PROFINER: PROGRESSIVE ENTITY CALIBRATION AND CONFLICT-AWARE INFERENCE FOR ZERO-SHOT NAMED ENTITY RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Zero-shot named entity recognition without task-specific annotations provides a flexible solution for adapting entity extraction to new domains and label spaces. However, direct prompting with large language models often suffers from incomplete entity discovery, inaccurate boundaries, type confusion, and inconsistent predictions among competing candidates. To address these distinct failure modes systematically, we propose ProFiNER, a progressive inference framework for robust zero-shot NER. Specifically, ProFiNER first integrates type-agnostic mention mining with decomposed type-wise question answering and iterative sampling to build a high-recall candidate entity pool. It then selectively refines uncertain entity boundaries and calibrates candidate types through label-wise score normalization and pairwise comparison in an extended label space containing an explicit non-entity category. Finally, structurally incompatible candidates are represented by a conflict graph and jointly resolved through evidence-based adjudication and global selection. Experiments on multiple NER benchmarks with different language model backbones demonstrate that ProFiNER consistently improves entity recognition over direct prompting. Further ablation analyses demonstrate that our type calibration mechanism effectively alleviates semantic confusion, while conflict-aware global inference significantly enhances the structural consistency of the final predictions.

1 INTRODUCTION

Named entity recognition (NER) is a fundamental information extraction task that identifies textual mentions belonging to predefined semantic categories (Yadav & Bethard, 2018). Supervised approaches have achieved remarkable performance by learning from large-scale manually annotated datasets (Yadav & Bethard, 2018; Huang et al., 2021). However, in specialized domains with fine-grained categories or frequently evolving label schemas, obtaining high-quality annotations is costly and time-consuming and often requires substantial domain expertise (Huang et al., 2021; Zheng et al., 2022). Moreover, supervised models trained on fixed domains and label inventories often struggle to generalize directly to new task settings (Zheng et al., 2022). Recently, large language models (LLMs), with their extensive pretrained knowledge, strong instruction-following capabilities, and in-context reasoning abilities, have provided a promising new direction for zero-shot NER without task-specific training (Xie et al., 2023; 2024). Early works introduced LLMs into zero-shot NER by reformulating the task into prompt-based generation and instruction-following (Ashok & Lipton, 2023; Xie et al., 2023; Wang et al., 2025). Subsequent works have made significant strides by leveraging LLM capabilities, including self-improvement and integration of annotation guidelines, to recognize entities from natural language definitions (Xie et al., 2024; Sainz et al., 2024).
However, NER requires both entity boundaries and semantic types to be predicted correctly. These direct LLM-based generation methods typically perform mention detection, boundary identification, and type assignment within a single prediction, leading to challenges in correcting early omissions and boundary errors during subsequent processing. Furthermore, the semantic similarity between fine-grained entity types can lead to confusion, especially when candidates are generated independently and assigned conflicting types. Additionally, factors such as prompt formulation, demonstration selection, and generation randomness can significantly affect the quality of LLM-based

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

extraction (Ashok & Lipton, 2023; Xie et al., 2023; 2024). Such errors are especially difficult to calibrate in zero-shot settings because no labeled target-task data are available. Therefore, the central challenge is how to transform the semantic knowledge of LLMs into NER predictions with accurate boundaries, reliable types, and global structural consistency. Yet, current approaches often focus on addressing only one or two of these aspects in isolation. For example, prompt optimization approaches such as Tong et al. (2025) refine instruction formulation without explicitly handling boundary calibration or fine-grained type disambiguation. Similarly, post-hoc filtering strategies, while useful for refining outputs, lack explicit mechanisms to enforce global structural coherence among entities.

To address these challenges, we propose **ProFiNER**, a progressive inference framework for zero-shot NER. Rather than requiring an LLM to produce the final entity set in a single pass, as in common generation-based formulations (Ashok & Lipton, 2023; Wang et al., 2025), ProFiNER decomposes the recognition process into four coordinated stages. First, **Dual Candidate Mining** combines repeated type-agnostic mention mining with type-wise question answering, improving candidate coverage while preserving plausible boundary variants. Next, **Boundary-Aware Calibration** selectively employs LLM-based assessment to calibrate candidate entity boundaries. Then, **Type-Aware Calibration** evaluates each candidate against the full entity label inventory and an auxiliary non-entity label, followed by pairwise comparisons to distinguish semantically similar entity types. Finally, **Conflict-Aware Selection** constructs a sentence-level candidate conflict graph and integrates candidate evidence, pairwise judgments, and global structural constraints to select a mutually compatible set of typed entities. By decomposing the task into recall-oriented candidate discovery, boundary and type calibration, and precision-oriented global selection, ProFiNER transforms error-prone one-step generation into a progressive decision pipeline that retains, compares, and incrementally refines candidate entities.

Overall, our main contributions include: (1) We propose **ProFiNER**, a progressive inference framework for zero-shot NER. It decomposes one-step entity generation into four coordinated stages: dual candidate mining, boundary-aware calibration, type-aware calibration, and conflict-aware selection. (2) We design a dual-path candidate induction mechanism paired with selective boundary-aware calibration to maximize coverage while retaining plausible entity boundaries for subsequent refinement. (3) We develop label-space type calibration and conflict-aware global selection mechanisms to address semantic label confusion, candidate overlap, and structural inconsistency.

2 Related Work

LLMs for Named Entity Recognition. Existing approaches to LLM-based named entity recognition can be broadly categorized into tuning-based and tuning-free methods. Tuning-based methods improve entity recognition through instruction tuning or other forms of supervised adaptation. For example, Zhou et al. (2024) distill open-domain entity knowledge from ChatGPT-generated annotations; Shen et al. (2023) jointly model entity localization and typing through trainable prompts; and Guo et al. (2025) introduce boundary-aware learning to improve few-shot boundary detection and cross-domain type alignment. In contrast, tuning-free methods keep model parameters frozen and perform entity recognition through inference design. Existing studies explore different output formats and querying strategies, including direct generation (Wang et al., 2025), code-style generation (Li et al., 2023), and label-wise extraction (Xie et al., 2023). Other approaches improve recognition by automatically constructing demonstrations (Xie et al., 2024), iteratively optimizing prompts (Tong et al., 2025), or incorporating retrieval, semantic abstraction, and prediction verification (Mu et al., 2026; Kim & Yoon, 2026). Despite advances in prompt design, extraction strategies, and prediction verification, broader candidate coverage often introduces interdependencies among boundary decisions, type assignments, and conflicts between candidates, making improvements to individual components insufficient to ensure globally consistent predictions. To address this challenge, ProFiNER provides a progressive inference framework that organizes candidate generation, local calibration, and global conflict resolution into interconnected stages to coordinate recognition decisions at different levels.

Candidate Refinement and Structured Inference. The LLM inference strategies discussed above are closely related to span-based NER and structured prediction, which provide methodological foundations for refining intermediate candidates and selecting the final entity set. Early span-based

methods enumerate possible spans and classify them independently (Sohrab & Miwa, 2018). Subsequent studies further explore candidate processing and structured decision-making: Locate-and-Label filters and adjusts span proposals before assigning entity types (Shen et al., 2021); Filtered Semi-Markov CRF prunes unlikely segments before structured decoding (Zaratiana et al., 2023a); and Global Span Selection selects a structurally compatible entity set by maximizing a sentence-level segmentation score (Zaratiana et al., 2022). These studies establish a foundation for combining candidate refinement with global selection. In multi-round LLM extraction, however, connecting these operations also requires updating intermediate predictions: candidates introduced to broaden coverage exhibit varying degrees of boundary and type uncertainty, boundary refinement changes the spans being typed, and updates to candidate validity and type scores influence the final entity selection. The key is therefore not merely to introduce candidate refinement or structural constraints, but to ensure that subsequent decisions build on preceding calibration results. ProFiNER brings candidate refinement and set-level structured selection into tuning-free LLM inference, coordinating boundary calibration, type assignment, and conflict resolution through a progressive process that allows targeted intervention at intermediate stages.

3 METHOD

3.1 OVERALL WORKFLOW

As illustrated in Fig. 1, the overall pipeline consists of four stages: dual candidate mining, boundary-aware calibration, type-aware calibration, and conflict-aware selection. First, an LLM mines candidate entity mentions from an unlabeled corpus, which are then clustered and scored to form an initial candidate set. Next, mention boundaries and types are calibrated separately. Finally, a conflict graph is constructed, and global selection is performed using the Pareto frontier and the maximum-weight independent set (MWIS) to produce the final entity spans and type labels.

3.2 DUAL CANDIDATE MINING

Jointly predicting mentions, boundaries, and types in a single pass can suppress recall when labels are uncertain, while stochastic generation introduces boundary variation. Dual Candidate Mining therefore combines complementary prompting paths and repeated sampling to preserve plausible candidates for subsequent calibration.

Let $\mathcal{L} = \{l_1, \ldots, l_L\}$ denote the entity type set, where $L = |\mathcal{L}|$. Dual Candidate Mining comprises two complementary paths: **Type-agnostic Mention Mining** and **Type-wise Question Answering**. For each sentence $x_j = (w_{j,0}, \ldots, w_{j,T_j-1})$, Type-agnostic Mention Mining performs K_{agn} type-agnostic sampling runs to identify plausible entity spans without assigning semantic types. Type-wise Question Answering performs K_{typ} type-conditioned sampling runs for each $l_i \in \mathcal{L}$, querying for mentions of that type and providing initial type evidence. Together, the two paths combine type-independent mention discovery with targeted retrieval guided by individual entity types.

To aggregate local boundary variations, overlapping spans are grouped into boundary clusters $g(B)$. Let $R_{\text{agn}}(B)$ and $R_{\text{typ}}(B)$ denote the sampling runs that predict the exact boundary B, and $R_{\text{agn}}(g(B))$ and $R_{\text{typ}}(g(B))$ those predicting any boundary within its cluster. We characterize candidate uncertainty through candidate existence, boundary stability, and type consistency. First, different sampling may repeatedly identify the same entity while returning several similar but non-identical boundaries. Counting only the repetitions of an exact boundary would therefore distribute the evidence for an otherwise stable mention across multiple boundary variants. To address this issue, we define the cluster-level s_{cluster} as

$$s_{\text{cluster}}(B) = \frac{|R_{\text{agn}}(g(B))| + |R_{\text{typ}}(g(B))|}{K_{\text{agn}} + L \cdot K_{\text{typ}}}, \tag{1}$$

which measures whether the model consistently discovers an entity within a local region. Second, to distinguish among different boundary variants within the same cluster, we define the boundary stability as

$$s_{\text{boundary}}(B) = \frac{|R_{\text{agn}}(B)| + |R_{\text{typ}}(B)|}{|R_{\text{agn}}(g(B))| + |R_{\text{typ}}(g(B))|}, \tag{2}$$

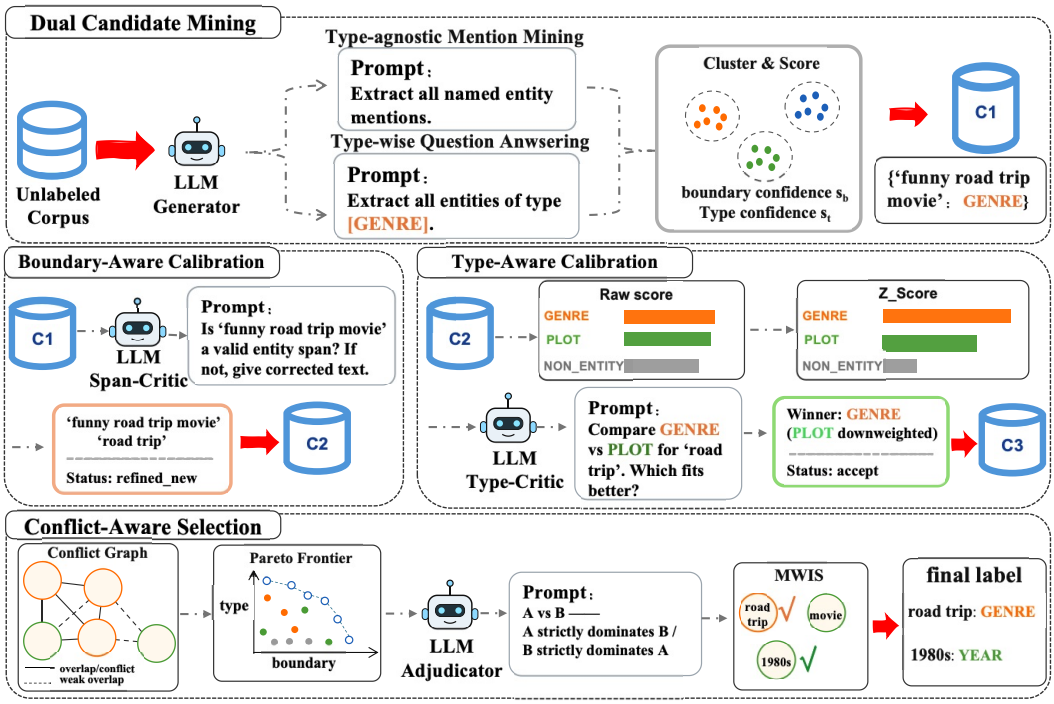

Figure 1: Overview of the proposed four-stage framework. (1) Candidate entities are generated through type-agnostic mention mining and decomposed type-wise QA. (2) Their boundaries are refined by a span critic. (3) Entity types are calibrated using score normalization and pairwise comparison. (4) Conflicting candidates are resolved to produce the final labels.

which measures the reproducibility of the specific boundary B. Finally, although type-conditioned sampling reduces direct competition among multiple labels, different type-specific queries may still provide conflicting type evidence for the same boundary cluster. To quantify the relative evidence for each type, let $n_{g,i}$ denote the number of predictions of type l_i in boundary cluster g. Given the prediction counts of all types, we select the type with the highest empirical probability as the candidate's initial type. The final s_{type} of candidate B in the first stage is then defined as the empirical probability of the selected type:

$$s_{\text{type}}(B) = \max_{1 \le i \le L} \frac{n_{g(B),i}}{\sum_{k=1}^{L} n_{g(B),k}}. \tag{3}$$

Together, s_{cluster}, s_{boundary}, and s_{type} preserve distinct evidence about candidate discovery, boundary stability, and type consistency without collapsing it into a single score. Candidates across all sentences form $\mathcal{C}_1$.

3.3 Boundary-Aware Calibration

Although $\mathcal{C}_1$ provides high candidate coverage, it may contain competing boundaries, and s_{boundary} measures generation consistency rather than boundary validity. Joint boundary and type reassessment can propagate errors between the two decisions (Wang et al., 2025; Shen et al., 2023; Li et al., 2023). We therefore selectively assess entity validity and boundary accuracy for candidates with low boundary support or overlapping spans, without reassigning types. Isolated, high-confidence candidates pass directly. Selected candidates are evaluated by a boundary critic:

$$\mathcal{M}_b(x_j, c, t_c) \to \left(y_{\text{ent}}, y_{\text{valid}}, \widetilde{z}\right), \tag{4}$$

where x_j denotes the sentence, c represents the current candidate boundary, y_{ent} indicates whether the span constitutes an entity, y_{valid} indicates whether its boundary is accurate, and $\widetilde{z}$ denotes the

proposed correction when the boundary is misaligned. The type information t_c serves only as auxiliary context for boundary assessment.

To avoid inadvertently discarding correct candidates, Boundary-Aware Calibration does not directly replace an original candidate with its proposed correction. If the critic confirms that the current boundary is valid, the candidate is retained. If it corresponds to an entity but its current boundary is judged inaccurate, we discount its boundary score s_{boundary} and introduce the corrected span into the candidate pool as a competing alternative:

$$s'_{\text{boundary}}(c) = \kappa_{\text{disc}} \cdot s_{\text{boundary}}(c), \qquad s_{\text{boundary}}(\widetilde{z}) = \kappa_{\text{corr}}, \tag{5}$$

where $s_{\text{boundary}}(c)$ denotes the boundary score of the original candidate c, $s_{\text{boundary}}(\widetilde{z})$ represents the assigned score for the newly proposed correction $\widetilde{z}$, and κ_{corr} is the initial boundary score assigned to the corrected span $\widetilde{z}$, and κ_{disc} is a discount factor that reduces the original candidate's boundary score when its boundary is judged inaccurate. The retained candidates and newly added corrections together form $\mathcal{C}_2$.

3.4 Type-Aware Calibration

This stage performs type calibration on the candidate set $\mathcal{C}_2$ generated in the previous stage. For each candidate $c \in \mathcal{C}_2$, the type evaluator assigns a score $s_c(l)$ to every label l, including NON-ENTITY, to measure the compatibility between candidate c and label l. Let B denote the span boundary of c. Unlike the candidate-level type confidence $s_{\text{type}}(B)$, derived from the maximum empirical type probability during Dual Candidate Mining, $s_c(l)$ is a per-label score produced at this stage.

Per-Type Score Standardization. Raw scores for different entity types may exhibit distinct centers and dispersions. Moreover, LLM scores may contain label-specific scale biases, making scores across labels not directly comparable. We therefore estimate the statistics of each label over the entire candidate pool $\mathcal{C}_2$:

$$\mu_l = \frac{1}{|\mathcal{C}_2|} \sum_{c \in \mathcal{C}_2} s_c(l), \qquad \sigma_l = \sqrt{\frac{1}{|\mathcal{C}_2|} \sum_{c \in \mathcal{C}_2} \big(s_c(l) - \mu_l\big)^2}, \tag{6}$$

and transform the raw scores as

$$\zeta_c(l) = \frac{s_c(l) - \mu_l}{\max(\sigma_l, \epsilon)}, \tag{7}$$

where $\epsilon > 0$ is a lower bound on the standard deviation that prevents instability for low-variance labels. Thus, $\zeta_c(l)$ measures how strongly the score of candidate c for label l deviates from the label-specific reference distribution.

Budget-Constrained Pairwise Adjudication. When candidate c overlaps another, we use a type adjudicator to compare semantically similar labels. Boundary overlap may reflect not only ambiguity among entity types but also misidentification of a non-entity span as an entity. We therefore include NON-ENTITY in pairwise adjudication to jointly assess type assignment and entityhood.

Using $\zeta_c(l)$, we rank the pairs of the candidate label set $\mathcal{L}_c$ by their ambiguity gap $g_c(l, l') = |\zeta_c(l) - \zeta_c(l')|$ and collect the closest ones in Π_c, always including the top-ranked pair, subject to a per-candidate comparison budget K_{cmp}:

$$|\Pi_c| \;=\; \min\Big\{K_{\text{cmp}},\; \tbinom{|\mathcal{L}_c|}{2}\Big\}. \tag{8}$$

For each pair $(l, l') \in \Pi_c$, the adjudicator returns

$$\delta_c(l, l') = \begin{cases} +1, & l \text{ wins}, \\ -1, & l' \text{ wins}, \\ 0, & \text{Other}. \end{cases} \tag{9}$$

When two labels have similar standardized scores, the existing evidence may be insufficient to distinguish them reliably. We therefore define the comparison weight according to their score distance:

$$w_c(l, l') = \max\Big(w_{min}, g_c(l, l')\Big), \tag{10}$$

which yields the final type score

$$\phi_c(l) = \zeta_c(l) + \sum_{(l,l')\in\Pi_c} w_c(l,l')\,\delta_c(l,l'). \tag{11}$$

The calibrated score $\phi_c(l)$ combines standardized type evidence with pairwise adjustments to guide type assignment and candidate retention. Candidates accepted or retained for further verification, with their assigned types and calibrated scores, form $\mathcal{C}_3$.

3.5 Conflict-Aware Selection

After Type-Aware Calibration, each candidate in $\mathcal{C}_3$ is associated with calibrated type evidence. However, candidate-wise calibration cannot ensure that predictions within the same sentence are mutually compatible, as multiple candidates may have identical or overlapping boundaries. We therefore model conflicts among candidates and perform global selection by integrating boundary, type, and pairwise adjudication evidence.

For sentence x_j, let $\mathcal{C}_{3,j}$ denote its candidate set. Each candidate $c_v \in \mathcal{C}_{3,j}$ corresponds to a node $v \in V_j$, with its span denoted by I_v. We construct a candidate conflict graph $G_j = (V_j, E_j)$, where an edge connects two candidates with overlapping spans, indicating that they cannot both be selected. We then decompose the graph into connected components and perform adjudication and selection independently within each component.

To reduce inference cost, we apply Pareto screening within each connected component of the sentence-level conflict graph. Each candidate's evidence vector comprises boundary evidence and type evidence. A candidate Pareto-dominates another if it is no worse in every dimension and strictly better in at least one. All non-dominated candidates form the Pareto frontier. We adjudicate only conflicting candidate pairs whose endpoints both lie on the frontier. For same-type candidates, we perform order-swapped bidirectional adjudication, map the reversed result back to the original candidate identities, and accept only outcomes for which both decisions agree. For different-type candidates, we perform a single adjudication without swapping the order. Based on the adjudication outcome, the winner receives a confidence-weighted positive increment, while the loser receives a negative increment of equal magnitude. The adjudication margin is computed as

$$m_v = \mathrm{clip}_{[-0.8,\,0.8]}\left(\frac{\sum_{k=1}^{n_v}\Delta_{v,k}}{\sqrt{1+n_v}}\right),$$

where $\Delta_{v,k}$ denotes the score change for candidate v in its k-th adjudication, and n_v denotes its number of adjudications. We constrain m_v to $[-0.8, 0.8]$ to prevent excessive influence.

For node v corresponding to candidate c_v, let B_v denote its candidate boundary and $\hat{l}_v$ its predicted type. We combine the cluster-level support $s_{\mathrm{cluster}}(B_v)$ and the boundary score $s_{\mathrm{boundary}}(B_v)$ retained or updated during Boundary-Aware Calibration with the initial type confidence $s_{\mathrm{type}}(B_v)$ from Dual Candidate Mining, and the calibrated type score $\phi_{c_v}(\hat{l}_v)$ from Type-Aware Calibration. For candidates newly generated by boundary correction in Boundary-Aware Calibration, no induction-stage type prior is available; we therefore set $s_{\mathrm{type}}(B_v) = 0$, so that their type evidence is provided solely by $\phi_{c_v}(\hat{l}_v)$. The resulting evidence is further adjusted by the adjudication margin m_v, yielding the final node score

$$q_v = \lambda_C\, s_{\mathrm{cluster}}(B_v) + \lambda_B\, s_{\mathrm{boundary}}(B_v) + \lambda_T\, s_{\mathrm{type}}(B_v) + \lambda_\phi\, \phi_{c_v}(\hat{l}_v) + \lambda_m\, m_v. \tag{12}$$

We formulate final candidate selection as a maximum-weight independent set problem:

$$\begin{aligned} \mathbf{y}_j^* &= \arg\max_{\mathbf{y}} \sum_{v\in V_j} q_v y_v, \\ \text{s.t.} \quad & y_u + y_v \le 1, \quad \forall\,\{u,v\} \in E_j, \\ & y_v \in \{0,1\}, \quad \forall\, v \in V_j. \end{aligned} \tag{13}$$

Here, $y_v = 1$ indicates that node v is selected. The constraints ensure that no pair of conflicting candidates can simultaneously appear in the final output. Accordingly, the selected candidates for sentence x_j are

$$\mathcal{C}_{4,j} = \{c_v \in \mathcal{C}_{3,j} \mid y_v^* = 1\}. \tag{14}$$

Aggregating selected candidates across sentences yields the final conflict-resolved entity set $\mathcal{C}_4$.

4 EXPERIMENT

4.1 EXPERIMENTAL SETTINGS

Datasets and Evaluation. We conduct experiments on five NER datasets: CoNLL-2003 (Tjong Kim Sang & De Meulder, 2003), MIT-Movie, MIT-Restaurant (Ushio & Camacho-Collados, 2021), and WikiGold (Balasuriya et al., 2009), as well as the CrossNER benchmark (Liu et al., 2020). We evaluate performance using entity-level micro-F1.

LLMs. We evaluate our framework using three LLMs: Qwen2.5-14B, Qwen2.5-32B (Qwen et al., 2025), and Qwen3.5-27B (Qwen Team, 2026). We apply all three models to entity extraction in a zero-shot setting without task-specific parameter updates. Unless otherwise specified, we use the locally deployed Qwen2.5-32B model for the analytical experiments. Detailed hyperparameter settings and prompts are provided in the supplementary materials.

Baselines. We compare our framework with several representative NER and information extraction methods, including InstructUIE (Wang et al., 2023), GoLLIE-34B (Sainz et al., 2024), GLiNER-S and GLiNER-L (Zaratiana et al., 2023b), Self-Improving (Xie et al., 2024), EvoPrompt (Tong et al., 2025), JPT (Ewais et al., 2026) and DiZiNER (Kim & Yoon, 2026).

4.2 MAIN RESULTS

Table 1 presents the overall zero-shot NER results on the four main benchmarks. Our method achieves the best performance on most datasets and the highest average performance among methods with complete benchmark coverage. It also remains competitive on CoNLL03, where it performs only slightly below the best result. More importantly, under each fixed backbone, our method consistently outperforms direct prompting across all datasets. This trend holds across different model families and scales, demonstrating that the improvements arise from the proposed framework rather than from a particular backbone configuration. Table 2 further evaluates our method in fine-grained cross-domain settings. Our method consistently improves upon the direct-prompt baseline across all five domains and achieves the best overall average performance. It ranks first or second in every domain, showing consistent effectiveness across diverse text distributions and fine-grained entity schemas. Together, these results indicate that our method generalizes effectively across domain shifts rather than being specialized to a particular dataset or entity domain.

Table 1: Overall zero-shot NER results across four benchmarks. Dashes denote unreported results, and averages are computed only for methods with complete benchmark coverage.

Method	CoNLL03	Restaurant	Movie	WikiGold	Avg.
InstructUIE (Wang et al., 2023)	–	20.99	63.00	–	–
GoLLIE-34B (Sainz et al., 2024)	–	52.70	62.40	–	–
GLiNER-S (Zaratiana et al., 2023b)	65.36	40.82	50.38	66.91	55.87
GLiNER-L (Zaratiana et al., 2023b)	62.63	45.38	62.33	62.55	58.22
Self-Improving (Xie et al., 2024)	74.99	–	–	73.98	–
EvoPrompt (Tong et al., 2025)	81.33	69.26	70.93	72.79	73.58
JPT (Ewais et al., 2026)	78.10	61.90	73.40	73.04	71.61
DiZiNER (Kim & Yoon, 2026)	**86.90**	67.30	76.20	–	–
Qwen2.5-14B					
Direct prompt	69.08	50.31	53.76	62.94	59.02
Ours	70.90	55.70	64.37	64.05	63.76
Qwen2.5-32B					
Direct prompt	75.31	66.87	65.71	75.84	70.93
Ours	82.53	72.23	75.89	78.99	77.41
Qwen3.5-27B					
Direct prompt	81.93	72.79	75.17	80.67	77.64
Ours	86.55	**74.56**	**77.15**	**83.33**	**80.40**

Table 2: Zero-shot NER performance on the CrossNER dataset. Best scores per domain are shown in bold.

Methods	AI	Literature	Music	Politics	Science	Average
UniNER (Zhou et al., 2024)	54.2	60.9	64.5	61.4	63.5	60.9
GLiNER (Zaratiana et al., 2023b)	57.2	64.4	69.6	72.6	62.6	65.3
GoLLIE (Sainz et al., 2024)	61.6	62.7	68.4	60.2	56.3	61.8
KnowCoder-7B (Li et al., 2024)	60.3	61.1	70.0	72.2	59.1	64.5
SaM (Cai et al., 2026)	58.2	68.7	71.2	68.2	65.1	66.3
GNER (Ding et al., 2024)	68.2	68.7	81.2	75.1	**76.7**	74.0
DiZiNER (Kim & Yoon, 2026)	71.1	72.7	80.6	79.4	74.8	75.7
JPT (Ewais et al., 2026)	**71.9**	72.2	**85.3**	77.0	71.3	75.5
Qwen3.5-27B (direct prompt)	67.57	70.93	80.07	76.26	71.74	73.31
Ours	71.3	**72.8**	82.7	**80.4**	75.3	**76.5**
Gain over Qwen3.5-27B	**+3.73**	**+1.87**	**+2.63**	**+4.14**	**+3.56**	**+3.19**

5 Analysis

5.1 Dual Candidate Mining

Table 3 presents ablations of the two candidate-generation paths across schema granularities. On coarse-grained datasets, removing Type-agnostic Mention Mining markedly degrades performance, whereas removing Type-wise Question Answering has a comparatively small effect. With few broad categories, type-agnostic mining provides sufficient candidate coverage, while specific types are largely determined during calibration. On fine-grained datasets, removing Type-wise Question Answering is most detrimental, although removing Type-agnostic Mention Mining also reduces performance. Closely related categories make the type evidence from unguided candidates too weak for calibration to effectively resolve, increasing the value of type-specific guidance. The two paths therefore play complementary roles: Type-agnostic Mention Mining primarily supports candidate coverage and is relatively insensitive to schema granularity, whereas Type-wise Question Answering strengthens type discrimination and becomes increasingly important as the label inventory expands and becomes more fine-grained.

Table 3: Ablation of the two candidate-generation paths across coarse- and fine-grained schemas. We evaluate the contributions of Type-agnostic Mention Mining and Type-wise Question Answering under different levels of schema granularity.

	Coarse-grained		Fine-grained	
Configuration	**CoNLL03**	**WikiGold**	**Restaurant**	**Movie**
Ours	**82.53**	**78.99**	**72.23**	**75.89**
w/o Type-agnostic Mention Mining	78.38	71.57	69.99	66.42
w/o Type-wise Question Answering	82.42	76.73	25.99	49.36

5.2 Effect of Each Stage

This section conducts a stage-level ablation of the three subsequent stages, namely Boundary-Aware Calibration, Type-Aware Calibration and Conflict-Aware Selection, by removing each stage from the full model while retaining the first stage in all settings. As shown in Table 4, all three stages contribute to the overall performance. The largest impact comes from Type-Aware Calibration: removing it decreases the average F1 by 8.13. Since a mistyped span is counted as entirely wrong under strict matching, correcting the systematic type confusion of the base model with dataset-level statistics removes the dominant source of errors. Conflict-Aware Selection yields positive gains on all datasets, with an average improvement of 1.43, as it discards contradictory overlapping candidates that would otherwise survive into the final prediction set. Boundary-Aware Calibration

improves three of the four datasets but slightly degrades CoNLL2003, whose relatively standardized entity boundaries leave little room for refining already accurate candidate spans. Together, the three stages provide complementary benefits without relying on one dominant component.

Table 4: Stage-level ablation results. We define $\Delta = \mathrm{F1}_{\mathrm{full}} - \mathrm{F1}_{\mathrm{ablated}}$. Thus, $\Delta > 0$ indicates that the removed stage contributes positively to performance, whereas $\Delta < 0$ indicates that removing the stage improves performance. Dual Candidate Mining is retained in all settings.

	CoNLL03		**WikiGold**		**Movie**		**Restaurant**	
Configuration	F1	Δ	F1	Δ	F1	Δ	F1	Δ
Full model	82.53	–	78.99	–	75.89	–	72.23	–
w/o Boundary-Aware Calibration	82.95	−0.42	77.22	+1.77	74.87	+1.02	72.11	+0.12
w/o Type-Aware Calibration	69.63	+12.90	66.56	+12.43	72.16	+3.73	68.79	+3.44
w/o Conflict-Aware Selection	81.05	+1.48	77.65	+1.34	74.30	+1.59	70.94	+1.29

Analysis of Type-Aware Calibration. To further verify whether Type-Aware Calibration truly calibrates entity types, we compute the rate at which it successfully corrects erroneous types and compare it with the rate at which it mistakenly alters originally correct types. As shown in Figure 2, across the four datasets, the error-correction rates are consistently high, whereas the incorrect relabeling rates remain very low. This indicates that Type-Aware Calibration corrects a substantial proportion of erroneous entity types while rarely affecting originally correct ones, confirming that it indeed performs effective calibration on entity types.

Analysis of Conflict-Aware Selection. To examine where Conflict-Aware Selection is effective, we divide the test samples into four groups according to the number of conflict edges before selection: 0, 1, 2, and ≥ 3. As shown in Figure 3, the module has almost no effect on conflict-free samples, whereas clear error reductions are observed once conflicts are present. The reductions are often more pronounced in conflict-dense samples, although the exact trend varies across datasets. These results confirm that Conflict-Aware Selection operates locally by resolving incompatibilities among competing candidate entities.

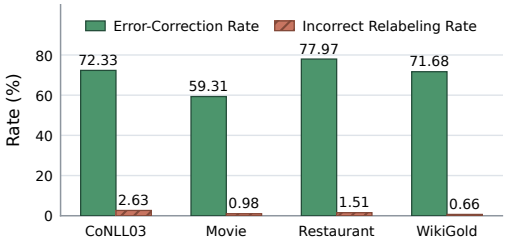

Figure 2: Type-Aware Calibration: error correction and incorrect relabeling rates across four datasets.

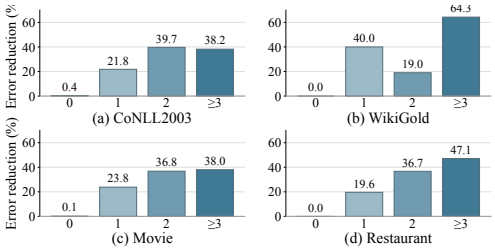

Figure 3: Entity error reduction from conflict-aware selection, grouped by the number of conflict edges.

6 Conclusion

We present ProFiNER, a progressive framework for zero-shot NER without task-specific annotations or parameter updates. By coordinating dual candidate mining, selective boundary calibration, type calibration, and conflict-aware global selection, ProFiNER improves candidate coverage and prediction consistency. Experiments and ablations support the effectiveness of this coordinated inference process. Future work will extend the framework to nested and overlapping entities, explore early-exit mechanisms to reduce latency, and adapt it to relation extraction.

AI USE STATEMENT

In this work, we used large language models (LLMs) to generate natural-language descriptions of entity types, which were incorporated into our method as auxiliary semantic information. LLMs were also used to assist with the implementation and editing of experimental code. We did not use generative AI to formulate theoretical or mathematical claims, propose hypotheses, design experiments, analyze data, or interpret experimental results. Tasks related to mathematical proofs and qualitative or thematic data analysis were not applicable to this work.

Additionally, we used LLMs to improve the readability and linguistic quality of the manuscript. All generated type descriptions were manually reviewed for correctness, relevance, and consistency with the corresponding entity definitions. AI-assisted code was inspected and tested by the authors, and all AI-edited text was carefully reviewed to ensure that it preserved the intended meaning and did not introduce unsupported claims or references. We take full responsibility for the final content of this work, including all text, code, and research artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work focuses on methodological research for zero-shot NER and does not involve the collection of new human-subject data or interaction with human participants. Our experiments are conducted on established NER benchmark datasets and publicly available or appropriately licensed language models. We do not intentionally collect or process private or personally identifiable information beyond what may already be contained in the original benchmark datasets.

Like other information extraction systems, the proposed method may inherit biases or errors from the underlying language models and training data. In particular, incorrect entity boundaries or type assignments could lead to unreliable downstream information extraction results. Therefore, predictions produced by the proposed framework should not be treated as authoritative in high-stakes applications without appropriate human verification.

We report experimental settings, evaluation protocols, and results as accurately and transparently as possible, and we encourage responsible use of the proposed method in accordance with the licenses and intended uses of the underlying datasets and models.

REPRODUCIBILITY STATEMENT

We provide the information necessary to facilitate reproduction of our experiments in the main paper, appendix, and supplementary materials. The datasets, evaluation metrics, language models, and experimental settings are described in the experimental section. The appendix provides the prompts and additional details of the proposed framework. We additionally provide anonymized source code as supplementary material to support reproduction of the reported experiments and evaluation.

REFERENCES

Dhananjay Ashok and Zachary C. Lipton. Promptner: Prompting for named entity recognition, 2023.

Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. Named entity recognition in Wikipedia. In *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources (People's Web)*. Association for Computational Linguistics, August 2009.

Ruichu Cai, Juntao Gan, Miao Mai, Zhifeng Hao, and Boyan Xu. SAM-NER: Semantic archetype mediation for zero-shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2026*, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Yuyang Ding, Juntao Li, Pinzheng Wang, Zecheng Tang, Bowen Yan, and Min Zhang. Rethinking negative instances for generative named entity recognition, 2024.

Ahmed Ewais, Ahmed Hashish, and Amr Ali. Just pass twice: Efficient token classification with llms for zero-shot ner. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11480–11497, 2026.

Quanjiang Guo, Yihong Dong, Ling Tian, Zhao Kang, Yu Zhang, and Sijie Wang. BANER: Boundary-aware LLMs for few-shot named entity recognition. In *Proceedings of the 31st International Conference on Computational Linguistics*. Association for Computational Linguistics, January 2025.

Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. Few-shot named entity recognition: An empirical baseline study. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, November 2021.

Siun Kim and Hyung-Jin Yoon. DiZiNER: Disagreement-guided instruction refinement via simulating pilot annotation for zero-shot named entity recognition. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Peng Li, Tianxiang Sun, Qiong Tang, Hang Yan, Yuanbin Wu, Xuanjing Huang, and Xipeng Qiu. CodeIE: Large code generation models are better few-shot information extractors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, July 2023.

Zixuan Li, Yutao Zeng, Yuxin Zuo, Weicheng Ren, Wenxuan Liu, Miao Su, Yucan Guo, Yantao Liu, Xiang Li, Zhilei Hu, Long Bai, Wei Li, Yidan Liu, Pan Yang, Xiaolong Jin, Jiafeng Guo, and Xueqi Cheng. KnowCoder: Coding structured knowledge into LLMs for universal information extraction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, August 2024.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. Crossner: Evaluating cross-domain named entity recognition, 2020.

Wenxuan Mu, Jinzhong Ning, Di Zhao, and Yijia Zhang. A multi-agent llm framework for multi-domain low-resource in-context ner via knowledge retrieval, disambiguation and reflective analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40:32528–32536, Mar. 2026. doi: 10.1609/aaai.v40i38.40529. URL https://ojs.aaai.org/index.php/AAAI/article/view/40529.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. GoLLIE: Annotation guidelines improve zero-shot information-extraction. In *The Twelfth International Conference on Learning Representations*, 2024.

Yongliang Shen, Xinyin Ma, Zeqi Tan, Shuai Zhang, Wen Wang, and Weiming Lu. Locate and label: A two-stage identifier for nested named entity recognition. 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)*. Association for Computational Linguistics, August 2021.

Yongliang Shen, Zeqi Tan, Shuhui Wu, Wenqi Zhang, Rongsheng Zhang, Yadong Xi, Weiming Lu, and Yueting Zhuang. PromptNER: Prompt locating and typing for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, July 2023.

Mohammad Golam Sohrab and Makoto Miwa. Deep exhaustive model for nested named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, October-November 2018.

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

Zeliang Tong, Zhuojun Ding, and Wei Wei. EvoPrompt: Evolving prompts for enhanced zero-shot named entity recognition with large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*. Association for Computational Linguistics, January 2025.

Asahi Ushio and Jose Camacho-Collados. T-NER: An all-round python library for transformer-based named entity recognition. In Dimitra Gkatzia and Djamé Seddah (eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, April 2021.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. GPT-NER: Named entity recognition via large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*. Association for Computational Linguistics, April 2025.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. Instructuie: Multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*, 2023.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. Empirical study of zero-shot NER with ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, December 2023.

Tingyu Xie, Qi Li, Yan Zhang, Zuozhu Liu, and Hongwei Wang. Self-improving for zero-shot named entity recognition with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*. Association for Computational Linguistics, June 2024.

Vikas Yadav and Steven Bethard. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*. Association for Computational Linguistics, August 2018.

Urchade Zaratiana, Niama Elkhbir, Pierre Holat, Nadi Tomeh, and Thierry Charnois. Global span selection for named entity recognition. In *Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS)*. Association for Computational Linguistics, December 2022.

Urchade Zaratiana, Nadi Tomeh, Niama El Khbir, Pierre Holat, and Thierry Charnois. Filtered semi-Markov CRF. In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics, December 2023a.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. Gliner: Generalist model for named entity recognition using bidirectional transformer, 2023b.

Junhao Zheng, Haibin Chen, and Qianli Ma. Cross-domain named entity recognition via graph matching. In *Findings of the Association for Computational Linguistics: ACL 2022*. Association for Computational Linguistics, May 2022.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. UniversalNER: Targeted distillation from large language models for open named entity recognition. In *The Twelfth International Conference on Learning Representations*, 2024.

A HYPERPARAMETER SETTINGS

Table 5 summarizes the hyperparameters used in ProFiNER. We use fixed values for the sampling and calibration parameters, while the weights used to combine different evidence sources in the final candidate scoring function are selected from predefined search ranges. Specifically, we set the numbers of type-agnostic and type-wise sampling runs to $K_{\text{agn}} = 10$ and $K_{\text{typ}} = 10$, respectively. For Boundary-Aware Calibration, the discount factor for an invalidated original boundary and the initial score assigned to a corrected boundary are set to $\kappa_{\text{disc}} = 0.60$ and $\kappa_{\text{corr}} = 0.65$. In Type-Aware Calibration, we use $\epsilon = 0.50$ and set the maximum number of pairwise type comparisons to $K_{\text{cmp}} = 4$. The minimum comparison weight used in pairwise type adjudication is set to $w_{\min} = 0.30$.

For Conflict-Aware Selection, the confidence levels returned by the adjudicator are mapped to confidence-weighted score increments: $\alpha(\text{low}) = 0.10$, $\alpha(\text{medium}) = 0.25$, and $\alpha(\text{high}) = 0.40$. The winner receives the corresponding positive increment, while the loser receives an increment of the same magnitude with opposite sign.

The final candidate score integrates cluster-level evidence, boundary evidence, initial type evidence, calibrated type evidence, and the adjudication margin. Their corresponding coefficients are searched within $\lambda_C \in [0.10, 0.25]$, $\lambda_B \in [0.30, 0.70]$, $\lambda_T \in [0.10, 0.25]$, $\lambda_\phi \in [0.01, 0.20]$, and $\lambda_m \in [0.01, 1.00]$.

Table 5: Hyperparameter settings and search ranges used in ProFiNER.

Hyperparameter	**Value / Search Range**
K_{agn}	10
K_{typ}	10
κ_{disc}	0.60
κ_{corr}	0.65
ϵ	0.50
K_{cmp}	4
$w_{\min}$	0.30
$\alpha(\text{low})$	0.10
$\alpha(\text{medium})$	0.25
$\alpha(\text{high})$	0.40
λ_C	[0.10, 0.25]
λ_B	[0.30, 0.70]
λ_T	[0.10, 0.25]
λ_ϕ	[0.01, 0.20]
λ_m	[0.01, 1.00]

B CASE STUDY

We take a closer look at representative cases in which initially noisy entity hypotheses are progressively corrected, as shown in Fig. 4. ProFiNER preserves alternative span boundaries, calibrates their entity types, and resolves incompatible hypotheses through conflict-aware global selection. These cases illustrate how complementary boundary, semantic, and global evidence jointly correct local prediction errors and recover gold-consistent entity sets.

Input Sentence: do you know where i can find a pg sports movie from the 1940 s that was directed by wych kaosayananda.
Gold Label: {("pg", RATING), ("sports", GENRE), ("1940 s", YEAR), ("wych kaosayananda", DIRECTOR)}.
Progressive refinement traces:
Boundary: the critic retains **pg** and **sports** while introducing the competing expansions **pg sports** and **sports movie**.
Type: **pg**: GENRE -> RATING; **sports** remains GENRE.
Initial candidates: {("pg", GENRE), ("pg sports", GENRE), ("sports", GENRE), ("1940 s", YEAR), ("wych kaosayananda", DIRECTOR)}.
Ours pred.: {("pg", RATING), ("sports", GENRE), ("1940 s", YEAR), ("wych kaosayananda", DIRECTOR)}.

Input Sentence: However this was not the first time Hitzig had experienced the interaction between the brain and electricity; earlier in his career as a physician working with the Prussian Army he experimented on wounded soldiers whose skulls were fractured by bullets.
Gold Label: {("Hitzig", PER), ("Prussian Army", ORG)}.
Progressive refinement traces:
Boundary: the critic adds **the Prussian Army** as an alternative; conflict resolution removes both the determiner-bearing span and the incomplete span **Prussian**.
Type: **Prussian Army**: MISC -> ORG.
Initial candidates: {("Hitzig", PER), ("Prussian", MISC), ("Prussian Army", MISC)}.
Ours pred.: {("Hitzig", PER), ("Prussian Army", ORG)}.

Figure 4: Qualitative cases illustrating the progressive correction of entity boundaries and types. Green denotes gold-consistent entities, whereas blue denotes erroneous or competing initial hypotheses.

C Prompt

C.1 Dual Candidate Mining

Prompt for Type-agnostic Mention Mining

```
You are a named entity recognition expert. Extract all named entity
mentions from the given text, regardless of entity type.
Provide answers in the following JSON format: {"entities":
[{"text": "...", "start": int, "end": int}]}.
If there are no entity mentions, return: {"entities": []}.
Text: "{sentence}"
Answer:
```

Prompt for Type-wise Question Answering

```
Given entity type: [{entity_type}].
Definition of {entity_type}: {type_definition}
You are a named entity recognition expert. Extract all entities of
type [{entity_type}] from the given text.
Provide answers in the following JSON format: {"entities":
[{"text": "...", "start": int, "end": int}]}.
If there is no entity of this type, return: {"entities": []}.
Text: "{sentence}"
Answer:
```

C.2 Boundary-Aware Calibration

Prompt for Boundary-Aware Calibration

```
Given original text: "{sentence}"
Given candidate span: "{span_text}" (character positions {start} to
{end})
Given type hint: {type_hint} (ignore the type constraint if it is
[UNK])
You are an expert in named entity boundary detection. Determine
whether the given text span forms a complete named entity mention.
If not, provide the correct entity boundary.
Please determine:
1. Does the candidate span form a complete entity mention with
valid boundaries?
2. If it is incomplete, what should the complete span be? Provide
the corrected text.
3. If it is not an entity at all, mark it as "not an entity".
Provide answers in the following JSON format: {"is_valid": bool,
"corrected_text": "..." | null, "is_entity": bool}.
Answer:
```

C.3 Type-Aware Calibration

Prompt for Type-Aware Calibration

```
Given allowed labels:
{entity_type_schema}
You are a named entity type calibration expert. For each candidate
span, score every allowed entity type and NON_ENTITY from 1 to 10.
A score above 5 means the label may be plausible enough for
pairwise comparison. UNLABELED means the current candidate has no
trusted type label; infer its type from the span, context, and
allowed labels. If multiple labels seem plausible, lower all their
scores; only assign a high score (>=8) when this label is clearly
better than alternatives.
Provide answers in the following JSON format: [{"id": int,
"scores": {"TYPE": 1.0}, "review": bool, "reasoning": "brief
rationale"}].
Candidates:
{candidates_json}
Answer:
```

Prompt for Pairwise Type Adjudication

```
Given allowed labels:
{entity_type_schema}
Given sentence: "{sentence}"
Given candidate span: "{span_text}"
You are a pairwise named entity type adjudicator. For each
requested label pair, decide which label better describes the same
candidate span in the sentence. Do not introduce new labels.
Requested comparisons:
{pair_comparison_lines}
Provide answers in the following JSON format: [{"id": int,
"label_a": "TYPE", "label_b": "TYPE", "winner": "label_a",
"reasoning": "brief rationale"}].
For each item, winner must be exactly one of: label_a, label_b,
TIE, UNCLEAR. Do not put an entity type name in winner.
Pairs:
{pairs_json}
Answer:
```

C.4 Conflict-Aware Selection

Prompt for Conflict-Aware Selection

```
Given entity type schema:
{entity_type_schema}
Given sentence: "{sentence}"

Candidate A:
{candidate_a}
Candidate B:
{candidate_b}

You are a pairwise NER adjudicator. Compare Candidate A and
Candidate B under the given entity schema, focusing on entity
validity and exact BIO-style boundary correctness.

Rules:
1. Decide whether each candidate is a valid entity annotation under
the schema.
2. If both are valid but conflict, prefer the candidate with the
more accurate entity boundary.
3. Do not prefer longer or shorter spans by default. Extra or
omitted tokens are valid only if they belong to the entity mention
according to the type definition.
4. Do not propose new spans or types, and do not modify offsets.
5. If both candidates are equally valid, choose "both_valid"; if
neither is valid, choose "neither_valid"; if evidence is
insufficient, choose "uncertain".

Provide answers in the following JSON format:
{"bio_boundary_preference": "A | B | tie | not_applicable |
uncertain",
 "decision": "A_strictly_dominates_B | B_strictly_dominates_A |
both_valid | neither_valid | uncertain",
 "basis": "boundary | type | entityness | nesting | evidence",
 "confidence": "high | medium | low",
 "reason": "one brief sentence"}.
Answer:
```

D Cost Analysis

Table 6 reports the number of logical model requests normalized by the number of input sentences. We distinguish logical requests from generated completions because a single Stage 1 request can return multiple samples. The Stage 1 share is computed relative to the complete Stage 1–4 pipeline, while downstream adaptive requests include boundary refinement, type calibration, and conflict resolution in Stages 2–4.

Table 6: Logical request profile of the complete pipeline.

Dataset	Total requests / sentence	Stage 1 share	Downstream adaptive requests / sentence
CoNLL2003	5.75	86.9%	0.75
WikiGold	5.58	89.6%	0.58
MIT-Movie	13.82	94.1%	0.82
MIT-Restaurant	10.24	87.9%	1.24

E PERFORMANCE OF DUAL CANDIDATE MINING

To examine the predictions produced by repeated sampling before subsequent calibration and selection, we evaluate the candidate set $\mathcal{C}_1$ generated by Stage 1 alone. This setting retains both Type-agnostic Mention Mining and Type-wise Question Answering, using the sampling configuration of ProFiNER. Candidate types are obtained from the first-stage aggregation procedure, without applying Boundary-Aware Calibration, Type-Aware Calibration, or Conflict-Aware Selection. Table 7 reports precision, recall, F1, and boundary recall on the four benchmarks.

Table 7: Stage 1 only performance with repeated sampling, before calibration and selection. All values are percentages.

Dataset	**Precision**	**Recall**	**F1**	**Boundary recall**
CoNLL-2003	69.55	67.16	68.33	93.84
WikiGold	69.89	61.57	65.47	84.84
MIT-Movie	73.67	72.67	73.17	79.06
MIT-Restaurant	69.53	70.33	69.93	75.37