ZeroNER: Fueling Zero-Shot Named Entity Recognition via Entity Type Descriptions

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

Abstract

What happens when a named entity recognition (NER) system encounters entities it has *never* seen before? In practical applications, models must generalize to unseen entity types where labeled training data is either unavailable or severely limited—a challenge that demands zero-shot learning capabilities. While large language models (LLMs) offer extensive parametric knowledge, they fall short in cost-effectiveness compared to specialized small encoders. Existing zero-shot methods predominantly adopt a relaxed definition of the term with potential leakage issues and rely on entity type names for generalization, overlooking the value of richer descriptions for disambiguation. In this work, we introduce ZeroNER, a description-driven framework that enhances *hard* zero-shot NER in low-resource settings. By leveraging general-domain annotations and entity type descriptions with LLM supervision, ZeroNER enables a BERT-based student model to successfully identify unseen entity types. Evaluated on three real-world benchmarks, ZeroNER consistently outperforms LLMs by up to 16% in F1 score, and surpasses lightweight baselines that use type names alone. Our analysis further reveals that LLMs derive significant benefits from incorporating type descriptions in the prompts.[1]

1 Introduction

Named entity recognition (NER) involves identifying entity mentions and assigning them to predefined types (classes). Despite the upswing in harnessing the expressive power of pre-trained language models, current solutions mostly demand large-scale datasets and prioritize a handful of commonly occurring types (e.g., person, location, organization) (Wang et al., 2023). Real-world NER applications frequently grapple with specialized domains and lack readily available labels for non-standard types, with new targets constantly emerging. In this context, adopting or periodically retraining state-of-the-art (SOTA) models becomes impractical. Thus, the impetus shifts to zero-shot learning (ZSL) (Xian et al., 2019; Aly et al., 2021; Picco et al., 2023), where networks must generalize to entity types not experienced during training by effectively transferring knowledge gleaned from observed types. In this paradigm, prior knowledge emerges as an essential asset (Hsieh et al., 2023). Simultaneously, creating substantial ground truths remains an expensive and time-consuming activity, prone to human errors and inconsistencies that might affect model effectiveness.

The advent of large language models (LLMs) has revolutionized zero-shot NER by enabling recognition of arbitrary entity types through natural language instructions (Wang et al., 2023; Zhou et al., 2023). This represents a significant departure from

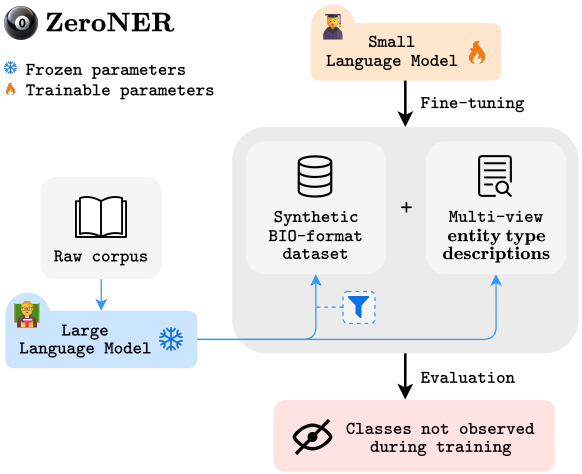

Figure 1: **Overview of ZeroNER.** The framework derives large-scale, high-quality annotations and entity-type descriptions from a teacher LLM. A student SLM is then fine-tuned on this general-domain data and evaluated on unseen entity types without any type of leakage.

[*]Equal contribution (co-first authorship).
[1]Data, code, model: zeroner; research group website: unibo-nlp.github.io

Findings of the Association for Computational Linguistics: ACL 2025, pages 15594–15616

the rigidity of traditional models. However, LLMs demand significant computational power. Although some LLMs are accessible through APIs (OpenAI, 2024), large-scale usage can incur prohibitive costs.

To address these challenges, we introduce ZERONER, a novel framework designed to enhance zero-shot NER capabilities in small language models (SLMs). ZERONER operates by distilling both annotations and entity type descriptions from LLMs (Figure 1). Rather than depending on large autoregressive models, our approach employs a more efficient BERT-based cross-encoder that incorporates textual descriptions as contextual information to accurately identify *any* named entity across domains. First, we generate a domain-diverse distillation dataset through a frozen LLM. Then, to ensure data quality, we implement an LLM-based self-correction mechanism combined with heuristic filtering. The SLM is fine-tuned on our general-domain silver dataset, which encapsulates broad NER knowledge, and is then directly evaluated on the test set of the target dataset.

We experiment with the zero-shot-adapted versions of three widely used NER datasets: MedMentions (Mohan and Li, 2019), OntoNotes 5.0 (Pradhan et al., 2013), and LegalNER (Kalamkar et al., 2022). Unlike recent approaches, we adopt a *hard* zero-shot evaluation protocol to ensure realistic performance assessment. Our evaluation adheres to the strict constraints established by Xian et al. (2019): (1) complete separation of entity types across (silver, general-domain) training, (gold, in-domain) development, and (gold, in-domain) test splits, (2) reservation of the rarest labels for development and test sets, and (3) class imbalance accounting via *macro*-averaged per-class metrics. Our SLMs not only outperform all evaluated LLMs with up to 8B parameters but also exceed the performance of alternative SLMs that benefit from data contamination, i.e., training exposure to entity classes used for benchmarking.

Our contributions are as follows:

- We release a lightweight, open-source model for zero-shot NER using class descriptions, achieving SOTA results over costly LLMs.
- We prove that entity type descriptions boost LLM performance in zero-shot NER, underscoring their value for generalization.
- We advocate for a *hard* zero-shot evaluation protocol that eliminates entity type leakage and ensures a more accurate assessment of zero-shot NER capabilities.
- We provide an open-source silver dataset, suitable for both research and commercial use, to promote further advances in zero-shot NER.

2 Related Work

Entity Type Descriptions. A growing body of work explores the use of textual metadata to improve generalization in NER. Obeidat et al. (2019) and Aly et al. (2021) demonstrated that injecting manually crafted definitions of entity types into encoders supports a more accurate classification. Nguyen et al. (2021) further extended this idea by incorporating structured information from knowledge graphs, allowing the model to capture cross-domain relationships between entities. Recently, interest has moved toward adapting LLMs to follow explicit entity type definitions within the prompt. Methods such as GoLLIE and SLIMER (Sainz et al., 2024; Zamai et al., 2024) fine-tune LLMs to align with natural language guidelines, achieving strong zero-shot performance on NER tasks. However, such gains come with considerable computational costs that are not applicable in resource-constrained environments. To the best of our knowledge, OpenBioNER (Cocchieri et al., 2025) is the only prior work that, like ours, investigates the use of entity type descriptions within encoder-based models for open-domain objectives. However, (i) its focus is restricted to the biomedical domain, (ii) it foresees a second-stage in-domain fine-tuning, and (iii) it does not account for type leakage between silver train annotations and evaluation splits.

Knowledge distillation. Knowledge distillation (KD) centers on transferring knowledge from large, high-capacity teacher models to more compact student models (Bucila et al., 2006; Hinton et al., 2015). In this work, we draw upon LLM-based KD in a self-supervised setting (Agrawal et al., 2022; Zhou et al., 2023), where a teacher model is used to annotate unlabeled data. Recent efforts have applied this procedure to a wide array of downstream tasks (Bussotti et al., 2024), including NER. Two main directions have emerged in this space: generative and encoder-based student models.

Generative student models employ autoregressive architectures to model NER labels as output texts. InstructUIE (Wang et al., 2023) fine-tunes Flan-T5-xxl (Chung et al., 2022) on a mixture of information extraction datasets, yielding strong zero-shot NER performance. UniversalNER (Zhou et al., 2023) fine-tunes LLaMA using data annotated by

ChatGPT, often matching or surpassing ChatGPT's own zero-shot NER results. Importantly, it requires extracting one entity type at a time, increasing LLM inference costs for standard multi-class problems. GNER (Ding et al., 2024) incorporates negative instances and adopts a BIO-tagging scheme for finer-grained supervision. However, all generative approaches are inherently limited by slow, token-by-token decoding and high resource demands.

In contrast, encoder-based student models resort to more computationally efficient architectures as backbones for KD. GLiNER (Zaratiana et al., 2023) distills UniversalNER annotations into a DeBERTa (He et al., 2021) student model. It formulates NER as a span–entity type matching problem in embedding space and has demonstrated superior zero-shot results compared to both ChatGPT and fine-tuned LLMs. Bogdanov et al. (2024) pretrain RoBERTa (Liu et al., 2019) on ChatGPT annotations through contrastive-learning, achieving results on par with LLMs in few-shot settings.

Despite these advances, a recurring limitation across current distillation frameworks lies in the use of silver datasets that include entity types overlapping with those in evaluation benchmarks. This results in evaluations conducted under *soft* zero-shot conditions, where type-level contamination can lead to overly optimistic assessments of generalization capabilities (see §D). As our work is grounded in *hard* zero-shot NER, direct comparisons with these prior approaches are not fair and therefore fall outside the scope of this paper.

3 Method

3.1 Task Definition

We formulate NER as a sequence labeling task. Given a sentence $\mathbf{s} = \{t_1, \ldots, t_n\}$ comprising n tokens and a description $\mathbf{d_c}$ for each entity class $c \in \mathbb{C}^{test}$ in the test set, we predict a sequence of annotations $\hat{\mathbf{y}} \in (\mathbb{C}^{test})^n$. The classification of a token at position i is determined by $\arg\max_{c \in \mathbb{C}^{test}} \mathcal{F}_\theta(\mathbf{s}, t_i, \mathbf{d_c})$, where $\mathcal{F}$ models the semantic affinity between t_i and $\mathbf{d_c}$ in the context of $\mathbf{s}$. Parameters θ must be acquired without labeled data for $\mathbb{C}^{test}$, but with labeled data and descriptions for training classes in $\mathbb{C}^{train}$. ZeroNER considers a wealth of LLM-generated examples as training data; each input-output pair $<\tilde{\mathbf{s}}, \tilde{\mathbf{y}}>$ refers to a sentence $\tilde{\mathbf{s}}$ from an external corpus $\mathcal{R}$ and a target sequence $\tilde{\mathbf{y}}$ linked to a large volume of artificial classes $\tilde{\mathbb{C}}$ and their descriptions.

3.2 Synthetic BIO-format Dataset

The following paragraphs break down the LLM-to-dataset workflow, depicted in Figure 2. We define $\mathcal{R}$ by randomly sampling 50k instances[2] from the Pile Uncopyrighted,[3] avoiding any legally encumbered content. Although ZeroNER is model-agnostic, we select Llama-3.1-8B-instruct as the LLM teacher due to its strong empirical performance on the benchmarks considered and its permissive license, which allows smaller models to be trained on its outputs without any restriction.

Prompt-based tagging. We approximate expert labelers by prompting the teacher LLM. Specifically, we design a prompt template to jointly perform (1) entity recognition, (2) entity classification, and (3) domain extraction, while also handling cases of nested entities not supported by the BIO-tagging scheme (see §A). This process leads to ∼350K annotated sentences. Subsequently, we employ the teacher LLM itself to evaluate its predictions using an LLM-as-a-judge approach, assessing correctness and completeness under predefined criteria (see §A). Each prediction is scored from 0 to 3 on both dimensions. To ensure quality, we retain only annotations with a completeness score of at least 2 and a correctness score of 3, resulting in a final dataset of approximately 91K sentences.

Filtering. To further promote data quality without relying on human supervision, we introduce a series of "critic rules" to regulate the accuracy and confidence of the knowledge being transferred. (1) We exclude sentences accompanied by improperly formatted generations or without any positive annotations—1.44% of the total. (2) We filter out recognized entities not mentioned in the input sentence. (3) We discard entities whose mentions consist solely of functional words, i.e., stopwords, verbs, conjunctions, or prepositions. (4) We use lemmatization to tackle the complexity arising from entity mentions having different labels, such as synonyms or inflected forms. For example, *lawyers* and *defense lawyer* can be assigned to the root *lawyer*. (5) We eliminate rare entity classes that appear only once, ∼800 cases.

BIO-format instances. We parse the filtered LLM output and represent named entities in BIO format (short for Beginning, Inside, Outside).

[2] A number of passages known to be sufficient for satisfactory NER distillation (Zhou et al., 2023).

[3] The Pile Uncopyrighted

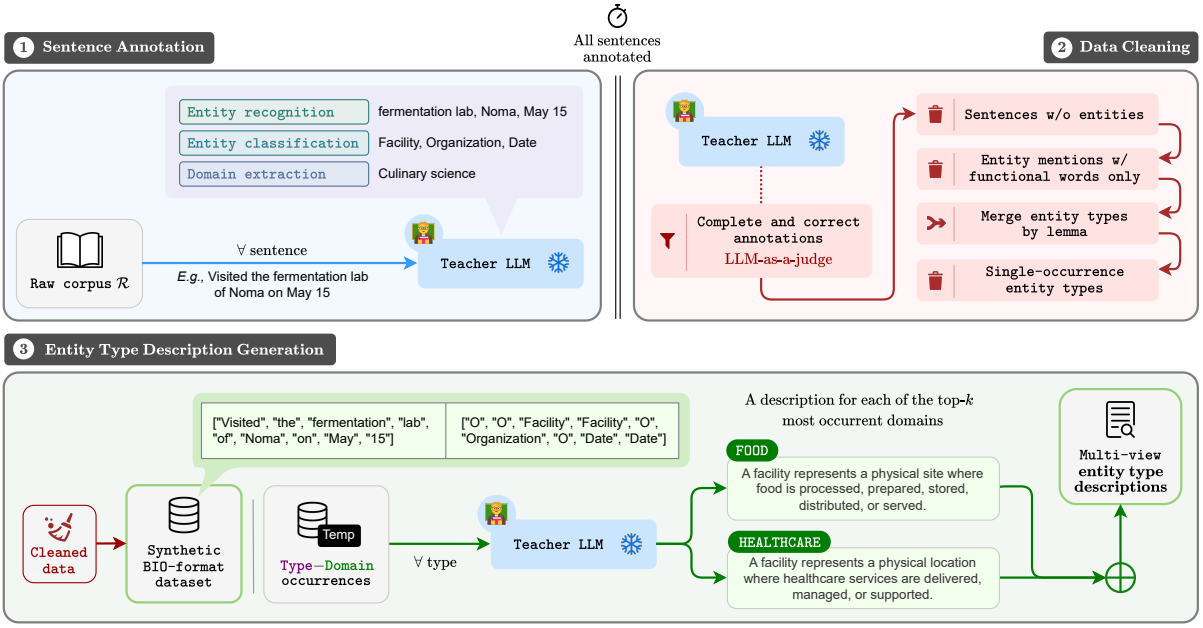

Figure 2: **ZeroNER pipeline for transfer dataset construction.** First, the teacher LLM annotates raw sentences with entity mentions, types, and domains. Second, silver annotations undergo data cleaning for the BIO-format dataset population. Third, the teacher LLM is prompted to generate multi-view descriptions of all the detected entity types according to the most frequent domains in which they occur.

Since the LLM thrives on free generation, the derived annotations are not accompanied by offset indices communicating their token position in the original sentence. Any predicted entity type overlapped with benchmark labels ($\mathbb{C}^{dev}$ or $\mathbb{C}^{test}$) is overwritten with the negative class (O). Given the small number of consecutive entities belonging to the same class (∼1.22% on average across all datasets), we remove all I- and B- prefixes, resulting in a single label per entity class.

Data statistics. After filtering, our dataset comprises 91,768 sentences, encompassing 207,245 entities and 5,731 distinct entity types. Table 1 analyzes our transfer BIO-format dataset. The distributions of entity types and domains exhibit a pronounced heavy-tailed pattern, where the top 1% accounts for 70% of the cumulative frequencies. We unveil a broad spectrum of classes that span various disciplines. A noteworthy observation is the existence of granularity variations within particular entity types, e.g., *Protein* and *Gene* are subsets of *Biological entity*. These attributes greatly enhance the dataset's potential to universally capture LLM capabilities.

3.3 Entity-type Descriptions

As extra guidance during $\mathcal{F}$ training, descriptions establish a more explicit connection between t_i and the expected class y_i. For instance, given a sentence *"The court ruled that the provisions of the Freedom of Information Act were applicable"* and a class *Law*, the description *"Law refers to a system of rules created and enforced through social or governmental institutions to regulate behavior"* can help the model to make the clue, even without previous exposure to training examples. Plural domains could offer separate definitions for a specific class, granting more opportunities to absorb LLM knowledge. For this reason, we devise a prompting technique oriented to multi-view descriptions (see §A). Formerly collected domain labels are used to count the predominant fields in which each entity type finds mention. For every class $\tilde{\mathbf{y}}$, we request the LLM to generate individual descriptions linked to the three leading domains.

Frequency	Main entity types
Top-1% (70%)	Person, Organization, Location, Event, Disease, Time
1%-10% (21%)	Project, Research, Ingredient, Statistic, Year, Prize
10%-100% (9%)	Healthcare, Political, Ownership, Astronomy, Metadata

Table 1: **Entity type distribution overview.** Examples of classes grouped by frequency range, along with the proportion of entity occurrences each range represents. Domains include politics, law, finance, medicine, literature, history, music, and math.

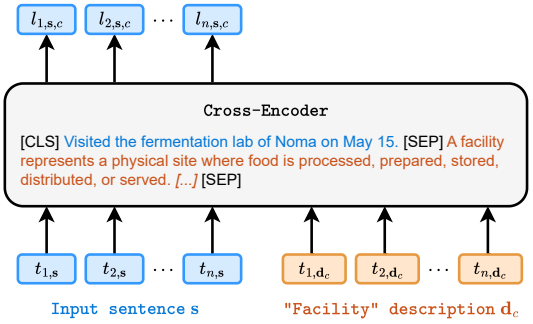

Figure 3: **Student model architecture.** The target entity type description is injected into the input sentence token representations through cross-attention, aiding the model in classifying their semantic relatedness.

Ultimately, the outputs are concatenated following their ranking sequence. We note that the average number of tokens per multi-view description is 47.

3.4 Student Network

Our SLM is based on pre-trained BERT-base (Devlin et al., 2019). ZeroNER's $\mathcal{F}$ modeling parallels the cross-attention encoder (X-ENC) presented by Aly et al. (2021), as illustrated in Figure 3. For notational convenience, we use $\mathbb{C}$ to refer to target classes, regardless of whether they are synthetically generated (training) or come from a gold dataset (evaluation). With every description $\mathbf{d}_c$, X-ENC produces a vector representation $\mathbf{v}_{i,c} \in \mathbb{R}^h$ for each token t_i in sentence $\mathbf{s}$:

$$\mathbf{v}_{1,c}, \ldots, \mathbf{v}_{n,c} = \text{X-ENC}(\mathbf{s}, \mathbf{d}_c), \quad (1)$$

where $h = 768$ and the input tuple $(\mathbf{s}, \mathbf{d}_c)$ is structured in the form `[CLS]` $\mathbf{s}$ `[SEP]` $\mathbf{d}_c$. The vector $\mathbf{v}_{i,c}$ is transformed through a learnable linear layer to quantify $l_{i,c} \in \mathbb{R}$. The value $l_{i,c}$ indicates how likely the token t_i belongs to the entity class c:

$$l_{i,c} = \mathbf{v}_{i,c} \cdot \omega^T + b. \quad (2)$$

To identify entities other than classifying them, we append the token scores $l_{i,c_1}; \ldots; l_{i,c_{|\mathbb{C}|}}$ with the score of belonging to the negative class c_{neg}:

$$l_i = (l_{i,c_1}; \ldots; l_{i,c_{|\mathbb{C}|}}; l_{i,c_{neg}}). \quad (3)$$

To derive c_{neg}, we use the same class-aware encoding as in (Aly et al., 2021), which involves combining the cross-encoder representations of positive classes and then applying max-pooling. After undergoing Softmax, the top-scoring class is chosen.

	OntoNotes-ZS		MedMentions-ZS		LegalNER-ZS	
	dev	**test**	**dev**	**test**	**dev**	**test**
# sentences	1,358	426	1,289	1,048	602	1,227
# words	39,349	12,624	37,297	29,783	20,623	42,808
# entities	1,735	533	1,710	1,430	808	1,777
# compound entities	219	229	292	167	387	841
# consecutive same class	11	0	5	10	0	0
# types	4	3	5	5	3	4
avg sentence length	28.98	29.63	28.93	28.42	34.26	34.89
avg entities per sentence	1.28	1.25	1.33	1.36	1.34	1.45

Table 2: **Summary statistics of zero-shot NER benchmarks**. For each dataset and split, we report the number of sentences, words (whitespace-separated tokens), entities (including compound and consecutive same-class entities), entity types, as well as average sentence length and entities per sentence.

3.5 Training

In ZeroNER, the student SLM is trained to mimic the type-unbounded silver entity annotations of the teacher LLM, additionally exploiting its type descriptions. We manage the class imbalance induced by c_{neg} by incorporating class weights q_c into the cross-entropy loss:

$$\mathcal{L} = -\sum_{c}^{\mathbb{C}} q_c \cdot y_{i,c} \cdot log(p(\hat{y}_{i,c})), \quad (4)$$

where $y_{i,c}$ is the truth label and $p(y_{i,c})$ is the Softmax probability for class c. We consistently set q to 1 for positive classes, while fine-tuning it as a hyperparameter for negative class, using the ratio $\frac{\text{\# entities}}{\text{\# non-entity words}}$ from the training data as a reference.

To effectively manage the variety of entity types, we adopt a progressive training approach, gradually introducing the model to different classes, i.e., *incremental learning*. At each step, the model processes a random subset of 20 to 25 entity types (Figure 4). Any target entity types not among the sampled set are labeled as 'O'. In our methodology, the model is trained solely on entity type descriptions (e.g., *"Human being able to speak and think"*) rather than explicit class names such as *Person*. Even if a class appears in both the distilled and test datasets, variations in descriptions—often reflecting domain-specific nuances—lead the model to treat them as distinct types. To ensure that successive evaluations reflect truly *hard* generalization, at each training step, we also exclude from the sampled types: (1) those with labels that exactly match or contain any benchmark label (e.g., *Art* vs. *Work*

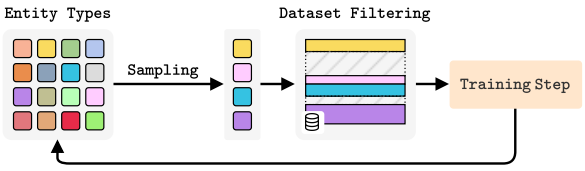

Figure 4: **Iterative class sampling and synthetic instance selection during training.** In each step, a random assortment of entity types is selected, and sentences of the dataset that do not contain any of these targets are filtered out for that step.

of Art), and (2) those whose descriptions exhibit high cosine similarity (above 0.95) to benchmark descriptions, thus mitigating overlap due to synonyms (e.g., *Location* vs. *Place*).[4] To sum up, this filtering prevents the model from encountering nearly identical training and test data.

4 Experimental Settings

Datasets. To gauge ZeroNER holistically, we conduct experiments on three heterogeneous English-language NER datasets: MedMentions-ST21pv (Mohan and Li, 2019), OntoNotes 5.0 (Pradhan et al., 2013), and LegalNER (Kalamkar et al., 2022). MedMentions *(CC0 license)* holds 4K biomedical abstracts and a fine-grained class ontology rooted in the Unified Medical Language System (UMLS). OntoNotes *(LDC license)* is a large-scale multigenre corpus targeting domain-general entity types, including values. LegalNER *(MIT license)* annotates Indian court judgments published between 1950 and 2022.

Building on the recommendations of Xian et al. (2019), we create zero-shot adapted versions of these benchmarks (identified by the "-ZS" suffix). In pursuit of this, we adhere to the conversion protocol suggested by Aly et al. (2021), alternatively leaving the rarest classes in the dev and test sets. The chosen types are guaranteed to be not trivial to recognize; we omit classes whose surface form displays regular patterns (e.g., *Percent*, *Date*). After filtering, the total number of entity types within MedMentions, OntoNotes, and LegalNer is 21, 11, and 13, respectively. Annotations on split-disconnected classes are eliminated. An overview of the data distribution is shown in Table 2. Regarding class description sources for evaluation datasets, we draw on the UMLS Metathesaurus for MedMentions and annotation guidelines for OntoNotes and LegalNER. See §B, §C, and §I for details.

Baselines. We compare ZeroNER with SLMs and LLMs on pure zero-shot.

- SMXM (Aly et al., 2021). BERT-large model with X-ENC for class description integration, similar to ZeroNER. Unlike our approach, this model is fine-tuned on the train split of the target dataset—where train, dev, and test entity types are disjoint—rather than a synthetic dataset. All reported results are recomputed and not taken from the original paper.[5]
- Llama-3.1-8B, Llama-3.2-3B (Grattafiori et al., 2024), Phi-3.5-mini (Abdin et al., 2024), Granite-3.0-8B (Granite Team, IBM, 2024), Mistral-7B-v0.3 (Jiang et al., 2023), and Qwen2.5-7B (Yang et al., 2024): SOTA instruct-tuned, open-source LLMs with autoregressive decoder-only architectures. We construct two types of prompts to detect entities—first by leveraging only the type names, and then by incorporating type descriptions. For details, see §A.
- GLiNER Models (Zaratiana et al., 2023): Current leading models in the zero-shot NER setting. These models employ cross-encoders with DeBERTa as backbone architectures, modeling the NER task as matching entity types with textual spans in a latent space. We evaluate both the v1 and v2.1 families across their small, medium, and large variants. The v1 family is restricted to research use as it was trained on ChatGPT annotations, whereas the v2.1 family is licensed under Apache 2.0, having been trained on Mistral annotations. Although we include these models for comparison, it is important to note that many labels were likely encountered during their training, artificially boosting their performance.

Metrics. We evaluate models with *macro-averaged* precision, recall, and F_1 metrics at the span level. Per-class averaging prevents frequent entity types from dominating performance.

Implementation details. All experiments were conducted on an Nvidia A100 GPU with 80GB of

[4]Similarity measured with sentence-transformers/all-MiniLM-L6-v2

[5]We identify three reasons: (1) discrepancies in entity occurrences after reconstructing the non-public SMXM-ZS datasets; (2) significantly lower performance of the available model checkpoints compared to reported metric; (3) the original training likely used different descriptions from ours, though this cannot be confirmed.

			OntoNotes-ZS			MedMentions-ZS			LegalNER-ZS			
		Size	R	P	F1	R	P	F1	R	P	F1	AVG
Type name	Granite-3.0-8b-instruct	8B	0.30	0.14	0.19	0.38	0.14	0.20	0.45	0.24	0.31	0.23
	Llama-3.1-8B-Instruct	8B	**0.43**	0.21	0.27	0.41	0.32	0.32	0.44	0.19	0.26	0.28
	Llama-3.2-3B-Instruct	3B	0.34	0.10	0.15	0.41	0.26	0.27	0.44	0.19	0.26	0.23
	Mistral-7B-Instruct-v0.3	7B	0.30	0.19	0.23	0.25	0.21	0.22	0.30	0.19	0.23	0.23
	Phi-3.5-mini-instruct	3.8B	**0.43**	0.15	0.22	0.40	0.20	0.22	0.41	0.19	0.25	0.23
	Qwen2.5-7B-Instruct	7B	0.35	0.23	0.27	0.15	0.34	0.18	0.44	0.27	0.32	0.26
	GLiNER-S-v1*	166M	0.41	0.30	0.32	0.51	0.38	0.43	0.38	0.25	0.28	0.34
	GLiNER-M-v1*	209M	0.39	0.34	0.32	0.53	0.44	0.48	0.47	0.27	0.33	0.38
	GLiNER-L-v1*	459M	0.42	0.34	**0.35**	**0.54**	**0.46**	**0.49**	**0.49**	**0.31**	**0.34**	**0.39**
	GLiNER-S-v2.1*	166M	0.24	0.31	0.21	0.38	0.41	0.39	0.33	**0.31**	0.24	0.28
	GLiNER-M-v2.1*	209M	0.28	0.36	0.28	0.42	0.43	0.42	0.37	0.25	0.28	0.33
	GLiNER-L-v2.1*	459M	0.30	**0.42**	0.30	0.44	0.42	0.42	0.32	0.23	0.24	0.32
Type description	Granite-3.0-8b-instruct	8B	0.34	0.22	0.26	0.22	0.17	0.18	0.35	0.29	0.32	0.25
	Llama-3.1-8B-Instruct	8B	0.48	0.33	0.38	0.40	0.40	0.34	0.27	0.20	0.23	0.32
	Llama-3.2-3B-Instruct	3B	0.55	0.24	0.33	**0.47**	0.24	0.28	0.42	0.25	0.31	0.31
	Mistral-7B-Instruct-v0.3	7B	0.32	0.32	0.32	0.19	0.20	0.17	0.32	0.32	0.32	0.27
	Phi-3.5-mini-instruct	3.8B	0.49	0.22	0.30	0.44	0.22	0.24	**0.47**	0.20	0.27	0.27
	Qwen2.5-7B-Instruct	7B	0.49	0.36	0.41	0.30	0.31	0.28	0.38	**0.40**	**0.37**	0.35
	SMXM	345M	0.31	0.14	0.19	0.44	0.18	0.23	0.22	0.18	0.15	0.19
	ZeroNER (Ours)	110M	**0.61**	**0.44**	**0.50**	0.32	**0.42**	**0.35**	0.43	0.26	0.32	**0.39**

* Lower-case entity types (threshold=0.5); see §E for title-case results. GLiNER scores are provided as a reference, as exposed to test entity type names during training.

Table 3: **Test set zero-shot results.** Performance of ZeroNER, LLMs, and GLiNER models across three benchmarks: OntoNotes-ZS, MedMentions-ZS, and LegalNER-ZS. Results are presented for two settings: using type names and using type descriptions. AVG represents the macro-averaged F1 score across all benchmarks. For each category, the best-performing model is highlighted in bold, and the second-best is underlined.

VRAM. The model was trained on the distillation dataset for 3 epochs. Each epoch took approximately 18 hours to complete, amounting to a total training time of 54 hours. At each epoch, we saved the best checkpoint according to the dev set of our benchmarks. We used a batch size of 8 and a constant learning rate of 2e-5, the latter selected following manual hyperparameter tuning. The entity masking probability was set to 0.3, following Aly et al. (2021). We leave 42 as the default training seed. For LLM inference, we leveraged the VLLM library[6] (version 0.6.3.post1). More details about both ZeroNER training and LLM inference are discussed in §F.

5 Benchmark Contamination

To assess potential data contamination between training corpora of the considered GLiNER baselines and the benchmarks contemplated in this study, we implement a multi-tiered detection approach with increasing levels of semantic overlap tolerance. We first identify *exact matches* where entity type labels appear identically in both datasets. *Fuzzy matches* leverage Python's `difflib.SequenceMatcher` to detect semantically similar labels (e.g., *organization* vs. *organisation*) using a similarity threshold of 0.9, capturing morphological variations and minor spelling differences. *Word boundary matches* identify cases where entity types share common semantic components (e.g., *immigration judge* overlapping with *judge*), indicating conceptual overlap. Finally, *substring matches* detect direct containment relationships between labels, revealing hierarchical contamination patterns. Each contamination type receives a severity weight (exact: 1.0, fuzzy: 0.8, word boundary: 0.6, substring: 0.4) to compute an overall contamination severity score, ensuring entities are categorized only once to prevent double-counting. Table 4 lists contamination metrics for both `pile-mistral-v0.1` and `Pile-NER-type`, the training datasets of GLiNER-v2.1 and GLiNER-v1 models, respectively. Under pile-mistral-v0.1, all three benchmarks exhibit 100% entity-type contamination and high severity scores (LegalNER: 90–100%; OntoNotes: 90–93.3%; MedMentions: 76–92%), confirming that `pile-mistral`'s raw corpus already contains most benchmark labels. Instead, `Pile-NER-type`, although still showing full

[6] github.com/vllm-project/vllm

Benchmark	Split	# Types	# Contaminated	Cont. Rate	Severity	Exact	Fuzzy / Word / Sub
			`pile-mistral-v0.1`				
OntoNotes-ZS	Dev	4	4	100.00%	90.00%	3.0	0.0 / 0.6 / 0.0
OntoNotes-ZS	Test	3	3	100.00%	93.33%	2.0	0.8 / 0.0 / 0.0
MedMentions-ZS	Dev	5	5	100.00%	76.00%	2.0	0.0 / 1.8 / 0.0
MedMentions-ZS	Test	5	5	100.00%	92.00%	4.0	0.0 / 0.6 / 0.0
LegalNER-ZS	Dev	3	3	100.00%	100.00%	3.0	0.0 / 0.0 / 0.0
LegalNER-ZS	Test	4	4	100.00%	90.00%	3.0	0.0 / 0.6 / 0.0
			`Pile-NER-type`				
OntoNotes-ZS	Dev	4	4	100.00%	90.00%	3.0	0.0 / 0.6 / 0.0
OntoNotes-ZS	Test	3	3	100.00%	93.33%	2.0	0.8 / 0.0 / 0.0
MedMentions-ZS	Dev	5	5	100.00%	64.00%	1.0	0.0 / 1.8 / 0.4
MedMentions-ZS	Test	5	5	100.00%	80.00%	2.0	0.8 / 1.2 / 0.0
LegalNER-ZS	Dev	3	3	100.00%	93.33%	2.0	0.8 / 0.0 / 0.0
LegalNER-ZS	Test	4	2	50.00%	40.00%	1.0	0.0 / 0.6 / 0.0

Table 4: **GLiNER contamination metrics.** Overlap analysis between the entity types in LegalNER-ZS, OntoNotes-ZS, and MedMentions-ZS (dev/test) and the GLiNER training corpora: `pile-mistral-v0.1` and `Pile-NER-type`.

type overlap, demonstrates lower severity on MedMentions (64–80%) and markedly reduced contamination on LegalNER's test split (only 50% of types, severity 40%).

6 Results and Discussion

Our core results are detailed in Table 3. In this section, we present the evaluation of ZeroNER against various baseline models and discuss key insights. We refer the reader to §H for data cleaning ablations and §G for efficiency considerations.

ZeroNER versus SMXM. Compared to SMXM (Aly et al., 2021), our method achieves substantially higher results, although it relies on a smaller BERT backbone and does not utilize in-domain supervised training. ZeroNER registers a 100% increase in average performance. This gap highlights the advantage of using distilled annotations and entity descriptions, even when the training data distribution differs from the test benchmarks, as well as the strength of silver data in the absence of direct supervision.

ZeroNER versus LLMs. Despite their extensive parameter space and recent advancements, LLMs continue to struggle in zero-shot NER. ZeroNER consistently surpasses all LLMs of varying sizes, with performance gains ranging from 4% to 16% relative to the strongest baseline, Qwen2.5.

ZeroNER versus GLiNER. Notably, ZeroNER outperforms 5 out of 6 evaluated GLiNER models. It reliably achieves F1 score improvements of up to 10% across all model sizes of the v2.1 version, and also beats the base and medium configurations of the v1 version. Figure 5 provides a per-type comparison between ZeroNER and the three contaminated variants of GLiNER-v2.1 on both dev and test splits. ZeroNER exhibits stronger performance on broad, context-sensitive categories such as *FAC*, *LOC*, *GPE LAW*, and *NORP*, where natural language descriptions facilitate more accurate disambiguation of in-domain usage. Conversely, GLiNER yields higher scores on heavily contaminated labels—most notably *ORG*. This advantage is attributed to the training on `pile-mistral-v0.1` corpus, which includes ∼20K annotations labeled as *Organization* or close variants (e.g., *political organization*, *religious organization*), establishing pattern-label associations. We further observe that ZeroNER's performance on certain categories—such as *Judge* and *Statute*—can deteriorate when descriptions become overly detailed. For example, the most effective description for JUDGE is a concise formulation like *"The name of the judge presiding over the current case."*

Do LLMs need descriptions? LLMs, even with their extensive pre-trained knowledge, face challenges in generalizing when relying solely on entity type names. The LLaMA family is the only evaluated model group that has limited benefit from incorporating entity descriptions. Other LLMs display clear F1 growth when such descriptions are provided, with Qwen 2.5 securing an overall increase of 8%. As noted by Aly et al. (2021), OntoNotes guidelines are highly specific and representative of the targets. This is evident in the

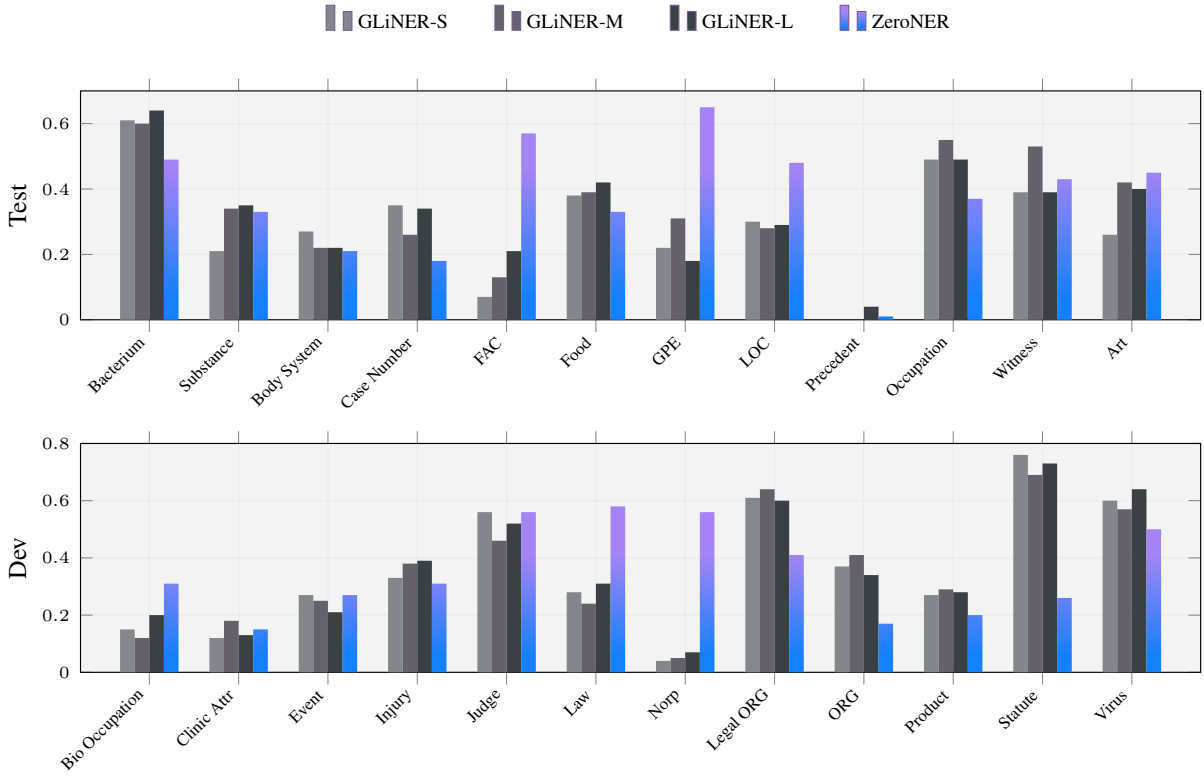

Figure 5: **Impact of entity type contamination.** Per-type F1 scores comparing ZeroNER and the contaminated variants of GLiNER-v2.1 on both development and test splits.

OntoNotes-ZS results, where all LLMs—including LLaMA—demonstrate improved performance, as plotted in Figure 6. These observations suggest that some models require a certain level of description quality to benefit from additional context, whereas models like Qwen 2.5 appear more capable of leveraging such information for zero-shot adaptation.

7 Conclusions

We present ZeroNER, a knowledge distillation framework specifically designed for hard zero-shot NER. After rigorous evaluation on heterogeneous datasets, ZeroNER proves capable of recognizing unseen entities using only a textual description of them, eliminating the need for re-training. Regardless of model scale, ZeroNER delivers stronger results than all considered LLMs, combining accuracy with computational efficiency. In comparison to GLiNER, which relies on embedding-level alignment between input sentence and class name tokens, ZeroNER retains a pronounced advantage, even in the absence of target contamination. Moreover, our findings indicate that LLMs display meaningful generalization potential when supplied with well-crafted descriptions, opening the door to future work on test-time scaling.

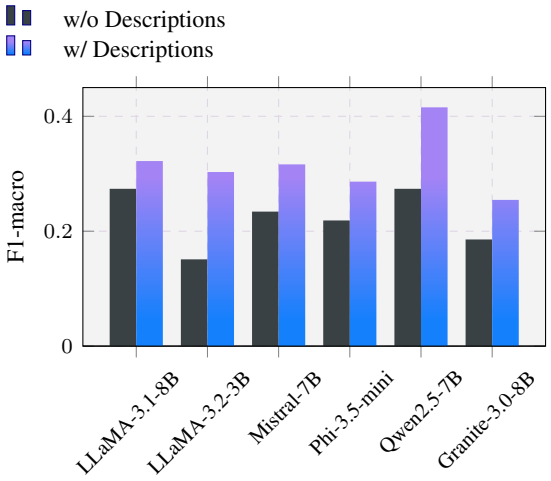

Figure 6: **Impact of entity type descriptions on macro-F1 scores in LLMs.** Comparison of LLMs on the OntoNotes-ZS test set with and without the incorporation of descriptions for the target classes in the prompt.

Limitations and Future Work

ZeroNER, though effective, comes with certain inherent limitations that warrant discussion. Compared to GLiNER, it currently lacks support for nested entities. The sensitivity of the ZeroNER to the quality of descriptions presents another con-

sideration. Poorly constructed definitions can lead to degraded results, and adapting ZeroNER to specific benchmarks may require manual refinement. In addition, model performance is likely influenced by the choice of encoder. We selected BERT for fair comparison with the SMXM baseline, but stronger architectures such as RoBERTa, DeBERTa, or ModernBERT may offer further improvements—a direction we leave open for future exploration, along with multi-linguality. Looking ahead, future work may also explore techniques inspired by soft prompting (Ramnath et al., 2025) to automatically learn entity descriptions that maximize the likelihood of successful recognition by LLMs. More generally, automating the generation of high-quality type descriptions represents a promising direction for future research, as illustrated by recent work such as Picco et al. (2024). Additionally, our insights suggest designing hybrid training strategies that alternate between using the type name and the type description. Such an approach could promote more adaptive behavior, favoring minimal definitions for well-represented categories and switching to richer descriptions for ambiguous or underrepresented types. From an interpretability standpoint, it would also be valuable to investigate which tokens within a description contribute most to performance. Techniques such as input perturbation or attribution methods—already applied in other fields (Domeniconi et al., 2014a)—could be adapted to analyze token-level influence. Beyond NER, ZeroNER may be extended to other low-resource tasks. Three promising directions involve applying it to cross-domain document classification (Domeniconi et al., 2014b, 2016, 2017; Moro et al., 2018), node classification and graph clustering (Lodi et al., 2010), and semantic parsing (Frisoni et al., 2021, 2022).

Acknowledgments

Research partially supported by AI-PACT project (CUP B47H22004450008, B47H22004460001); National Plan PNC-I.1 DARE initiative (PNC0000002, CUP B53C22006450001); PNRR Extended Partnership FAIR (PE00000013, Spoke 8); 2024 Scientific Research and High Technology Program, project "AI analysis for risk assessment of empty lymph nodes in endometrial cancer surgery," the Fondazione Cassa di Risparmio in Bologna; Chips JU TRISTAN project (G.A. 101095947).

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, and Ahmed Awadallah. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, et al. 2022. Large language models are few-shot clinical information extractors. In *EMNLP*, pages 1998–2022. ACL.

Rami Aly, Andreas Vlachos, and Ryan McDonald. 2021. Leveraging type descriptions for zero-shot named entity recognition and classification. In *ACL/IJCNLP*, pages 1516–1528, Online. ACL.

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

Cristian Bucila, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *KDD*, pages 535–541. ACM.

Jean-Flavien Bussotti, Luca Ragazzi, Giacomo Frisoni, Gianluca Moro, and Paolo Papotti. 2024. Unknown claims: Generation of fact-checking training examples from unstructured and structured data. In *EMNLP*, pages 12105–12122. ACL.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, et al. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Alessio Cocchieri, Giacomo Frisoni, Marcos Martínez Galindo, Gianluca Moro, Giuseppe Tagliavini, and Francesco Candoli. 2025. Openbioner: Lightweight open-domain biomedical named entity recognition through entity type description. In *NAACL*, pages 818–837. ACL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186. ACL.

Yuyang Ding, Juntao Li, Pinzheng Wang, Zecheng Tang, et al. 2024. Rethinking negative instances for generative named entity recognition. In *ACL*, pages 3461–3475. ACL.

Giacomo Domeniconi, Marco Masseroli, Gianluca Moro, and Pietro Pinoli. 2014a. Discovering new gene functionalities from random perturbations of known gene ontological annotations. In *KDIR*, pages 107–116. SciTePress.

Giacomo Domeniconi, Gianluca Moro, Andrea Pagliarani, and Roberto Pasolini. 2017. On deep learning in cross-domain sentiment classification. In *IC3K*, pages 50–60. SciTePress.

Giacomo Domeniconi, Gianluca Moro, Roberto Pasolini, and Claudio Sartori. 2014b. Cross-domain text classification through iterative refining of target

categories representations. In *KDIR*, pages 31–42. SciTePress.

Giacomo Domeniconi, Konstantinos Semertzidis, Vanessa López, Elizabeth M. Daly, Spyros Kotoulas, and Gianluca Moro. 2016. A novel method for unsupervised and supervised conversational message thread detection. In *DATA*, pages 43–54. SciTePress.

Giacomo Frisoni, Gianluca Moro, and Lorenzo Balzani. 2022. Text-to-text extraction and verbalization of biomedical event graphs. In *COLING*, pages 2692–2710. ICCL.

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.

Abbas Ghaddar and Philippe Langlais. 2018. Robust lexical features for improved neural network named-entity recognition. In *COLING*, pages 1896–1907. ACL.

Granite Team, IBM. 2024. Granite 3.0 language models.

Aaron Grattafiori, Abhimanyu Dubey, and Abhinav Jauhri. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *ICLR*. OpenReview.net.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, et al. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *ACL (Findings)*, pages 8003–8017. ACL.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, et al. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Prathamesh Kalamkar, Astha Agarwal, Aman Tiwari, Smita Gupta, et al. 2022. Named entity recognition in Indian court judgments. In *NLLP*, pages 184–193, Abu Dhabi, United Arab Emirates (Hybrid). ACL.

Peng-Hsuan Li, Ruo-Ping Dong, Yu-Siang Wang, Ju-Chieh Chou, et al. 2017. Leveraging linguistic structures for named entity recognition with bidirectional recursive neural networks. In *EMNLP*, pages 2664–2669, Copenhagen, Denmark. ACL.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, et al. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Stefano Lodi, Gianluca Moro, and Claudio Sartori. 2010. Distributed data clustering in multi-dimensional peer-to-peer networks. In *ADC*, volume 104 of *CRPIT*, pages 171–178. Australian Computer Society.

Sunil Mohan and Donghui Li. 2019. Medmentions: A large biomedical corpus annotated with UMLS concepts. In *AKBC*.

Gianluca Moro, Andrea Pagliarani, Roberto Pasolini, and Claudio Sartori. 2018. Cross-domain & in-domain sentiment analysis with memory-based deep neural networks. In *IC3K*, pages 125–136. SciTePress.

Hoang Van Nguyen, Francesco Gelli, and Soujanya Poria. 2021. DOZEN: cross-domain zero shot named entity recognition with knowledge graph. In *SIGIR*, pages 1642–1646. ACM.

Rasha Obeidat, Xiaoli Fern, Hamed Shahbazi, and Prasad Tadepalli. 2019. Description-based zero-shot fine-grained entity typing. In *NAACL*, pages 807–814, Minneapolis, Minnesota. ACL.

OpenAI. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Gabriele Picco, Leopold Fuchs, Marcos Martínez Galindo, Alberto Purpura, et al. 2024. Description boosting for zero-shot entity and relation classification. In *ACL*, pages 9441–9457. ACL.

Gabriele Picco, Marcos Martínez Galindo, Alberto Purpura, Leopold Fuchs, et al. 2023. Zshot: An open-source framework for zero-shot named entity recognition and relation extraction. In *ACL*, pages 357–368. ACL.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, et al. 2013. Towards robust linguistic analysis using ontonotes. In *CoNLL*, pages 143–152. ACL.

Kiran Ramnath, Kang Zhou, Sheng Guan, Soumya Smruti Mishra, et al. 2025. A systematic survey of automatic prompt optimization techniques. *CoRR*, abs/2502.16923.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, et al. 2024. Gollie: Annotation guidelines improve zero-shot information-extraction. In *ICLR*. OpenReview.net.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, et al. 2023. Instructuie: Multi-task instruction tuning for unified information extraction. *CoRR*, abs/2304.08085.

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.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, et al. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Andrew Zamai, Andrea Zugarini, Leonardo Rigutini, Marco Ernandes, et al. 2024. Show less, instruct more: Enriching prompts with definitions and guidelines for zero-shot NER. *CoRR*, abs/2407.01272.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2023. Gliner: Generalist model for named entity recognition using bidirectional transformer. *CoRR*, abs/2311.08526.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, et al. 2023. Universalner: Targeted distillation from large language models for open named entity recognition. *CoRR*, abs/2308.03279.

A Prompts

Figure 7 outlines the prompt used to generate annotations over the Pile. Figure 8 shows the prompt used to employ self-correction over annotations generated by the teacher LLM. Figure 9 shows the prompt used to generate multi-view descriptions. Figure 10 shows the prompt used to make inference with LLMs on OntoNotes-ZS. Table 5 documents some examples of multi-view descriptions generated by the teacher LLM.

B Benchmark Dataset Details

- **OntoNotes 5.0.** Multi-genre dataset designed for domain-general entity types. In particular, OntoNotes offers annotations in three genres: newswire, broadcast conversation, and web data. Various iterations of the OntoNotes dataset have found application in academic papers. Our version of OntoNotes aligns with that employed in (Li et al., 2017; Ghaddar and Langlais, 2018). Removed irrelevant entity classes: *Money, Ordinal, Percent*.
- **MedMentions.** The dataset contains not just named entities but also high-level concepts. For example, in the passage *"modeling nurse-patients,"* the term *"modeling"* is annotated with the concept *Research Activity*, thus adding an extra layer of complexity to the NER task. Original GitHub repository.[7] Removed irrelevant entity classes: / (i.e., no trivial label identified).
- **LegalNER.** We considered the dataset officially released by the authors,[8] which comprises only the train and dev sets for a total of 12,069 labeled sentences. Entities are sourced from both the preamble and the main body of judgments. The preamble includes structured metadata such as party names, judge and lawyer identities, dates, and court details. Removed irrelevant entity classes: *Date*.

Class distributions after zero-shot adaption—divided by train, dev, and test sets—are reported in Figure 11.

C LegalNER-ZS Construction

Unlike OntoNotes-ZS and MedMentions-ZS, which use the disjoint splits from Aly et al. (2021), LegalNER lacks predefined zero-shot splits. We

[7] github.com/chanzuckerberg/MedMentions
[8] github.com/Legal-NLP-EkStep/legal_NER

therefore construct LegalNER-ZS manually following the same principles. The original dataset consists of two variants (PREAMBLE and JUDGEMENT), each providing only train and dev sets. We merge these to form a unified dataset for split construction.

Our procedure involves:

1. **Entity stratification:** We divide entity types by frequency into disjoint sets: *train* (COURT, PETITIONER, RESPONDENT, LAWYER, PROVISION, OTHER_PERSON), *dev* (JUDGE, ORG, STATUTE), and *test* (GPE, PRECEDENT, CASE_NUMBER, WITNESS).
2. **Trivial type removal:** DATE is excluded from validation due to its trivial nature, which could distort model performance.
3. **Strict partitioning:** We retain only documents whose entities belong entirely to one set (train, dev, or test). Mixed-entity documents are discarded to ensure no entity type overlaps across splits.
4. **Preprocessing:** Annotations are converted to IOB2 format using NLTK tokenization with span alignment. We discard samples with HTML markup or without valid entities from the split-specific type set.

This results in a clean zero-shot setup where dev and test sets contain entirely unseen entity types, ensuring robust generalization assessment.

D Pile-NER-type Contamination

Table 6 shows the overlap between Pile-NER-type (pretraining dataset for GLiNER and UniversalNER) and the benchmark entity types used for zero-shot evaluation of such models. This intersection undermines the validity of the "zero-shot" claims in the original papers. It confirms that standard zero-shot NER evaluations using Pile-NER-type are heavily confounded by label contamination.

E Further GLiNER Results

While all results in the main paper use lower-case type names for GLiNER inference, the original authors also recommend testing with title-case labels, since the model can be sensitive to casing. Therefore, in Table 7 we report test set performance even in this setting. As shown, the GLiNER-v1 models are largely unaffected by casing, whereas the GLiNER-v2.1 variants register a modest increase in overall performance. Overall, ZeroNER remains the best-performing model, on par with GLiNER-L-v1.

F Hyperparameters and Implementation Details

Table 8 lists all hyperparameters examined in the inference and fine-tuning phases, highlighting the final values. The parameters of the linear classification layer ω^T have been randomly initialized from a uniform distribution $U(-\sqrt{b}, \sqrt{b})$ with $b = \frac{1}{\text{in-features}}$. To improve the evaluation of the semantic comprehension capabilities of LLMs, we systematically converted labels that included abbreviations or specific formats into their natural language equivalents. For example, we add a short explanatioon to tell the model thta "LOC" corresponds to "LOCATION", while "FAC" to "FACILITY" (see Figure 10). Span-level scores are computed with the `seqeval` library.[9]

G Inference Efficiency

Table 9 reports the inference-time comparison between ZeroNER and the three GLiNER-v2.1 baselines on OntoNotes-ZS (3 types), MedMentions-ZS (5 types), and LegalNER-ZS (4 types). ZeroNER consistently achieves lower total inference time and higher throughput (samples/s), with a per-sample latency as low as 0.01 s. For instance, on OntoNotes-ZS, ZeroNER processes over 82 samples/s—more than 4× faster than GLiNER-L.

However, because ZeroNER's cross-encoder pairs each input sentence with every type description, inference cost inevitably grows as the number of types increases (i.e., longer cumulative descriptions) and as input text length grows (i.e., more tokens to encode per pair). This effect is visible on LegalNER-ZS: although ZeroNER still achieves competitive throughput (68.51 samples/s) and matches the lowest latency (0.01 s), its total time (17.91 s) is higher than on MedMentions-ZS. LegalNER-ZS sentences are substantially longer than those in the other benchmarks, so each sentence–description pair incurs more computation. These results underscore that, while ZeroNER is very efficient in practice, inference costs scale with both the number of target types and the length of the input text.

[9] https://github.com/chakki-works/seqeval

H Self-Correction

Table 10 illustrates the impact of our self-correction pipeline on model performance. By identifying and correcting noisy annotations using the model's own predictions, we obtain a higher quality dataset that yields superior results across all benchmarks. The filtered dataset demonstrates consistent improvements, with F1 scores increasing from 0.47 to 0.50 on OntoNotes-ZS , from 0.33 to 0.35 on MedMentions-ZS, and from 0.30 to 0.32 on LegalNER-ZS . The overall macro-averaged F1 score improves from 0.37 to 0.39, confirming that annotation quality is more critical than quantity for zero-shot NER performance. These results validate our hypothesis that self-correction mechanisms can effectively identify and remediate systematic annotation errors, leading to more robust model training and better generalization to unseen entity types.

I Manual Descriptions

Table 11, Table 12, and Table 13 enumerate all the entity-type descriptions for knowledge fine-tuning in MedMentions-ZS, OntoNotes-ZS, and LegalNER-ZS.

Prompt for Pile Entity Annotation

Annotate the given sentence with entity mentions and their corresponding labels. Then, define the most appropriate domain for the sentence.

Requirements:

Report annotations as a list of tuples in the following format: [(mention, label), ...]

Each mention must include:
- The exact string as it appears in the text.
- The corresponding entity label.

Ensure that:
- Labels do not overlap.
- Each entity is assigned to only one label.

Output following this schema:
Annotations: [list of tuples]
Domain: domain of the sentence

Now, consider the following sentence.

Sentence: {{Sentence}}

Figure 7: **Prompt #1.** Prompt used to annotate sentences on the Pile.

Prompt for Self-Correction

You are an expert evaluator.
Your task is to evaluate the quality of named entity annotations provided as input by the user, according to the following criteria.

Criterion 1: Completeness of annotations
3 points: All entities present in the sentence are annotated, with nothing missed.
2 points: Most entities are annotated, but one or two entities are missing.
1 point: Several entities are missing from the annotation.
0 points: The annotation is incomplete, with most entities not recognized.

Criterion 2: Correctness of entity labels
3 points: All entities are labeled with the correct entity type (e.g., PERSON, ORGANIZATION).
2 points: Most entities are labeled correctly, but there are one or two minor errors.
1 point: Some entities are correctly labeled, but there are several incorrect labels.
0 points: Entity labels are mostly incorrect, with a significant number of entities mislabeled.

Return you answer in the following output format, allowing for easy parsing:
explanation: (your explanation for the assessment)
score: {"completeness" : a score from 0 to 3, "correctness": a score from 0 to 3}

Figure 8: **Prompt #2.** Prompt used to self-correct predictions in terms of correctness and completeness.

Prompt for Multi-View Descriptions

Provide a short description for the class {{Class}} for each of the following domains: {{Domain1, Domain2, ...}}. Include examples. Use a maxium of 100 words.

Output your answer in the following schema:

descriptions:
domain 1: ...
domain 2: ...
...

Don't add futher information.

Figure 9: **Prompt #3.** Prompt used to generate multi-view descriptions.

Class	Domains	Multi-View Description
Animal	Veterinary Science Neuroscience	Animals are organisms that are not plants. Animals have nervous systems, and are capable of movement. They are multicellular organisms that are not plants. Animal is a living thing with a central nervous system and voluntary muscles. It is a member of the kingdom Animalia, a group of organisms that includes all animals, as well as some single-celled organisms such as amoebas and paramecia.
Scientific Work	Medicine Biochemistry	Scientific work is a systematic process of collecting, analyzing and interpreting data to test a hypothesis and to generate new knowledge. Scientists use scientific methods to investigate the natural world. The methods are designed to be repeatable, so that the results of one investigation can be verified by other scientists. Scientists use these methods to test hypotheses, which are predictions about how nature works.
Cell Line	Biochemistry Cell Biology	A cell line is a cell culture that is derived from a single cell and has undergone a number of cell divisions in vitro. Cell lines are used in a variety of ways including the study of disease, drug discovery, and the study of cellular processes. A cell line is a population of cells that have been grown in culture for an extended period of time. Cell lines are derived from primary cells, which are cells that have been isolated from a tissue sample and are still alive.
Chemical	Biochemistry Toxicology	A chemical is a substance that has the ability to react with other substances to form new substances. The reaction can be a chemical change, a physical change, or a combination of both. Chemical is a substance with a specific chemical composition and a definite chemical structure.

Table 5: **Qualitative examples of entity type multi-view descriptions.**

Prompt for Zero-Shot NER

System: You are an expert entity classifier. You have to identify the entities within the given sentence that belong to one of the specified labels. There are no overlapped entities, so each word can belong to only one label. You can use only the possible labels provided. No other label is allowed.

Provide output in the following format: Return a list, marked with square brackets '[' and ']', containing string tuples. Each tuple should follow the pattern: ("entity", "label").
Prefix the entire list with '### entities:'. For example: ### entities: [("entity 1", "label of entity 1"), ("entity 2", "label of entity 2"), ...]

If no entities are found, return an empty list like this: ### entities: []
Don't add further information.

— Standard —

User:
Possible labels: ['FAC', 'LOC', 'WORK OF ART']

Sentence: The station called me at noon and said something happened at Jingguang Bridge and that I had to go to the station immediately to research the upcoming program .

Consider that FAC corresponds to Facility and LOC corresponds to Location.

— w/ Descriptions —

User:
Possible labels:
FAC: *Names of man-made structures: infrastructure (streets, bridges), buildings, monuments, etc. belong to this type. Buildings that are [...]*
LOC: *Names of geographical locations other than GPEs. These include [...]*
WORK OF ART: *Titles of books, songs, television programs and other creations. Also includes awards. These are [...]*

Sentence: The station called me at noon and said something happened at Jingguang Bridge and that I had to go to the station immediately to research the upcoming program .

Figure 10: **Prompt #4.** Prompt adopted for zero-shot NER inference with LLMs. The example refers to a test sentence sourced from OntoNotes-ZS.

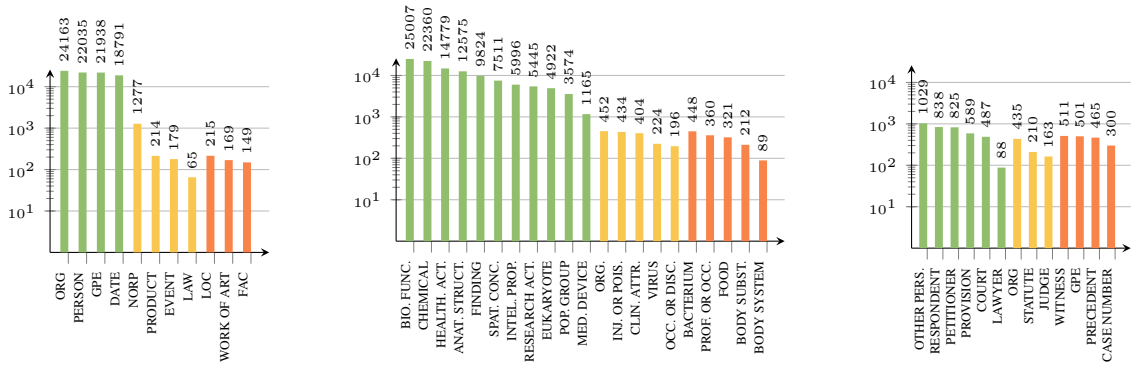

Figure 11: **Entity type frequency distribution in the considered benchmarks.** Class occurrences in OntoNotes-ZS, MedMentions-ZS, and LegalNER-ZS (green=train, orange=dev, red=test).

Dataset	Overlaps (dataset type, Pile-NER type)
ACE 2005	(Facility, Medical facility),(Organization, Person or organization),(Weapon, Weapon system),(Geographical_social_political, Political), (Vehicle, Vehicles)
Broad Tweet Corpus	(Organization, Person or organization)
CoNLL 2003	(Organization, Person or organization)
MultiNERD	(Disease, Disease or medical condition),(Food, Food source),(Time, Time period),(Organization, Person or organization),(Media, Print media),(Instrument, Musical instrument),(Geographical_phenomenon, Phenomenon),(Biology, Molecular biology),(Plant, Type of plant), (Event, Sports event), (Vehicle, Vehicles)
Ontonotes	(Facility, Medical facility),(Quantity, Physical quantity),(Date, Date format),(Organization, Person or organization),(Ordinal, Ordinal number),(Geographical_social_political, Political),(Time, Time period),(Law, Law or bill),(Work_of_art, Work),(Percent, Percentage),(Cardinal, Cardinal number),(Location, Anatomical location),(Product, Product id),(Language, Programming language),(National_religious_political, Political),(Person, Personal information),(Event, Sports event),(Money, Amount of money)
PolyglotNER	(Organization, Person or organization)
TweetNER7	(Corporation, Type of corporation),(Group, Musicgroup), (Creative_work, Creative_work_element),(Event, Sports event), (Product, Product id)
WikiANN en	(Organization, Person or organization)
WikiNeural	(Organization, Person or organization)
AnatEM	(Anatomy, Brain anatomy)
bc2gm	(Gene, Gene/protein)
bc4chemd	(Chemical, Chemical compound)
bc5cdr	(Chemical, Chemical compound),(Disease, Disease or medical condition)
CrossNER_AI	(Conference, Sports conference),(University, College/university),(Researcher, Research),(Organization, Person or organization),(Task, Behavioral task),(Product, Product id),(Country, De facto independent country),(Algorithm, Query evaluation algorithm)
CrossNER_literature	(Magazine, Manga magazine),(Poem, Greek epic poem), (Book, Book series),(Organization, Person or organization),(Award, Award show), (Event, Sports event),(Country, De facto independent country),(Writer, Singersongwriter)
CrossNER_music	(Organisation, Organisations),(Album, Studio album),(Award, Award show),(Organization, Person or organization), (Band, Bandleader),(Event, Sports event),(Song, Song title),(Musical_artist, Musical), (Misc, Miscellaneous),(Person, Personal information), (Musical_instrument, Instrument),(Country, De facto independent country)
CrossNER_politics	(Politician, Georgian politician),(Organization, Person or organization),(Event, Sports event),(Country, De facto independent country),(Political_party, Political)
CrossNER_science	(Theory, Level of theory),(University, College/university),(Chemical_compound, Chemical),(Organization, Person or organization),(Award, Award show),(Scientist, Neuroscientist),(Academic_journal, Journal),(Chemical_element, Chemical),(Event, Sports event),(Discipline, Academic discipline),(Enzyme, Enzyme phase),(Protein, Gene/protein),(Country, De facto independent country)
FabNER	(Manufacturing_standard, Standard),(Concept_or_principle, Concept),(Machine_or_equipment, Equipment),(Enabling_technology, Technology),(Material, Construction material),(Biomedical, Medical),(Process_parameter, Parameter),(Mechanical_property, Property),(Engineering_feature, Feature),(Manufacturing_process, Process),(Application, Software application)
FindVehicle	(Orientation_of_vehicle, Vehicle), (Truck, Truck),(Motorcycle, Motor),(Color_of_vehicle, Vehicle),(Position_of_vehicle, Position), (Vehicle_type, Vehicle),(Vintage_car, Car), (Suv, Car),(Mpv, Car),(Recharge, Charge),(Vehicle_range, Vehicle),(Vehicle_model, Vehicle),(Sport, Sports team)
GENIA_NER	(Dna, Dna sequence), (Cell_line, Cell_line),(Protein, Gene/protein)
HarveyNER	(Exact_location, Location),(Area, Type of area)
mitmovie	(Title, Job title),(Average_ratings, Average),(Genre, Musicgenre),(Review, Type of review),(Song, Song title),(Year, Year founded)
mitrestaurant	(Amenity, Amenity),(Restaurant_name, Restaurant),(Hours, Hours),(Price, Price range)
ncbi	(Disease, Disease or medical condition)

Table 6: **Overlap of entity types between Pile-NER and the benchmarks used by GLiNER and UniversalNER.**

		OntoNotes-ZS			MedMentions-ZS			LegalNER-ZS			
Model	**Size**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**AVG**
GLiNER_S-v1	166M	0.44	0.34	0.35	0.49	0.33	0.40	0.47	0.26	0.33	0.36
GLiNER_M-v1	209M	0.41	0.33	0.34	0.53	0.41	0.46	0.45	0.27	0.33	0.38
GLiNER_L-v1	459M	0.44	0.35	0.37	0.52	0.45	0.48	0.42	0.32	0.32	0.39
GLiNER_S-v2.1	166M	0.28	0.31	0.24	0.40	0.38	0.39	0.34	0.31	0.26	0.30
GLiNER_M-v2.1	209M	0.32	0.34	0.29	0.43	0.38	0.39	0.36	0.25	0.26	0.31
GLiNER_L-v2.1	459M	0.30	0.40	0.29	0.44	0.43	0.43	0.36	0.24	0.27	0.33
ZeroNER (Ours)	110M	0.61	0.44	0.50	0.32	0.42	0.35	0.43	0.26	0.32	0.39

Table 7: **Zero-shot performance comparison between ZeroNER and title-cased GLiNER baselines.**

Hyperparameter	Search space
LLM inference (generation, evaluation)	
decoding_method	"greedy"
max_new_tokens	1024
temperature	0.0
top_p	1.0
ZeroNER pre-training	
min_sample_class	25
max_sample_class	20
max_description_length	150
max_sequence_length	300
mask_probability	0.3
linear_dropout	0.5
linear_units_symbol	100
adam_epsilon	1e-8
max_grad_norm	1.0
lr	{2e-5∗, 4e-6, 7e-6}
epochs	3
batch	8
SMXM fine-tuning	
### MedMentions-ZS and LegalNER-ZS	
max_description_length	100
max_sequence_length	200
### OntoNotes-ZS	
max_description_length	150
max_sequence_length	300
### All	
mask_probability	{0.3∗, 0.5}
linear_dropout	0.5
linear_units_symbol	100
adam_epsilon	1e-8
max_grad_norm	1.0
lr	4e-6
epochs	3
batch	2
val_steps	1

Table 8: **Explored hyperparameters along with their empirical search grid.** ∗ marks the final picked values.

Model	Total time (s)	Samples/s	Latency (s)
`OntoNotes-ZS (3 types)`			
gliner_large-v2.1	20.27	21.02	0.05
gliner_medium-v2.1	11.09	38.43	0.03
gliner_small-v2.1	8.31	51.27	0.02
ZeroNER (ours)	5.18	82.23	0.01
`MedMentions-ZS (5 types)`			
gliner_large-v2.1	37.02	28.31	0.04
gliner_medium-v2.1	20.33	51.55	0.02
gliner_small-v2.1	13.90	75.41	0.01
ZeroNER (ours)	13.60	77.07	0.01
`LegalNER-ZS (4 types)`			
gliner_large-v2.1	41.58	29.51	0.03
gliner_medium-v2.1	23.12	53.08	0.02
gliner_small-v2.1	15.84	77.46	0.01
ZeroNER (ours)	17.91	68.51	0.01

Table 9: **Inference time comparison between ZeroNER and GLiNER baselines.** Test set.

		OntoNotes-ZS			MedMentions-ZS			LegalNER-ZS			
Model	**Size**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**AVG**
ZeroNER	110M	0.61	0.44	0.50	0.32	0.42	0.35	0.43	0.26	0.32	0.39
ZeroNER (unfiltered)	110M	0.58	0.41	0.47	0.29	0.39	0.33	0.40	0.24	0.30	0.37

Table 10: **Impact of data filtering on ZeroNER performance.** Removing noisy annotations with the proposed data cleaning recipe improves F1 scores across all benchmarks.

Type	**Description**
NORP	This type represents adjectival forms of GPE and Location names, as well as adjectival forms of named religions, heritage and political affiliation. Also marked are head words which refer to people using the name of an entity with which they are affiliated, often a GPE or Organization. The distinction between NORP and other types is morphological. "American" and "Americans" are adjectival nationalities, while "America" and "US" are GPEs, regardless of context.
PRODUCT	This can be name of any product, generally a model name or model name and number. Named foods are also included. Credit cards, checking accounts, CDs, and credit plans are NOT marked. References that include manufacturer and product should be marked as two separate named entities, ORG + PRODUCT: [Apple] [iPod], [Dell] [Inspiron], [Ford] [Mustang].
EVENT	Named hurricanes, battles, wars, sports events, attacks. Metonymic mentions (marked with a ~) of the date or location of an event, or of the organization(s) involved, are included.
LAW	Any document that has been made into a law, including named treaties and sections and chapters of named legal documents.
FAC	Names of man-made structures: infrastructure (streets, bridges), buildings, monuments, etc. belong to this type. Buildings that are referred to using the name of the company or organization that uses them should be marked as FAC when they refer to the physical structure of the building itself, usually in a locative way: "I'm reporting live from right outside [Massachusetts General Hospital]"
LOC	Names of geographical locations other than GPEs. These include mountain ranges, coasts, borders, planets, geo-coordinates, bodies of water. Also included in this category are named regions such as the Middle East, areas, neighborhoods, continents and regions of continents. Do NOT mark deictics or other non-proper nouns: here, there, everywhere, etc. As with GPEs, directional modifiers such as "southern" are only marked when they are part of the location name itself.
WORK OF ART	Titles of books, songs, television programs and other creations. Also includes awards. These are usually surrounded by quotation marks in the article (though the quotations are not included in the annotation). Newspaper headlines should only be marked if they are referential. In other words the headline of the article being annotated should not be marked but if in the body of the text here is a reference to an article, then it is markable as a work of art.

Table 11: **OntoNotes-ZS class descriptions.** Source: annotation guidelines, LDC2013T19/OntoNotes-Release-5.0.

Type	Description
CLINICAL ATTRIBUTE	A clinical attribute is a measurable or observable characteristic used in clinical assessments or diagnosis. Examples include markers and biomarkers, visual acuity, [BMI], histological differentiation.
BIOMEDICAL OCCUPATION OR DISCIPLINE	A biomedical occupation or discipline refers to specialized fields or roles in the biomedical sciences or clinical research. Examples include Histopathology, pathology, complementary medicine, family medicine, FM, and specialty.
INJURY OR POISONING	An injury or poisoning refers to any form of physical harm or adverse physiological effect caused by trauma, exposure, or internal dysfunction. Examples include postoperative [AKI] (acute kidney injury) and toxicological conditions.
VIRUS	A virus is a microscopic infectious agent that replicates only inside the living cells of a host organism. Examples include lentiviruses used in gene therapy and pathogens like SARS-CoV-2.
ORGANIZATION	An organization is a structured group or institution involved in healthcare, research, or governance. Examples include government bodies supporting public health initiatives and clinical institutions conducting medical studies.
BACTERIUM	A bacterium refers to a type of microorganism that can exist as a single cell and may cause infections or play a role in various biological processes. Examples include species like Streptococcus pneumoniae and Streptomyces ahygroscopicus. Bacteria can be studied for their role in disease, antibiotic production, and other cellular functions.
BODY SUBSTANCE	A body substance is any material produced by or found within the body, such as blood, serum, saliva, sweat, or gastric acid. Specific examples include serum cytokine levels for immune responses, blood lipids for metabolic studies, and hemolymph glucose for stress responses.
FOOD	A food refers to any substance consumed to provide nutritional support for the body. This includes a wide range of items such as snacks, meat, dairy products, grains like wheat, and edible substances like carbohydrates, proteins, and fats. These items contribute to energy intake, nutrition, and overall diet, and can be metabolized into energy and body tissue by living organisms.
BODY SYSTEM	A body system consists of interconnected organs and tissues working together to carry out essential functions. Examples include the gastrointestinal tract for digestion, the nervous system for sensory and motor control, the hematological system for blood-related functions, and the endocrine system for hormone regulation.
PROFESSIONAL OR OCCUPATIONAL GROUP	A professional or an occupational group refers to groups of individuals who share the same profession, occupation, or role within a specific field. Examples include cardiologists, psychologists, assessors, hospice staff, and volunteers.

Table 12: **MedMentions-ZS class descriptions.** Source: UMLS Metathesaurus, metathesaurus.

Type	Description
JUDGE	The name of the judges from the current case.
ORG	The names of organizations mentioned in the text.
STATUTE	The name of the act or law mentioned in the judgment.
GPE	Names of countries, cities, states, districts, villages, or other geopolitical locations mentioned in legal judgments. Examples include [Gujarat], [Basarh], [West Pakistan], [India], [Gaya], [Ranchi], and [England].
PRECEDENT	Complete case citations that include party names and legal references used as precedents in judgments. The entire citation should be marked as one entity: [United India Insurance Co. Ltd. v. Rajendra Singh], [State of Rajasthan v. Rajendra Singh, (2009) 11 SCC 106 : (AIR 1998 SC 2554)], [H.R. Sugar Factory (P) Ltd. 187 ITR 363].
CASE NUMBER	Standalone case numbers or legal citations mentioned without party names. Examples include [(1962) 45 ITR 210 (SC)] and [Writ Petition No. 1177 of 1974].
WITNESS	Names of individuals who testified as witnesses in the current legal proceedings. Examples include [Chandregowda], [Seth], [Saudan Singh], and [Rajesh Kumar Srivastava].

Table 13: **LegalNER-ZS class descriptions.** Source: annotation guidelines, Legal-NLP-EkStep/legal_NER.