5**	**88.5**	**86.3**	83.6

Table 7: **Supervised in-domain performance as the negative (NEG) class weight varies.** We report the best results obtained within a maximum of eight training epochs.

Provide a description for the following entity types: {types}.
Consider scenarios where an entity type may have different meanings and provide context-aware descriptions accordingly. Always return only a python dictionary ’descriptions’ between ``` containing the descriptions for each entity type.

The expected output is the following:

~~~
```python
descriptions = {
  "type 1": {
    "general": "general description of type 1",
    "domain 1": "In domain 1, ... description of type 1 in domain 1",
    ...,
  },
  "type 2": {
    "general": "general description of type 2",
    "domain 2": "In domain 2, ... description of type 2 in domain 2",
    ...,
  },
  ...
}
```
~~~

Figure 7: Prompt template used to generate multi-domain descriptions.

Hyperparameters	
OPENBIONER pretraining f ine-tuning	
min_sample_class	15 / -
max_sample_class	25 / -
max_description_length	150 / 150
max_sequence_length	300 / 300
mask_probability	0.3 / 0.5
epochs	4 / 8
batch_size	8 / 8
val_steps	1
learning_rate	2e-5 / 2e-5
negative_weight	dynamic / 1.0
optimizer	Adam / Adam
weight_decay	0 / 0
linear_dropout	0.5 / 0.5
warmup_step	0 / 0
seed	42 / 42
BERT & BIOBERT fine-tuning	
learning_rate	2e-4
per_device_train_batch_size	64
per_device_eval_batch_size	64
num_train_epochs	5
evaluation_strategy	epoch
save_strategy	epoch
metric_for_best_model	f1_micro
gradient_accumulation_steps	1
eval_accumulation_steps	32
optimizer	AdamW
warmup_ratio	0.1
lr_scheduler_type	cosine
seed	42
LLAMA-3.1-8B-INSTRUCT MULTI-DOMAIN AND DOMAIN-SPECIFIC DESCRIPTION GENERATION	
n	1
temperature	0.5
max_tokens	1024
top_p	1.0
frequency_penalty	0
presence_penalty	0
seed	0
GPT-4O INFERENCE	
model	gpt-4o-2024-08-06
n	1
temperature	0.0 (*greedy decoding*)
max_tokens	256
top_p	1
frequency_penalty	0
presence_penalty	0

Table 8: Hyperparameters used for training/fine-tuning of OpenBioNER, fine-tuning BERT and BioBERT, description generation with Meta-LLaMA-3.1-8B-Instruct, inference with GPT-4o.

Type	**Description**
Observation	An observation is a recorded or documented event, action, or state that is observed or measured by an individual or a system. Observations can provide insights into phenomena, processes, or systems, and can be used to generate hypotheses test theories, or make predictions. In science, observations are used to gather data and evidence for scientific inquiry. For example, astronomers may make observations of celestial objects to study their properties and behaviors. In engineering, observations are used to monitor and control systems. For example, sensors may be used to observe the temperature, pressure, or flow rate of a process, and to adjust the system parameters accordingly. In social sciences, observations are used to study human behavior and social phenomena. For example, anthropologists may make observations of cultural practices and rituals to understand their meaning and significance.
Hypothesis	A proposed explanation for a phenomenon, based on available evidence and reasoning. Can be tested through experimentation or observation. In the context of science, a hypothesis is a tentative explanation for an observation or a set of observations, that can be tested through further investigation. In the context of philosophy, a hypothesis is a speculative idea or assumption, that may or may not be supported by evidence or reasoning.
Medical Assessment Tool	A medical assessment tool is a device, software, or technique used to evaluate a patient's health status or diagnose a medical condition. This can include physical exams, diagnostic tests, questionnaires, and other methods. In primary care, a medical assessment tool may be used to screen for chronic diseases, monitor symptoms, or guide treatment decisions. For example, a blood glucose meter is a medical assessment tool used to measure blood sugar levels in patients with diabetes. In hospital settings, medical assessment tools may be used to monitor patients' vital signs, assess pain levels, or evaluate cognitive function. For example, a pulse oximeter is a medical assessment tool used to measure oxygen saturation in the blood. In research settings, medical assessment tools may be used to collect data for clinical trials or epidemiological studies. For example, a quality of life questionnaire is a medical assessment tool used to measure the impact of a disease or treatment on a patient's daily life.
Gram-Positive Organism	A gram-positive organism is a type of bacterium that retains the crystal violet stain during the Gram stain process. This characteristic is due to the thick peptidoglycan cell wall that these bacteria have. Gram-positive organisms include common pathogens such as Staphylococcus aureus and Streptococcus pneumoniae. In medicine, gram-positive organisms are important pathogens that can cause a variety of infections, such as skin infections, pneumonia, and sepsis. Antibiotics such as penicillin and vancomycin are commonly used to treat infections caused by gram-positive bacteria. In microbiology, gram-positive organisms are studied to understand their unique cell wall structure and the mechanisms by which they cause disease. They are also used as models to study antibiotic resistance and the development of new antibiotics.

Table 9: Examples of *multi-domain* descriptions generated by LLaMA-3.1-8B-instruct for Pile-NER-biomed types.

Description	F1 Score
Anatomy refers to organs, tissues, cells, and their spatial and functional relationships. Anatomy explores physical structures, such as the heart, lungs, and bones, as well as microscopic elements like cells and tissues. Examples include the study of cell structures in HeLa cells, the arrangement of plectonemes in DNA, and the analysis of serum levels in medical research.	28.8
The anatomy refers to biological components at various scales, including cells, tissues, and organs. These entities can be identified by proper nouns referring to cell types (e.g. HeLa cells, neurospheres, NSCLC, SCC), body parts (e.g. serum, blood) or biological substances (e.g. vegetables, meats, cow milk) or tumors.	34.8

Table 10: Comparison of two descriptions for the entity type *Anatomy* and their respective F1 scores in the AnatEM dataset, illustrating how specificity and detail influence model performance.

Negative type	Description
`Out-of-domain`	Coal, water, oil, etc. are normally used for traditional electricity generation. However using liquefied natural gas as fuel for joint circulatory electircity generation has advantages. The chief financial officer is the only one there taking the fall. It has a very talented team, eh. What will happen to the wildlife? I just tell them, you've got to change. They're here to stay. They have no insurance on their cars. What else would you like? Whether holding an international cultural event or setting the city's cultural policies, she always asks for the participation or input of other cities and counties.
`LLM`	terms that do not represent specific biological concepts such as genes or hereditary units. These words serve to structure and enhance language but lack specialized meaning in genetics or biology. These words include common linguistic elements like articles, prepositions, conjunctions, or auxiliary verbs, which are necessary for sentence construction but do not convey scientific information. They do not refer to biological sequences, inheritance mechanisms, or traits, instead serving as general connectors or modifiers that provide clarity and coherence to statements without carrying domain-specific significance.
`Train`	In vivo epiluminescence microscopy of pigmented skin lesions . Current status of zinc deficiency in the pathogenesis of neurological , dermatological and musculoskeletal disorders . The 3 - hour test iodine (I - 132) uptake by the thyroid in children with growth deficiency .

Table 11: Example of various negative description types used for testing the model with BC2GM. The `out-of-domain` approach, adapted from Aly et al. (2021), comprises 10 negative sentences sourced from the OntoNotes dataset. The `LLM` description is generated by a LLM that had access to the positive entity descriptions. The `Train` description is formed by concatenating 3 negative sentences from the reference dataset.

> You are provided with a class.
>
> Your task is to generate a brief and accurate description that can generally represent that class in a sentence.
>
> To better understand possible domains of the class you are provided with a list of sentences containing entities related to the class, each entity is enclosed within ### markers.
>
> Include examples.
>
> The description should be between 40 and 120 words.
>
> Class: **{class}**
>
> Sentences:
> 1. **{sentence 1}**
>
> 2. **{sentence 2}**
> ...
>
> Please generate the description of the class, prefixed by exactly '###description:'

Figure 8: Prompt template used to generate domain-specific descriptions through few-shot examples.

> Sentence: **{sentence}**
> Labels: **{labels}**
> You are given a sentence splitted in token and relative label annotation. Respond with 'Yes' only if sentence and labels are both relative to medical domain, 'No' instead.
>
> Don't explain your answer.
>
> Use every label one time.
>
> Use None when there are no entities related to the label.

Figure 9: Prompt template used to filter only biomedical samples from Pile-NER.

> Please identify Bacterium, Body Substance, Body System, Food, Professional or Occupational Group from the given text, output using the format as "Entity: Bacterium: None|Body Substance: Word1|Body System: None|Food: Word2, Word3|Professional or Occupational Group: None|"
>
> Use every label one time.
>
> Use None when there are no entities related to the label.
>
> Text: **{text}**

Figure 10: Example of prompt templated used for GPT-4o to run evaluation on MedMentions-R.