Improving LLM-Based Entity Detection with Annotation Guidelines

Marcel Geromel and **Philipp Cimiano**
CITEC, Bielefeld University
{mgeromel,cimiano}@techfak.uni-bielefeld.de

Abstract

In this work, we investigate whether detailed, dataset-specific annotation guidelines can replace or augment annotated few-shot exemplars for prompt-based entity detection. To this end, we propose an in-context learning framework that conditions the LLM on a guideline document, task instructions, and optional demonstrations, enabling single-pass, primarily zero-shot nested entity detection. We evaluate multiple instructional aspects from the comprehensive ACE 2005 annotation guidelines, using three guideline-compliant datasets as benchmarks and GPT-5 variants for inference. Based on extensive experimentation, we demonstrate that detailed, dataset-specific instructions, especially those governing mention boundaries, consistently and robustly improve performance, even in conjunction with established few-shot methods. Indeed, even under zero-shot constraints, our guideline approach outperforms state-of-the-art few-shot methods on ACE-2005, achieving up to 81.2 percent F1 without relying on existing labeled datasets. Our findings present detailed annotation guidelines as a promising resource for improving instruction-based information extraction.

1 Introduction

Entity detection (ED)–the task that requires the identification and classification of mentions of specific entities, such as persons, organizations, or locations, in unstructured textual sources–remains a cornerstone of natural language processing (NLP). As a fundamental component in information extraction procedures, the accurate detection of entities (or mentions thereof) underpins a multitude of downstream applications, including knowledge base population (Al-Moslmi et al., 2020), question answering (Jin et al., 2022), and relation and event extraction (Wadden et al., 2019; Lin et al., 2020). Traditional approaches to detecting entity mentions have primarily depended on supervised machine learning and massive labeled datasets (Jehangir et al., 2023; Hu et al., 2024b), normally requiring extensive domain-specific resources (Patel et al., 2018; Loukas et al., 2022) and manual annotation.

The emergence and continuous advancement of large language models (LLMs), such as Llama 3.0 (Grattafiori et al., 2024), GPT-4o (OpenAI, 2024), and DeepSeek-V3 (DeepSeek-AI, 2024), has transformed the NLP landscape. These models, trained on massive textual corpora and fine-tuned for instruction following (Ouyang et al., 2022; Zhang et al., 2025a), demonstrate remarkable generalization capabilities (Yang et al., 2025; Budnikov et al., 2025) that facilitate a purely instruction (or *prompt*)-driven approach to various linguistic tasks, including information extraction (Xu et al., 2024; Zhang et al., 2025b), without explicitly requiring task-specific supervision.

Yet, despite these advances, previous research mostly adopts shot-based (Bölücü et al., 2023; Liu et al., 2024; Jiang et al., 2024; Zhang et al., 2024; Wang et al., 2025a) or multi-turn prompting (Xie et al., 2024; Wu et al., 2024; Wang et al., 2024; Ye and Zhang, 2025; Tong et al., 2025) paradigms, where annotated examples or iterative requests are required to nudge the LLM toward the desired behavior. These techniques, however effective, tend to increase the required computational expense or technical expertise, or demand cleverly selected, annotated exemplars to facilitate effective in-context learning (ICL), thus making the method's success dependent on the existence of specifically labeled datasets. Furthermore, considering the increasing attention on synthetic methods for labeling datasets where annotations are generated using pre-trained LLMs and intended for subsequent knowledge distillation (Zhang et al., 2023a; Goel et al., 2023; Tan et al., 2024; Pangakis and Wolken, 2024; Horych et al., 2025), we hypothesize that prompt-based approaches require meticulous, task-related instructions for producing consistent and reproducible an-

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 5251–5268
November 6-10, 2026 ©2026 Association for Computational Linguistics

notations. For humans as annotators, such concerns are normally addressed by designing and providing sufficiently detailed *annotation guidelines*.

This work explores whether, and to which extent, prompt-based entity recognition using pre-trained LLMs can be augmented by integrating *detailed* annotation guidelines in zero- and few-shot scenarios. To this end, we propose a guideline-driven ICL approach that renders labeled datasets and iterative prompting optional instead of necessary. In short, we contribute the following:

- We introduce a single-pass and primarily zero-shot ICL approach to entity detection, leveraging detailed annotation guidelines as opposed to annotated examples for inference.
- Adopting the GPT-5 variants for inference, we evaluate our approach using three datasets, explore the impact of various guideline aspects on performance, and compare (and combine) our approach to common few-shot methods.
- Based on extensive experiments, we demonstrate that employing detailed guidelines consistently boosts performance in both zero-shot and few-shot settings. Remarkably, even under zero-shot constraints, our guideline-based approach outperforms SOTA few-shot methods on ACE-2005, achieving 81.2 percent F1.

We continue by covering related works on example-based ICL (Section 2) before introducing our guideline-based ICL approach (Section 3), explaining our experiments (Section 4), and presenting and discussing our results (Sections 5 and 6).

2 Related Work

Example-based Learning Following the remarkable instruction-following capabilities of LLMs, plenty of recent works have proposed and analyzed various prompt-based methods for generative information extraction (Xu et al., 2024; Zhang et al., 2025b), particularly for detecting entities (Cai et al., 2023; Ashok and Lipton, 2023; Zhang et al., 2024; Cheng et al., 2024; Zhao et al., 2025; Xiao et al., 2025a; Wang et al., 2025a).

Thereamong, the paradigm most notoriously followed is few-shot learning, where k annotated exemplars are provided as demonstrations to promote in-context learning in LLMs.

These works, however, are primarily concerned with introducing or improving strategies for selecting these annotated examples, ranging from random sample selection (Bölücü et al., 2023; Ashok and Lipton, 2023) and clustering approaches (Jiang et al., 2024; Wang et al., 2024) to exemplar selection over sequence-level (Cai et al., 2023; Liu et al., 2024; Yan et al., 2025; Tang et al., 2025; Xiao et al., 2025a) or mention-level similarity (Wu et al., 2024; Li et al., 2024b; Wang et al., 2025a).

As a consequence, despite their empirical success, example-driven methods (the similarity-based strategies in particular) tend to encourage overfitting to or reproducing a demonstrated pattern (Fan et al., 2024; Li et al., 2024a; Schoch and Ji, 2025; Gou et al., 2025) as opposed to *generalizing*, while establishing a notable dependence on existing, adequately labeled examples.

Zero-Shot Learning In parallel, a growing body of research has explored various zero-shot methods for detecting entity mentions (Tirskikh and Konovalov, 2023; Xie et al., 2024; Wang et al., 2024; Hu et al., 2024a; Cocchieri et al., 2025; Wang et al., 2025b; Averly and Ning, 2025), where LLMs are mainly guided using natural-language instructions, without access to human-labeled examples.

Most zero-shot methods incorporate task-related information–such as brief label definitions (Picco et al., 2024; Sainz et al., 2024; Cocchieri et al., 2025) or preference guidelines (Zamai et al., 2024; Tong et al., 2025; Huang et al., 2025)–into the instructions to facilitate the desired outputs. Further methods propose (multi-turn) task decomposition, where mentions are extracted using pipelines (Xie et al., 2024; Wu et al., 2024; Ye and Zhang, 2025), structured code generation approaches (Sainz et al., 2024; Han et al., 2025), or *self-verification* to mitigate hallucinated mention predictions (Xie et al., 2024; Wang et al., 2025b).

In summary, zero-shot methods are largely limited to modifications of surface-level descriptions and instructions and, while these 'guidelines' typically describe all relevant aspects of *what* features to annotate, they seldom explain *how* (e.g., what rules to follow), disregarding the underlying procedures that governed the dataset's creation. Furthermore, most zero-shot methods evaluate model performance on different datasets using the same instructions, suggesting that differences in performance could result from an accidental alignment between the model's default behavior (conditioned via the instructions) and the annotation convention that shaped the dataset's labeling, as opposed to

reflecting an improved task understanding.

Therefore, we contend that introducing *real* annotation guidelines into the task descriptions could strengthen annotation consistency, which is especially important when considering recent works on labeling datasets using LLMs (Zhang et al., 2023b; Törnberg, 2024; Tan et al., 2024; Rønningstad et al., 2024; Gu et al., 2025). Notably, employing dataset-specific instructions, Hu et al. (2024a) have achieved an increase in performance for clinical entity recognition. However, these guidelines have been introduced *alongside* additional label definitions, whereby the precise impact on performance remains unclear.

3 Method

Task Definition In nested entity detection, an extension of traditional named entity recognition (NER), we seek to identify and classify entities (or mentions thereof) whose extent may overlap or be hierarchically embedded within one another.

More formally, provided a token sequence $x = (x_1, x_2, ..., x_n)$, we intend to extract the (taggable) mentions $m_i \in \mathcal{Y}(x)$ expressed in input x, having

$$\mathcal{Y}(x) \subset \{(s, e, t) \mid 1 \leq s \leq e \leq n, t \in \mathcal{T}\}$$

where each mention m_i is represented by its inclusive start (s) and end (e) token-level boundaries, representing the subsequence $(x_s, ..., x_e)$, followed by the referent's inherent type t, with $\mathcal{T}$ containing the predefined entity labels, such as `PER` (person) and `ORG` (organization). Furthermore, as mentions may be contained in another taggable mention, the following is permitted:

$$\exists m_i, m_j \in \mathcal{Y}(x) : [s_i, e_i] \subset [s_j, e_j]$$

Framework We introduce a prompt-based generative framework for nested ED, leveraging a pretrained LLM as a sequence-to-sequence 'annotator' A. Besides the input sequence x, annotator A may be conditioned on the following material:

- **Guideline Document** G – a structured description of annotation schema and guidelines, containing, e.g., definitions and (domain-specific) clarifications of entity labels or mentions in general, or inclusion and exclusion criteria for marking the mention boundaries (i.e., an entity's extent). For example, G may stipulate that organizational subunits, such as "Finance Department," are always labeled as an organization (`ORG`), even when contained within greater (taggable) entities.

- **Instruction Prompt** I – a specification describing the annotation task (i.e., detecting [nested] entity mentions) and the provided or expected input and output formatting. For instance, the annotation conventions in I, presented in Figure 1, demand that mentions be annotated via insertion of markup symbols, such as `<PER>` and `</PER>`, and leaving the sequence x otherwise *unaltered* to enforce reliable parsing of outputs.

- **Demonstrations** D – a collection of k input-output examples (x^i, y^i), where the exemplar inputs x^i have gold annotations y^i (i.e., sequence x^i with markers included), serving as in-context demonstrations. The examples presented may be sampled dynamically, e.g., based on some semantic, lexical, or structural similarity to target sequence x, to maximize their contextual relevance. Notably, D is introduced for comparing our guideline-based approach (using G) with common few-shot methods (using D), see Section 4.

For inference, the annotator A is presented a guideline document G, instructions I, (optional) demonstrations D, and our sequence x, concatenated, to generate a labeled sequence $\hat{y} = (\hat{y}_1, ..., \hat{y}_m)$, i.e.:

$$(\hat{y}_1, ..., \hat{y}_m) = A(x \mid G, I, D)$$

[GUIDELINES G]

Your Task

You are tasked with annotating entity mentions (including nested entities) in sentences, carefully following the provided Entity Detection guidelines (shown above).

To annotate an entity mention, enclose the entity's extent with `<TYPE>` and `</TYPE>` markers, replacing `TYPE` with an appropriate type from the following list:

[ENTITY_TYPES $\mathcal{T}$]

Your inputs will be formatted in `JSON`, mapping `sentence_ids` to sentences.

Your output must be formatted in `JSON`, mapping the `sentence_ids` to annotated sentences with suitable entity markers inserted (and otherwise left *unchanged*).

[DEMONSTRATIONS D]

Your Input:

[INPUT x]

Your Output:

Figure 1: The (prompt) template for our instructions I, provided in *markdown*. The **[placeholders]** are replaced at runtime, depending on the experiment.

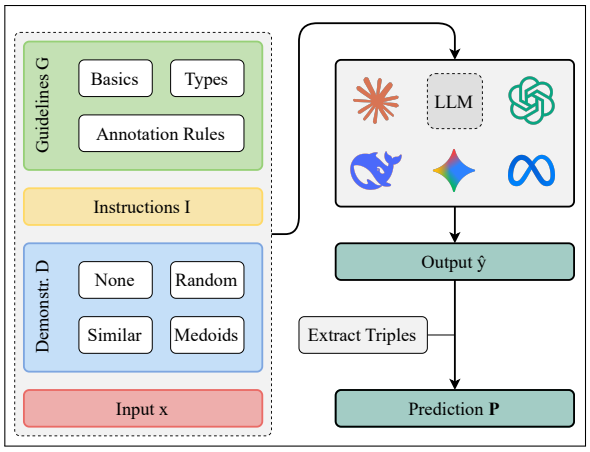


Figure 2: An illustration of our framework. The model is presented with some guideline document G, instructions I, (optional) demonstrations D, and an input sentence x.

such that $\hat{y}$ *minus* the markers recovers sequence x:

$$(x_1, ..., x_n) \mathrel{\widehat{=}} (\hat{y}_i \mid \hat{y}_i \notin \mathcal{S}_\mathcal{T})_{i=1}^{m} \quad (1)$$

with $\mathcal{S}_\mathcal{T}$ comprising the permitted markup symbols, e.g., `<PER>` and `</LOC>`, assuming the following condition is fulfilled: $\forall x_i : x_i \notin \mathcal{S}_\mathcal{T}$. The predicted mentions are ultimately recovered by transforming the mentions identified in sequence $\hat{y}$ into (s, e, t) triples via Algorithm 1, discarding any structurally invalid predictions (e.g., unclosed `<PER>` tags). No additional post-processing is performed.

In this framework, nested entity mentions can be naturally represented via embedded tag structures over the input x, permitting complex mention hierarchies without architectural constraints or requiring any heuristic methods (e.g., sequence matching) to determine the respective entity spans (s, e) from the model output $\hat{y}$, as is common when mentions are generated *directly* through LLMs. Our framework is illustrated in Figure 2.

Output Format To guarantee the properties from Equation 1 (i.e., parsable $\hat{y}$), we mask the model's token logits (next-token generation) such that each partial output $\hat{y}_1, ..., \hat{y}_{i-1}, 1 \leq i \leq m$, is proceeded by a) a markup symbol in $\mathcal{S}_\mathcal{T}$, or b) the input token x_p, where $p = i - |\{\hat{y}_j \mid 1 \leq j < i, \hat{y}_j \in \mathcal{S}_\mathcal{T}\}|$.

4 Experiments

Annotation Guidelines For our annotation guidelines G, we consider the comprehensive and well-documented *ACE 2005 English Annotation Guidelines for Entities*[1] (denoted *AGE*) as our reference document, because they convey particularly extensive instructions on both boundary detection and classification of entity mentions.

To condense AGE into document G, we produce a summary comprising any task-relevant concepts (e.g., linguistic) and instructions, such as annotating nominal or pronominal mentions, spanning the following: *§1 Basic Concepts*, *§2 Entity Class*, *§3 Mention Type*, *§4 Metonymy*, and *§5 Entity Types*. To ensure reproducible experiments, our complete document G is presented in Appendix A.

We emphasize that document G (and AGE) comprise examples for demonstrating the guidelines for boundary detection, particularly in *§2 Entity Class* and *§3 Mention Type*. The examples are, however, unlabeled regarding any entity types $\mathcal{T}$, and primarily illustrate the required linguistic concepts.

Datasets To properly evaluate G's influence on an LLM's predictive capability, we require labeled datasets with G-compliant annotations. As such, we conduct our experiments on ACE-2005 (Walker et al., 2006), Litbank (Bamman et al., 2019), and REFLEX (Walker et al., 2009), which are annotated via AGE, with marginal deviations for Litbank. For further details and information on pre-processing, see Appendix B.

Models Our guideline-approach is evaluated on state-of-the-art (SOTA) reasoning LLMs with varying footprints: GPT-5[2] and GPT-5 nano/mini.[3] For each model, we selected the `"medium"` reasoning effort and, apart from limiting the number of output tokens to 2^{15} (reasoning *and* responses, combined), other hyper-parameters remained unaltered.

We minimize the overhead (and cost) produced by document G by providing 8 randomly batched samples for annotation within each request. The batched samples are presented in JSON format, mapping the sentence ids to sentences, expecting an identically formatted JSON object as an output (see Figure 1). We ensure this format using OpenAI's Structured Outputs.[4] Likewise, we facilitate our *unchanged* property from Equation 1 by constraining each individual prediction $\hat{y}$ through the following regular expression:

$$\mathbf{P}(x, \mathcal{T}) := \mathcal{S}_\mathcal{T}^k \, x_1 \, \mathcal{S}_\mathcal{T}^k \, x_2 \, \cdots \, \mathcal{S}_\mathcal{T}^k \, x_n \, \mathcal{S}_\mathcal{T}^k$$

[1] https://www.ldc.upenn.edu/collaborations/past-projects/ace/

[2] gpt-5-2025-08-07

[3] gpt-5-{nano,mini}-2025-08-07

[4] https://platform.openai.com/docs/guides/structured-outputs

where $k \in \mathbb{N}$ determines the (maximum) amount of markers $\hat{y}_i \in \mathcal{S}_\mathcal{T}$ inserted between two consecutive input tokens x_i and x_{i+1}. Following the examples presented in the original AGE, we select $k = 3$. Our `json_schema` is presented in Appendix D.

Guideline Scenarios To investigate the influences of document AGE's a) conventions for mention and boundary detection, and b) label descriptions (and how to discriminate them), we consider four different scenarios:

- **Basic** – Our document G only expresses some basic task instructions and terminology, thus mirroring the original §1 Basic Concepts from AGE, see Figure 4. Such (or analogous) guidelines are commonly found in related studies.
- **+A** – Our document G is expanded (from Basic) by including any linguistic concepts from §2 Entity Class, §3 Mention Type, and §4 Metonymy, see Figures 4 and 5.
- **+D** – Our document G is expanded (from Basic) by incorporating any label definitions (and preferences for discrimination) from §5 Entity Types, presented in Figure 6.
- **+AD** – Our document G comprises our §1 Basic Concepts, any instructions for mention and boundary detection (+A), beside label descriptions and discrimination rules (+D).

We compare (and combine) these guideline scenarios with three common few-shot methods. The demonstrations D stem from the training samples (for ACE 2005) and non-test samples (for Litbank and REFLEX).

- **Random** – For each sequence presented (8 batched samples), D contains one labeled example, selected uniformly at random.
- **Similar** – For each sequence to-be-annotated, D comprises the semantically most-similar labeled example. For selection, we compute the sentence encodings for all candidate examples via `all-mpnet-base-v2`,[5] and determine the semantic similarity via Cosine similarity.
- **Medoids** – Here, D contains 8 exemplars that represent our respective dataset's sequences. Therefore, we perform k-Medoids clustering (Kaufman and Rousseeuw, 1990; Wang et al., 2024), choosing `all-mpnet-base-v2` for encoding sentences, and consider the obtained (eight) medoids as demonstrations.

Evaluation For an input x, let $\mathcal{P}$ denote the predicted entity mentions and $\mathcal{Y}$ denote the reference entity mentions. Following standard evaluation procedure, a prediction $(s, e, t) \in \mathcal{P}$ is deemed correct if $(s, e, t) \in \mathcal{Y}$. From this, we calculate our overall (micro) F1 scores.

5 Results

ACE-2005		Few-Shot Strategy			
Model	**Prompt**	**None**	**Random**	**Similar**	**Medoids**
gpt-5 (nano)	Basic	38.0 +4.6	39.6 +4.4	41.1 +4.4	38.4 +4.5
	+ A	39.5 +4.4	40.4 +4.5	41.8 +4.4	40.3 +4.7
	+ D	39.3 +4.1	40.0 +4.4	41.3 +4.3	40.4 +4.3
	+ AD	40.0 +4.9	40.4 +4.1	41.2 +4.6	40.4 +3.7
gpt-5 (mini)	Basic	55.0 +4.0	58.9 +4.4	61.7 +3.8	54.9 +2.7
	+ A	60.9 +3.7	63.1 +3.1	65.1 +3.7	63.6 +3.4
	+ D	56.0 +4.6	59.8 +4.1	61.8 +3.8	59.0 +3.5
	+ AD	61.4 +3.7	63.7 +3.4	64.6 +3.2	62.5 +3.5
gpt-5	Basic	65.9 +4.7	67.1 +4.5	69.6 +4.0	64.9 +4.4
	+ A	72.5 +3.8	–	75.1 +3.3	–
	+ D	66.8 +4.8	–	–	–
	+ AD	73.7 +3.4	–	75.8 +3.0	–

Litbank		Few-Shot Strategy			
Model	**Prompt**	**None**	**Random**	**Similar**	**Medoids**
gpt-5 (nano)	Basic	29.7 +1.7	32.3 +2.2	35.5 +2.6	31.3 +1.8
	+ A	32.0 +3.0	32.3 +4.0	35.1 +3.5	33.9 +2.8
	+ D	31.7 +2.1	32.9 +2.5	36.5 +3.6	32.5 +3.3
	+ AD	34.4 +2.8	35.0 +3.4	37.6 +5.2	34.2 +3.1
gpt-5 (mini)	Basic	45.9 +5.5	55.3 +6.9	60.1 +5.6	54.8 +5.2
	+ A	58.6 +5.7	61.3 +5.1	65.3 +4.6	63.3 +4.0
	+ D	53.6 +3.8	59.7 +5.8	63.7 +4.6	62.4 +5.9
	+ AD	61.5 +5.1	63.5 +4.1	66.4 +3.7	64.5 +5.0
gpt-5	Basic	61.7 +6.1	69.4 +4.4	72.4 +2.1	72.4 +2.3
	+ A	71.7 +4.2	–	75.8 +2.2	–
	+ D	63.7 +4.1	–	–	–
	+ AD	71.6 +3.2	–	75.4 +2.1	–

REFLEX		Few-Shot Strategy			
Model	**Prompt**	**None**	**Random**	**Similar**	**Medoids**
gpt-5 (nano)	Basic	34.1 +3.4	36.3 +3.5	37.1 +4.1	35.8 +3.6
	+ A	34.5 +4.2	36.9 +4.2	37.7 +3.9	36.6 +4.7
	+ D	35.0 +3.8	36.3 +4.1	37.9 +4.2	36.1 +3.5
	+ AD	35.1 +3.8	36.5 +3.7	38.5 +4.4	36.9 +4.0
gpt-5 (mini)	Basic	52.7 +4.3	58.0 +5.1	57.7 +4.6	58.8 +5.2
	+ A	59.7 +3.9	63.1 +4.7	63.5 +4.0	62.3 +4.1
	+ D	53.3 +4.5	58.5 +4.1	58.6 +4.2	58.4 +4.7
	+ AD	58.6 +3.7	62.5 +3.9	63.4 +3.8	61.9 +3.8
gpt-5	Basic	64.7 +4.6	66.2 +3.8	67.5 +4.0	67.8 +3.2
	+ A	71.2 +3.5	–	72.8 +2.9	–
	+ D	65.2 +5.0	–	–	–
	+ AD	71.7 +3.6	–	73.5 +3.5	–

Table 1: Our main results, presented as (mean) F1 scores where performance improvements from $2/5$-agreements are denoted via □ x. The gray area highlights combined scenarios (c) between our guideline approach and common few-shot methods.

We conducted our experiments five times (sam-

[5] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

ples were batched at random), therefore obtaining five prediction sets $\mathcal{P}_1, ..., \mathcal{P}_5$ for each sequence x. We present our main results in Table 1, demonstrating a) our obtained mean F1 scores, and b) the relative increase in performance when considering only mentions on which 2 out of the 5 annotators agree. We denote scores from such 2/5-agreements as $F_{2/5}$. The scenario *Basic + None* (obtaining no annotated exemplars) operates as our primary baseline, which is moreover expanded by:

a) annotated exemplars (e.g., *Basic + Random*)

b) additional guidelines (e.g., *Basic + D*)

c) both (a) and (b), as combined scenarios

We therefore examine our guideline approach (in b) in contrast to common few-shot methods (in a), and potential synergies between these methods (in c). For cost-related reasons, we examine only *Basic + {A, AD} + Similar* as combinations (c) for GPT-5. For statistics on inference cost, see Appendix D.

ΔF	Random	Similar	Medoids
nano	2.1 ± 0.50	3.9 ± 1.60	1.2 ± 0.78
mini	6.2 ± 2.90	8.7 ± 4.91	5.0 ± 4.60
gpt-5	3.5 ± 3.64	5.7 ± 4.32	4.3 ± 5.89

(a) Mean F1 increase from Few-Shot Strategies

ΔF	Basic +A	Basic +D	Basic +AD
nano	1.4 ± 0.89	1.4 ± 0.53	2.6 ± 1.90
mini	8.5 ± 3.64	3.1 ± 3.99	9.3 ± 5.48
gpt-5	7.7 ± 1.96	1.1 ± 0.73	8.2 ± 1.44

(b) Mean F1 increase from Guideline Approach

ΔF	Basic +A	Basic +D	Basic +AD
nano	4.3 ± 0.97	4.6 ± 1.92	5.1 ± 2.48
mini	13.4 ± 5.24	10.2 ± 6.65	13.6 ± 6.05
gpt-5	10.5 ± 3.17	–	10.8 ± 2.51

(c) Mean F1 increase upon including + *Similar*

Table 2: Mean F1 increase over baseline (*Basic + None*)

Few-Shot Strategies Naturally, the F1 results are improved across nearly all few-shot settings when compared to otherwise equivalent zero-shot scenarios. In agreement with related studies (Jiang et al., 2024; Wang et al., 2025a), our *Similar* examples facilitate exceptional improvements, surpassing *Random* and *Medoids* in most settings, thus confirming the potential of similarity-based examples. Notably, *Random* outperforms *Medoids* for various settings, suggesting a weak exemplar-selection for *Medoids*.

Table 2a presents the average F1 improvements over *Basic + None* per model and few-shot method. Noticeably, GPT-5 mini exhibits the highest F1 improvement, followed by GPT-5 and nano. However, the observed standard deviations are considerable, even with *Medoids* where demonstrations are static. These findings indicate that performance improvements from few-shot methods are not consistent.

Guideline Approach In Table 1, we observe that introducing additional guidelines consistently boosts model performance over otherwise equivalent Basic scenarios. The improvements are significantly more pronounced for +A (guidelines on boundary detection) than +D (type definitions), and generally most beneficial when combined (+AD). A notable exception is presented with GPT-5 nano, where +D and +A facilitate almost identical improvements. This, we believe, might confirm its diminished capacity for following instructions. Conversely, the improvements from additional guidelines are generally strongest for GPT-5, barring the 15.6 percent increase with GPT-5 mini on Litbank.

Table 2b presents the average increase over *Basic + None* per LLM used and guideline document. In comparison to the few-shot techniques (Table 2a), specifically our similarity-based example retrieval, our guideline-based approach obtains competitive (even superior) performance for GPT-5 and -mini, with relatively low standard deviations, suggesting more robust improvements than few-shot learning.

Guideline + Few-Shot Table 1 showcases that combining an extended guideline G (e.g., *Basic + A*) with demonstrations D significantly boosts F1 scores over individual guideline- or exemplar-only scenarios, especially for similarity-based examples. These results emphasize the usefulness of comprehensive annotation guidelines, even when promising examples are available, and *vice versa*. On average, our *Basic + AD* with *Similar* exemplars boosts F1 scores by 10.8, 13.6, and 5.1 percent over *Basic + None* for GPT-5, -mini, and -nano (see Table 2c).

2/5-Agreement Boosts results by 1.7 to 6.9 percent (4.0 ± 0.90), providing increases of 4.0 ± 0.56, 3.9 ± 1.37, and 4.0 ± 0.51 for ACE 2005, Litbank, and REFLEX, while yielding 3.7 ± 0.95, 4.3 ± 0.85, and 3.8 ± 0.82 percent for GPT-5, -mini, and -nano, respectively. This *significantly* emphasizes the importance of output aggregation–however simplistic.

External Baselines For reference, we compare our guideline approach with SOTA supervised and ICL approaches using ACE 2005 as an established

ACE 2005		P	R	F	$F_{2/5}$
BINDER	tuned	89.6	90.5	**90.0**	–
GoLLIE	tuned	–	–	89.6	–
GPT-NER (A)	32-shot	59.7	60.4	60.0	–
GPT-NER (B)†	16-shot	–	–	**76.3**	–
P-ICL†	20-shot	–	–	58.3	–
EvoPrompt	0-shot	–	–	51.2	–
EvoPrompt	14-shot	–	–	61.4	–
EL4NER	20-shot	–	–	69.3	–
Self-Improving†	0-shot	–	–	47.9	–
Self-Improving	16-shot	–	–	55.5	–
Basic	0-shot	76.0	58.2	65.9	70.6
+ AD	0-shot	79.9	68.4	73.7	77.1
+ AD (b = 1)	0-shot	82.0	76.3	**79.0**	**81.2**
+ AD + Similar	8-shot	81.5	70.8	75.8	78.8

Table 3: Our guideline approach (on GPT-5) compared with previous methods. We present the highest F1 scores in **bold**, and underline the second-best. The dagger (†) marks scores of Xiao et al. (2025b)'s re-implementation.

benchmark. Unfortunately, Litbank and REFLEX are, despite their similarity to ACE 2005, uncommon as benchmarks. Our comparison is, therefore, constrained to ACE 2005. For this, we include the *supervised* SOTA techniques **GoLLIE** (Sainz et al., 2024) and **BINDER** (Zhang et al., 2023c), and *in-context* methods **GPT-NER** (Wang et al., 2025a), **Self-Improving** (Xie et al., 2024), **P-ICL** (Jiang et al., 2024), **EvoPrompt** (Tong et al., 2025), and **EL4NER** (Xiao et al., 2025b). For additional information, see Appendix C.

The metrics are shown in Table 3. Notably, even with 8 samples per request and *zero-shot* learning, our guideline approach obtains SOTA performance among few-shot ICL methods. For 2/5-agreement, *Basic + AD* even outperforms GPT-NER (B) which requires an existing, fine-tuned entity detector and $|\mathcal{T}| + |\mathcal{P}|$ requests per sequence (see Appendix C). Further, providing *one* sample per request ($b = 1$), our method obtains 81.2 percent $F_{2/5}$ on ACE 2005, merely 8.8 percent behind BINDER (90.0 F1), the supervised SOTA.

5.1 Ablations

Ablation: Batch Size Hereby, we examine the influence of batch sizes 1, 4, and 16 relative to size 8. The experiments are, again, performed five-fold. We present the average and $F_{2/5}$ metrics in Table 4.

Noticably, performance is consistently increased as batch sizes are decreased, especially with GPT-5 mini, where F1 scores are enhanced by (up to) 9.5 percent over 8-batched samples. For GPT-5 with 2-out-of-5 agreements, we obtain 81.2, 76.3, and 79.9 percent F1 for ACE 2005, Litbank, and REFLEX.

ACE-2005	Ablation: Batch Size (Basic + AD + None)							
Model	**B. Size**	**P**	**Δ P**	**R**	**Δ R**	**F**	**Δ F**	**$F_{2/5}$**
gpt-5 (nano)	1	68.4	↑ 0.6	34.6	↑ 6.2	46.0	↑ 5.9	50.4
	4	69.0	↑ 1.1	30.6	↑ 2.2	42.3	↑ 2.3	46.6
	16	66.4	↓ 1.5	25.1	↓ 3.3	36.4	↓ 3.6	42.0
gpt-5 (mini)	1	76.3	↑ 5.7	65.7	↑11.4	70.6	↑ 9.2	72.9
	4	72.7	↑ 2.1	59.0	↑ 4.8	65.2	↑ 3.8	68.5
	16	68.6	↓ 2.0	49.7	↓ 4.5	57.7	↓ 3.7	61.4
gpt-5	1	82.0	↑ 2.0	76.3	↑ 7.9	79.0	↑ 5.3	81.2
	4	81.2	↑ 1.2	71.6	↑ 3.2	76.1	↑ 2.3	79.2
	16	78.6	↓ 1.4	63.8	↓ 4.6	70.4	↓ 3.3	74.8
Litbank	**Ablation: Batch Size (Basic + AD + None)**							
Model	**B. Size**	**P**	**Δ P**	**R**	**Δ R**	**F**	**Δ F**	**$F_{2/5}$**
gpt-5 (nano)	1	62.4	↓ 0.8	35.0	↑11.4	44.9	↑10.4	50.4
	4	62.4	↓ 0.7	27.5	↑ 3.8	38.1	↑ 3.7	42.1
	16	66.7	↑ 3.6	19.7	↓ 3.9	30.5	↓ 4.0	33.6
gpt-5 (mini)	1	70.0	↑ 1.7	67.3	↑11.4	68.6	↑ 7.1	70.9
	4	69.5	↑ 1.2	60.4	↑ 4.5	64.6	↑ 3.2	69.0
	16	67.8	↓ 0.5	50.1	↓ 5.8	57.6	↓ 3.9	63.1
gpt-5	1	75.3	↑ 0.3	74.8	↑ 6.5	75.1	↑ 3.5	76.3
	4	75.8	↑ 0.7	71.7	↑ 3.4	73.7	↑ 2.1	76.2
	16	75.9	↑ 0.8	63.8	↓ 4.5	69.3	↓ 2.2	72.8
REFLEX	**Ablation: Batch Size (Basic + AD + None)**							
Model	**B. Size**	**P**	**Δ P**	**R**	**Δ R**	**F**	**Δ F**	**$F_{2/5}$**
gpt-5 (nano)	1	66.1	↑ 1.3	31.5	↑ 7.4	42.6	↑ 7.5	46.7
	4	66.6	↑ 1.7	27.2	↑ 3.1	38.6	↑ 3.5	43.1
	16	65.4	↑ 0.5	22.2	↓ 1.9	33.2	↓ 2.0	38.5
gpt-5 (mini)	1	73.9	↑ 6.5	63.2	↑11.4	68.2	↑ 9.5	70.8
	4	70.5	↑ 3.1	57.1	↑ 5.2	63.1	↑ 4.5	67.2
	16	65.8	↓ 1.6	47.5	↓ 4.4	55.2	↓ 3.5	59.7
gpt-5	1	80.8	↑ 2.8	73.9	↑ 7.5	77.2	↑ 5.5	79.9
	4	79.1	↑ 1.1	69.7	↑ 3.3	74.1	↑ 2.4	77.0
	16	77.8	↓ 0.1	62.8	↓ 3.6	69.5	↓ 2.2	73.7

Table 4: The average metrics for different batch sizes (1, 4, and 16). The changes (Δ) are relative to batch size 8.

ACE-2005	Ablation: Basic Concepts							
Model	**Prompt**	**P**	**Δ P**	**R**	**Δ R**	**F**	**Δ F**	**$F_{2/5}$**
gpt-5 (nano)	Basic	67.2	↑ 0.5	27.0	↑ 0.4	38.5	↑ 0.5	43.0
	+ D	67.7	↑ 1.2	27.1	↓ 0.7	38.8	↓ 0.5	43.3
	Similar	69.9	↑ 1.2	29.2	↓ 0.1	41.2	↑ 0.1	45.3
gpt-5 (mini)	Basic	72.5	↓ 0.6	38.0	↓ 6.1	49.9	↓ 5.2	51.7
	+ D	71.9	↑ 3.5	37.7	↓ 9.8	49.5	↓ 6.6	51.3
	Similar	74.4	↓ 0.6	47.3	↓ 5.2	57.8	↓ 3.9	61.2
gpt-5	Basic	74.7	↓ 1.3	43.4	↓14.7	54.9	↓10.9	59.1
	+ D	73.5	↓ 1.0	42.9	↓17.6	54.2	↓12.6	58.4
	Similar	77.9	↓ 0.6	59.1	↓ 3.4	67.2	↓ 2.4	71.9
REFLEX	**Ablation: Basic Concepts**							
Model	**Prompt**	**P**	**Δ P**	**R**	**Δ R**	**F**	**Δ F**	**$F_{2/5}$**
gpt-5 (nano)	Basic	66.4	↑ 1.5	23.2	↑ 0.1	34.4	↑ 0.3	37.6
	+ D	67.3	↑ 4.0	23.7	↓ 0.5	35.0	↑ 0.0	38.0
	Similar	68.5	↑ 0.2	25.2	↓ 0.2	36.9	↓ 0.2	40.9
gpt-5 (mini)	Basic	71.0	↑ 0.5	35.9	↓ 6.2	47.7	↓ 5.0	49.6
	+ D	70.6	↑ 5.6	35.9	↓ 9.3	47.6	↓ 5.7	49.9
	Similar	70.9	↓ 0.2	44.1	↓ 4.4	54.4	↓ 3.3	58.7
gpt-5	Basic	73.1	↓ 1.2	43.8	↓13.4	54.8	↓ 9.9	60.0
	+ D	73.3	↓ 0.1	42.5	↓16.2	53.8	↓11.4	58.5
	Similar	76.3	↓ 0.4	55.4	↓ 4.9	64.1	↓ 3.3	69.0

Table 5: The (mean) scores for our ablated §1 Basic Concepts, with relative changes (Δ) to our original prompts.

Ablation: Baseline Notably, our primary baseline (*Basic + None*) already obtains SOTA results among the considered ICL methods in Table 3. We emphasize an important feature from §1 Basic Concepts (both ours and AGE's) that is commonly overlooked or ignored in related studies, including our

external baselines: Our guideline document *Basic* explicitly mentions that entity mentions can be *named, nominal, or pronominal* (see Figure 4, gray area). We examine that statement's importance by removing the gray-marked content from §1 Basic Concepts and repeating our *Basic* and *Basic + {D, Similar}* experiments for ACE 2005 and REFLEX.

As shown in Table 5, performance is considerably decreased for *Basic* and *Basic + D*, dropping by (up to) 12.6 percent F1 for GPT-5. Moreover, we observe a minimal decrease for GPT-5 nano (even a marginal increase), suggesting these guidelines are not used (or too confusing). For *Basic + Similar*, results are not impacted as much, as annotated exemplars still demonstrate nominal and pronominal entity mentions. In this scenario, our primary baseline achieves merely 54.9 and 54.8 percent F1 (on average) on ACE 2005 and REFLEX, thereby being surpassed by most external baselines.

6 Discussion

The results clearly showcase that including *detailed* guidelines into an LLM's instructions significantly improves the resulting performance, outperforming previous SOTA few-shot methods by 4.9 percent F1 on ACE 2005, *without* relying on existing labeled datasets and requiring (at most) one request per sample when predictions are not aggregated. In fact, our experiments demonstrate that thoroughly formulated instructions for detecting an entity mention's boundaries (+ *A*) provide *much* greater improvements than label descriptions and distinctions (+ *D*), where both, in combination (+ *AD*), usually achieve top performance. More generally speaking, we demonstrate that evaluating an LLM using inaccurate, dataset-agnostic task descriptions (e.g., *Basic*), thereby ignoring the concepts that governed the dataset's creation, does not meaningfully represent its extractive performance.

As presented in Section 4, our guideline approach moreover yields *considerable* performance improvements when used in conjunction with common few-shot methods, such as similarity-based demonstrations. These results clearly demonstrate the potential of detailed annotation guidelines for in-context learning, even when annotated datasets with promising exemplars are available. As demonstrated in Wang et al. (2025a), we believe that further improvements can be achieved by selecting the few-shot examples based on *mention*-level (as opposed to *sequence*) similarity. However, we leave these experiments for future works to explore.

Further, such a guideline approach is primarily applicable for annotating (additional) training samples for smaller, task-specific models, preferably from existing, *detailed* and expertly written annotation guidelines (such as the AGE). We repeatedly emphasize the word 'detailed' because adopting too nebulous guidelines, e.g., our primary baseline, bring about undesired, too contrasting task interpretations, therefore excluding or including important or irrelevant task-related aspects (such as *nominal* or *pronominal* entity mentions, see Section 5.1).

7 Conclusion

This work demonstrates that exhaustive, detailed annotation guidelines constitute an important yet underutilized mechanism for prompt-based entity detection with LLMs. By employing such dataset-specific task instructions, especially guidelines on boundary detection, our guideline-based approach yields considerable performance improvements for (nested) entity detection on benchmark datasets with guideline-compliant annotations. Our experiments demonstrate that, even with zero-shot constraints, such guideline-based approaches can outperform previous SOTA few-shot methods, significantly narrowing the performance gap between in-context learning and SOTA supervised approaches. Moreover, when combined, our guideline approach can furthermore augment few-shot demonstrations, improving performance even when labeled datasets are available.

Lastly, beyond performance, our findings indicate a broader methodical implication: *evaluating* LLMs using imprecise or dataset-agnostic task descriptions (as shown in our baseline ablation, Section 5.1) risks conflating genuine extractive capability with coincidental alignment (or misalignment) with annotation conventions. In contrast, by introducing more detailed instructions, performance assessments become more explainable, faithful, and reproducible.

Limitations

The usefulness of integrating annotation guidelines into task instructions is determined by a) the LLM's capacity to understand and execute the instructions (which commonly scales with model parameters), and b) the comprehensiveness and instruction granularity of the guideline document (see Section 4 and Table 5). For instance, GPT-5 nano improves

only narrowly when presented with extensive guidelines, obtaining at most 50.4 percent F1 on ACE 2005 when presented with one sample (per request) and 2/5-agreement for aggregation, whereas GPT-5 mini achieves 72.9 percent F1 under identical conditions (see Tables 2b and 4). As such, reproducing the experiments, particularly when targeting stronger performance, entails considerable hardware requirements. Furthermore, depending on or *following* their pre-training routines, LLMs cannot be guaranteed to properly execute the provided annotation guidelines without introducing or propagating some inherent, previously acquired biases.

Additionally, our experiments are conducted exclusively on English datasets for entity extraction. We make no claims regarding generalization to other tasks or languages. The selected datasets may or may not have been used to train the GPT-5 models. Regardless, our incremental experiments (from Basic to +AD) demonstrate significant improvements from employing annotation guidelines.

Acknowledgments

This work was funded by the European Union and the German state North Rhine-Westphalia within the following project of the European Regional Development Fund (EFRE-20801074): LLM4KMU - Optimized use of open source large language models in SMEs.

References

Tareq Al-Moslmi, Marc Gallofré Ocaña, Andreas L. Opdahl, and Csaba Veres. 2020. Named Entity Extraction for Knowledge Graphs: A Literature Overview. *IEEE Access*, 8:32862–32881.

Dhananjay Ashok and Zachary C. Lipton. 2023. PromptNER: Prompting For Named Entity Recognition.

Reza Averly and Xia Ning. 2025. Entity Decomposition with Filtering: A Zero-Shot Clinical Named Entity Recognition Framework. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2935–2951, Albuquerque, New Mexico. Association for Computational Linguistics.

David Bamman, Sejal Popat, and Sheng Shen. 2019. An Annotated Dataset of Literary Entities. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2138–2144, Minneapolis, Minnesota. Association for Computational Linguistics.

Necva Bölücü, Maciej Rybinski, and Stephen Wan. 2023. Impact of Sample Selection on In-Context Learning for Entity Extraction from Scientific Writing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5090–5107, Singapore. Association for Computational Linguistics.

Mikhail Budnikov, Anna Bykova, and Ivan Yamshchikov. 2025. Generalization Potential of Large Language Models. *Neural Computing and Applications*, 37:1973–1997.

Chenran Cai, Qianlong Wang, Bin Liang, Bing Qin, Min Yang, Kam-Fai Wong, and Ruifeng Xu. 2023. In-Context Learning for Few-shot Multimodal Named Entity Recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2969–2979, Singapore. Association for Computational Linguistics.

Qi Cheng, Liqiong Chen, Zhixing Hu, Juan Tang, Qiang Xu, and Binbin Ning. 2024. A Novel Prompting Method for Few-Shot NER via LLMs. *Natural Language Processing Journal*, 8:100099.

Alessio Cocchieri, Marcos Martínez Galindo, Giacomo Frisoni, Gianluca Moro, Claudio Sartori, and Giuseppe Tagliavini. 2025. ZeroNER: Fueling Zero-Shot Named Entity Recognition via Entity Type Descriptions. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 15594–15616, Vienna, Austria. Association for Computational Linguistics.

DeepSeek-AI. 2024. DeepSeek-V3 Technical Report.

Caoyun Fan, Jidong Tian, Yitian Li, Hao He, and Yaohui Jin. 2024. Comparable Demonstrations Are Important In In-Context Learning: A Novel Perspective On Demonstration Selection. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10436–10440.

Akshay Goel, Almog Gueta, Omry Gilon, Chang Liu, Sofia Erell, Lan Huong Nguyen, Xiaohong Hao, Bolous Jaber, Shashir Reddy, Rupesh Kartha, Jean Steiner, Itay Laish, and Amir Feder. 2023. LLMs Accelerate Annotation for Medical Information Extraction. In *Proceedings of the 3rd Machine Learning for Health Symposium*, volume 225 of *Proceedings of Machine Learning Research*, pages 82–100. PMLR.

Chuan Gou, Bangwei Li, Jianhua Dai, Xiaoyang Han, and Ming Cai. 2025. Mitigating Demonstration Bias through Global Coevolutionary Reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8564–8578, Vienna, Austria. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt,

Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The Llama 3 Herd of Models.

Feng Gu, Zongxia Li, Carlos Rafael Colon, Benjamin Evans, Ishani Mondal, and Jordan Lee Boyd-Graber. 2025. Large Language Models Are Effective Human Annotation Assistants, But Not Good Independent Annotators.

Sungwoo Han, Hyeyeon Kim, Jingun Kwon, Hidetaka Kamigaito, and Manabu Okumura. 2025. CodeNER: Code Prompting for Named Entity Recognition.

Tomáš Horych, Christoph Mandl, Terry Ruas, Andre Greiner-Petter, Bela Gipp, Akiko Aizawa, and Timo Spinde. 2025. The Promises and Pitfalls of LLM Annotations in Dataset Labeling: A Case Study on Media Bias Detection. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1370–1386, Albuquerque, New Mexico. Association for Computational Linguistics.

Yan Hu, Qingyu Chen, Jingcheng Du, Xueqing Peng, Vipina Kuttichi Keloth, Xu Zuo, Yujia Zhou, Zehan Li, Xiaoqian Jiang, Zhiyong Lu, Kirk Roberts, and Hua Xu. 2024a. Improving Large Language Models for Clinical Named Entity Recognition via Prompt Engineering. *Journal of the American Medical Informatics Association*, 31(9):1812–1820.

Zhentao Hu, Wei Hou, and Xianxing Liu. 2024b. Deep Learning for Named Entity Recognition: A Survey. *Neural Computing and Applications*, 36(16):8995–9022.

Shizhou Huang, Bo Xu, Yang Yu, Changqun Li, and Xin Alex Lin. 2025. GuideNER: Annotation Guidelines Are Better than Examples for In-Context Named Entity Recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(23):24159–24166.

Basra Jehangir, Saravanan Radhakrishnan, and Rahul Agarwal. 2023. A Survey on Named Entity Recognition — Datasets, Tools, and Methodologies. *Natural Language Processing Journal*, 3:100017.

Guochao Jiang, Zepeng Ding, Yuchen Shi, and Deqing Yang. 2024. P-ICL: Point In-Context Learning for Named Entity Recognition with Large Language Models.

Qiao Jin, Zheng Yuan, Guangzhi Xiong, Qianlan Yu, Huaiyuan Ying, Chuanqi Tan, Mosha Chen, Songfang Huang, Xiaozhong Liu, and Sheng Yu. 2022. Biomedical Question Answering: A Survey of Approaches and Challenges. *ACM Comput. Surv.*, 55(2).

Leonard Kaufman and Peter J. Rousseeuw. 1990. *Partitioning Around Medoids (Program PAM)*, chapter 2. John Wiley & Sons, Ltd.

Lvxue Li, Jiaqi Chen, Xinyu Lu, Yaojie Lu, Hongyu Lin, Shuheng Zhou, Huijia Zhu, Weiqiang Wang, Zhongyi Liu, Xianpei Han, and Le Sun. 2024a. Debiasing In-Context Learning by Instructing LLMs How to Follow Demonstrations. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7203–7215, Bangkok, Thailand. Association for Computational Linguistics.

Mingchen Li, Huixue Zhou, Han Yang, and Rui Zhang. 2024b. RT: A Retrieving and Chain-of-Thought Framework for Few-Shot Medical Named Entity Recognition. *Journal of the American Medical Informatics Association*, 31(9):1929–1938.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A Joint Neural Model for Information Extraction with Global Features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Junkai Liu, Jiayi Wang, Hui Huang, Rui Zhang, Muyun Yang, and Tiejun Zhao. 2024. *Improving LLM-Based Health Information Extraction with In-Context Learning*, pages 49–59. Springer Nature Singapore, Singapore.

Lefteris Loukas, Manos Fergadiotis, Ilias Chalkidis, Eirini Spyropoulou, Prodromos Malakasiotis, Ion Androutsopoulos, and Georgios Paliouras. 2022. FiNER: Financial Numeric Entity Recognition for XBRL Tagging. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

OpenAI. 2024. GPT-4o System Card.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training Language Models to Follow Instructions with Human Feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Nicholas Pangakis and Sam Wolken. 2024. Knowledge Distillation in Automated Annotation: Supervised Text Classification with LLM-Generated Training Labels. In *Proceedings of the Sixth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS 2024)*, pages 113–131, Mexico City, Mexico. Association for Computational Linguistics.

Pinal Patel, Disha Davey, Vishal Panchal, and Parth Pathak. 2018. Annotation of a Large Clinical Entity Corpus. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2033–2042, Brussels, Belgium. Association for Computational Linguistics.

Gabriele Picco, Leopold Fuchs, Marcos Martínez Galindo, Alberto Purpura, Vanessa López, and Hoang Thanh Lam. 2024. Description Boosting for Zero-Shot Entity and Relation Classification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9441–9457, Bangkok, Thailand. Association for Computational Linguistics.

Egil Rønningstad, Erik Velldal, and Lilja Øvrelid. 2024. A GPT among Annotators: LLM-based Entity-Level Sentiment Annotation. In *Proceedings of the 18th Linguistic Annotation Workshop (LAW-XVIII)*, pages 133–139, St. Julians, Malta. Association for Computational Linguistics.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2024. GoLLIE: Annotation Guidelines improve Zero-Shot Information-Extraction.

Stephanie Schoch and Yangfeng Ji. 2025. In-Context Learning (and Unlearning) of Length Biases. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7633–7671, Albuquerque, New Mexico. Association for Computational Linguistics.

Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large Language Models for Data Annotation and Synthesis: A Survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 930–957, Miami, Florida, USA. Association for Computational Linguistics.

Yongjian Tang, Rakebul Hasan, and Thomas Runkler. 2025. FsPONER: Few-shot Prompt Optimization for Named Entity Recognition in Domain-specific Scenarios.

Danil Tirskikh and Vasily Konovalov. 2023. Zero-Shot NER via Extractive Question Answering. In *Advances in Neural Computation, Machine Learning, and Cognitive Research VII*, pages 22–31, Cham. Springer Nature Switzerland.

Zeliang Tong, Zhuojun Ding, and Wei Wei. 2025. EvoPrompt: Evolving Prompts for Enhanced Zero-Shot Named Entity Recognition with Large Language Models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5136–5153, Abu Dhabi, UAE. Association for Computational Linguistics.

Petter Törnberg. 2024. Best Practices for Text Annotation with Large Language Models. *Sociologica*, 18(2):67–85.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, Relation, and Event Extraction with Contextualized Span Representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China. Association for Computational Linguistics.

Christopher Walker, Song Chen, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2009. REFLEX Entity Translation Training/DevTest. Philadelphia. Linguistic Data Consortium.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. ACE 2005 Multilingual Training Corpus. Philadelphia. Linguistic Data Consortium.

Anbang Wang, Difei Mei, Zhichao Zhang, Xiuxiu Bai, Ran Yao, Zewen Fang, Min Hu, Zhirui Cao, Haitao Sun, Yifeng Guo, Hongyao Zhou, and Yu Guo. 2024. ReverseNER: A Self-Generated Example-Driven

Framework for Zero-Shot Named Entity Recognition with Large Language Models.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025a. GPT-NER: Named Entity Recognition via Large Language Models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4257–4275, Albuquerque, New Mexico. Association for Computational Linguistics.

Zihan Wang, Ziqi Zhao, Yougang Lyu, Zhumin Chen, Maarten de Rijke, and Zhaochun Ren. 2025b. A Cooperative Multi-Agent Framework for Zero-Shot Named Entity Recognition. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, page 4183–4195, New York, NY, USA. Association for Computing Machinery.

Chenxiao Wu, Wenjun Ke, Peng Wang, Zhizhao Luo, Guozheng Li, and Wanyi Chen. 2024. ConsistNER: Towards Instructive NER Demonstrations for LLMs with the Consistency of Ontology and Context. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19234–19242.

Yuhui Xiao, Jianjian Zou, and Qun Yang. 2025a. Advancing Few-Shot Named Entity Recognition with Large Language Model. *Applied Sciences*, 15(7).

Yuzhen Xiao, Jiahe Song, Yongxin Xu, Ruizhe Zhang, Yiqi Xiao, Xin Lu, Runchuan Zhu, Bowen Jiang, and Junfeng Zhao. 2025b. EL4NER: Ensemble Learning for Named Entity Recognition via Multiple Small-Parameter Large Language Models.

Tingyu Xie, Qi Li, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2024. 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)*, pages 583–593, Mexico City, Mexico. Association for Computational Linguistics.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. 2024. Large Language Models for Generative Information Extraction: A Survey. *Frontiers of Computer Science*, 18(6):186357.

Faren Yan, Peng Yu, and Xin Chen. 2025. LTNER: Large Language Model Tagging for Named Entity Recognition with Contextualized Entity Marking. In *Pattern Recognition*, pages 399–411, Cham. Springer Nature Switzerland.

Rem Yang, Julian Dai, Nikos Vasilakis, and Martin Rinard. 2025. Evaluating the Generalization Capabilities of Large Language Models on Code Reasoning.

Longyi Ye and Huaping Zhang. 2025. TPL-NER: Three-Stage Prompt-Based Low-Resource Named Entity Recognition. In *Intelligent Multilingual Information Processing*, pages 425–438, Singapore. Springer Nature Singapore.

Andrew Zamai, Andrea Zugarini, Leonardo Rigutini, Marco Ernandes, and Marco Maggini. 2024. Show Less, Instruct More: Enriching Prompts with Definitions and Guidelines for Zero-Shot NER.

Meishan Zhang, Bin Wang, Hao Fei, and Min Zhang. 2024. In-Context Learning for Few-Shot Nested Named Entity Recognition. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10026–10030.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023a. LLMaAA: Making Large Language Models as Active Annotators. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13088–13103, Singapore. Association for Computational Linguistics.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023b. LLMaAA: Making Large Language Models as Active Annotators. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13088–13103, Singapore. Association for Computational Linguistics.

Sheng Zhang, Hao Cheng, Jianfeng Gao, and Hoifung Poon. 2023c. Optimizing Bi-Encoder for Named Entity Recognition via Contrastive Learning.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Guoyin Wang, and Fei Wu. 2025a. Instruction Tuning for Large Language Models: A Survey. *ACM Comput. Surv.*

Zikang Zhang, Wangjie You, Tianci Wu, Xinrui Wang, Juntao Li, and Min Zhang. 2025b. A Survey of Generative Information Extraction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4840–4870, Abu Dhabi, UAE. Association for Computational Linguistics.

Yufei Zhao, Xiaoshi Zhong, Erik Cambria, and Jagath C. Rajapakse. 2025. Large Language Models for Few-Shot Named Entity Recognition.

A Annotation Guidelines

Our complete guideline document G (summarized AGE) is outlined and presented in Figures 4, 5, and 6, where Figures 5 and 6 comprise the complete §3 Mention Type and §5 Entity Types.

Note: In our experiments, we replaced the brackets `[` and `]` (used for indicating entity mentions) with non-labeled tags `<E>` and `</E>`, surrounded with whitespace, because the desired output format also requires such markers (though labeled).

B Datasets

We evaluate our approach on ACE 2005 (Walker et al., 2006), Litbank (Bamman et al., 2019), and REFLEX (Walker et al., 2009):

ACE 2005 – This dataset (created by the LDC) is sourced from English, Chinese, and Arabic newswire, broadcast news and conversations, and telephone speech, among others, and was annotated for various information extraction tasks. ACE 2005 features 7 entity types: Person (`PER`), Organization (`ORG`), Geo-Political (`GPE`), Location (`LOC`), Facility (`FAC`), Vehicle (`VEH`), and Weapon (`WEA`). We primarily employ Lin et al. (2020)'s approach for pre-processing, and integrate all (nested) entities (not just the 'outermost' mentions). Lastly, we consider all 676 test sentences (in English).

Litbank – This dataset is comprised of 8562 sentences evenly extracted from 100 literary works (in English), and is primarily annotated via AGE – excluding the annotation of pronouns, interrogative words (e.g., *which*, *what*, and *whose*), and weapons (`WEA`). We account for these changes by modifying the corresponding rules and examples from G. The removed or changed sections and examples are marked in Figures 4, 5, and 6. Since Litbank provides no standard test split, we randomly select 500 sentences with 15 or more tokens for testing.

REFLEX – This dataset (created by the LDC) is sourced from English, Chinese, and Arabic newswire and weblog articles, and was annotated with (nested) entity mentions using AGE. For testing, we randomly sample documents (in English) from REFLEX until 500 sentences are collected, thus ultimately obtaining 503 instances. For pre-processing, we again adopt Lin et al. (2020)'s approach and integrate all nested entities. The following document ids have therefore been selected:

- `AFP_ENG_20030327.0224`
- `AFP_ENG_20030514.0706`
- `AGGRESSIVEVOICEDAILY_20041101.1806`
- `AGGRESSIVEVOICEDAILY_20041218.1004`
- `AGGRESSIVEVOICEDAILY_20050107.2012`
- `AGGRESSIVEVOICEDAILY_20050203.1356`
- `AGGRESSIVEVOICEDAILY_20050224.1207`
- `APW_ENG_20030311.0775`
- `APW_ENG_20030322.0119`
- `APW_ENG_20030327.0376`
- `APW_ENG_20030409.0013`
- `APW_ENG_20030410.0906`
- `APW_ENG_20030412.0531`
- `APW_ENG_20030502.0686`
- `FLOPPINGACES_20041115.1613.032`
- `FLOPPINGACES_20041228.0927.010`
- `FLOPPINGACES_20050101.2244.048`
- `MARKETVIEW_20041217.0801`
- `MARKETVIEW_20050127.0716`
- `MARKETVIEW_20050207.0746`

C External Baselines

We compare our guideline-based approach to SOTA supervised and ICL methods. Therefore, ACE 2005 serves as our benchmark.

- **BINDER** (Zhang et al., 2023c) – A bi-encoder framework for detecting entity mentions, mapping token spans and type descriptions into (shared) embedding spaces using contrastive learning, thus tackling entity detection from a representation learning perspective.

- **GoLLIE** (Sainz et al., 2024) – An LLM trained on various information extraction objectives, specifically instruction-tuned for complying with 'annotation guidelines' (i.e., entity type definitions). The evaluation (on ACE 2005) was performed fully supervised.

- **GPT-NER** (Wang et al., 2025a) – An ICL framework using kNN sequence- (version A) and span-level (version B; requires an existing NER tagger) representations for exemplar selection, followed by binary question-answering for self-verification. Overall, $|\mathcal{T}| + |\mathcal{P}|$ prompts are required per sequence, where $|\mathcal{P}|$ the number of extracted mentions.

- **P-ICL** (Jiang et al., 2024) – An ICL framework presenting both annotated sequences and various *point* entities (for each type) as demonstrations. To represent the dataset's distribution, these point entities are the centroids obtained by k-means clustering the known entities (per type).

- **EvoPrompt** (Tong et al., 2025) – An 'evolving prompts' framework using LLMs for continuous refinement of annotation guidelines (i.e., entity type definitions and distinctions) from (pseudo-)ICL demonstrations. The produced guidelines are then used for predicting subsequent pseudo-ICL demonstrations.

- **EL4NER** (Xiao et al., 2025b) – An ensemble ICL framework combining the predicted mentions from different LLMs. The approach improves performance and robustness by employing a span-level similarity approach for retrieving ICL demonstrations with subsequent self-verification to moderate ensemble noise.

- **Self-Improving** (Xie et al., 2024) – An ICL framework employing LLMs to pseudo-annotate datasets through self-consistency (SC) and removing any unreliable annotations. For inference, pseudo-labeled ICL demonstrations are retrieved via similarity-based example selection and SC-based ranking.

D Miscellaneous

Our algorithm for transforming the mentions identified in sequence $\hat{y}$ into (s, e, t) triples is presented in Algorithm 1. Following the AGE, our algorithm discards any leading (and trailing) tokens that contain *only* punctuation, see lines 12–13 (and 14–15).

Algorithm 1 Extract (s, e, t) Triples

Inputs: sequence $\hat{y}_1, ..., \hat{y}_m$, entity types $\mathcal{T}$
Output: entity spans (s, e, t) over target x

1: $\mathcal{P} \leftarrow \{\}$
2: $S \leftarrow \{t \mapsto \text{stack}() \mid t \in \mathcal{T}\}$
3: count $\leftarrow 0$
4: **for** $i = 1 \ldots m$ **do**
5: **if** $\hat{y}_i \in \mathcal{S}_\mathcal{T}$ **then**
6: $t \leftarrow \text{type}(\hat{y}_i)$
7: **if** $\hat{y}_i = <t>$ **then**
8: $S[t].\text{push}(i - \text{count})$
9: **else if** $S[t] = \emptyset$ **then**
10: $s \leftarrow S[t].\text{pop}()$
11: $e \leftarrow i - \text{count}$
12: **while** is_punct(x_s) **do**
13: $s \leftarrow s + 1$
14: **while** is_punct(x_e) **do**
15: $e \leftarrow e - 1$
16: **if** $s \leq e$ **then**
17: $\mathcal{P} \leftarrow \mathcal{P} \cup \{(s, e, t)\}$
18: count $\leftarrow$ count $+ 1$
19: **return** $\mathcal{P}$

We present our `json_schema` employed for enforcing our Equation 1 with OpenAI's Structured Outputs in Figure 3. For an open-source alternative, we recommend output format frameworks such as LM Format Enforcer.[6]

```
{
  "name": "output_struct",
  "strict": true,
  "schema": {
    "type": "object",
    "properties": {
      sent_id: {
        "type": "string",
        "pattern": P(x,T)
      }, ...
    },
    "required": [sent_id, ...],
    "additionalProperties": false
  }
}
```

Figure 3: Our `json_schema`, ensuring Equation 1 via Structured Outputs, where `sent_id` identifies an input sequence x (when batching) and `P(x,T)` represents our regular expression $\mathbf{P}(x, \mathcal{T})$.

In Table 7, we present detailed statistics about the inference cost incurred through our main experiments (see Table 1) based on OpenAI's pricing for BatchAPI requests (see Table 6).

Model	Input	Output
nano	$0.025	$0.20
mini	$0.125	$1.00
gpt-5	$0.625	$5.00

Table 6: The inference costs incurred by GPT-5-{nano, mini} and GPT-5 per 1M tokens when using OpenAI's BatchAPI.

[6] https://github.com/noamgat/lm-format-enforcer

Model	Prompt	ACE-2005		Litbank		REFLEX		Total
		input	output	input	output	input	output	
gpt-5 (nano)	Basic	$0.02	$0.78	$0.01	$0.46	$0.02	$0.51	$1.80
	Basic +A	$0.05	$0.87	$0.03	$0.49	$0.03	$0.60	$2.07
	Basic +D	$0.04	$0.86	$0.03	$0.56	$0.03	$0.57	$2.09
	Basic +AD	$0.07	$0.89	$0.05	$0.55	$0.05	$0.63	$2.24
gpt-5 (mini)	Basic	$0.08	$1.66	$0.06	$1.08	$0.07	$1.16	$4.11
	Basic +A	$0.24	$2.04	$0.17	$1.30	$0.19	$1.42	$5.36
	Basic +D	$0.20	$1.95	$0.15	$1.20	$0.16	$1.34	$5.00
	Basic +AD	$0.36	$2.13	$0.27	$1.29	$0.27	$1.48	$5.80
gpt-5	Basic	$0.41	$13.19	$0.29	$8.21	$0.33	$9.14	$31.57
	Basic +A	$1.03	$15.51	$0.75	$10.59	$0.76	$10.53	$39.17
	Basic +D	$0.87	$14.80	$0.63	$10.34	$0.64	$10.23	$37.51
	Basic +AD	$1.65	$16.02	$1.21	$11.09	$1.22	$11.14	$42.33
Total		$5.02	$70.70	$3.65	$47.16	$3.77	$48.75	$179.05

Table 7: The (average) costs incurred by our main experiments (see Table 1), averaged over the four considered ICL selection methods: *None*, *Random*, *Similar*, and *Medoids*.

Annotation Guidelines: Entity Detection

§1. Basic Concepts

The entity detection task requires that specific classes of entities mentioned in textual sources be detected and classified precisely. An *entity* represents something with definable existence or identity–such as an object, a concept, an event, a living being, or a location–that is referenced (i.e., *mentioned*) or distinguished via textual means.

Entities are commonly referenced via names, common nouns, or pronouns. For example:

Named – Bartleby, William Wilson
Nominal – the black cat, the captain of the Pequod
Pronominal – she, his, they, this, which, myself

Only entities from these seven types are *taggable*:

1. Person (PER)
2. Organization (ORG)
3. Geo-Political Entity (GPE)
4. Location (LOC)
5. Facility (FAC)
6. Vehicle (VEH)
7. Weapon (WEA)

§2. Entity Class

An entity mention can be referential, attributive, or negatively quantified.

Note: The following examples (and sections) use [brackets] to denote the extent of entity mentions. Further, these examples are provided to demonstrate *specific* annotation rules or concepts, and do, therefore, not necessarily represent *complete* annotations.

2.1 Negatively Quantified (NEG)

An entity mention is *negatively quantified* when its quantification yields or describes an empty set.

Examples:

- "There are [no confirmed suspects]."
- "[None of the books] are interesting."

However, the mention 'doctors' in "They are not [doctors]" is **not** negatively quantified, because it is introduced by a negated predicate.

2.2 Attributive/Ascriptive (ATR)

An entity mention is *attributive* when it assigns, relates, or ascribes some properties, features, or attributes to some (possibly other) entity.

Examples:

- "Richard is [an engineering specialist in Amsterdam]."
- "Dr. Leshner, [the director of NIDA, [a federal agency]]."

2.3 Specific Referential (SPC)

An entity mention is *specific referential* when it references a particular and distinctive (potentially unnamed) entity.

Examples:

- "[[John Doe]'s lawyer] is Bureau Chief for [the Washington Post]."
- "At [Columbia University] , [more than ten people] were injured."

2.4 Generic Referential (GEN)

An entity mention is *generic referential* when the reference is unspecific and indistinct, referring to some type or kind of entities instead.

Examples:

- "The construction of [aircraft] and [submarines] is supported by [governments]."
- "[Parents] and [officials] are enraged by [students]' violence against [teachers]."

2.5 Under-Specified Referential (USP)

An entity mention is *under-specified referential* when the reference is neither generic nor specific, often introduced by the following constructions:

- Quantified noun phrases in unasserted contexts where the entities in question cannot be verified (conditional, modal, future, uncertain, negated, question, hypothetical)
- Imprecise quantifications (such as *some* (plural), *most*, *more than half*, etc.)
- Mentions of a larger group of entities where the *actual* members are not necessarily identifiable

Examples:

- "Next year, [five people] will be elected." (future)
- "[How many buildings] are there?" (question)
- "[Some visitors] visited [several countries] ." (imprecise quantity)
- "[Sources] claimed... [Officials] reported..." (conflated identities)

§3. Mention Type

→ Figure 5

§4. Metonymy

4.1 Nickname Metonymy

Nickname Metonymy is a figure of speech in which a certain entity is referred to through the mention of something associated. In such cases, the metonymic mentions are annotated as mentions of the *referred* entities.

Two common examples are: a) the mention of a capital city or government seat in stead of the country's government, and b) the mention of a country or city to indicate sports teams.

Examples:

- "[Beijing] will not continue sales of anti-ship missiles to Iran."
- "[America] brought home the gold."

4.2 Cross-Type Metonymy

Cross-Type Metonymy occurs when multiple 'aspects' of an entity are referenced. For instance, a hospital may be understood as both an Organization and a Facility. In such cases, each mention is annotated according to the specific 'aspect' (type) it expresses.

Examples:

- "Wouters rushed to [St. John Hospital] . He was released an hour later, [hospital] spokeswoman Rebecca O'Grady said Thursday." Here, 'St. John Hospital' functions as a facility while 'hospital' in 'hospital spokeswoman' functions as an organization.

§5. Entity Types

→ Figure 6

Figure 4: Our guideline document (i.e., summarized AGE), presenting *Basic Concepts*, *Entity Class*, and *Metonymy*. Section §1 corresponds to **Basic**, Sections §2, §3, and §4 are introduced via **+A**, and Section §5 represents **+D**. The gray zone is ablated in Section 5. The green areas are removed/changed for Litbank (due to annotation differences).

§3. Mention Type

Entity mentions are oftentimes nested, containing other entity mentions.

Examples:

- "[The president of [Toyota Motor Corporation]] ."
- "[The students [who] entered [the new building]] ."

3.1 Simple Mentions

Simple entity mentions are usually (not always) complete noun phrases (NP). For each NP mention, we extract its entire *extent*. In addition, such mentions can assume (be encountered in) various linguistic shapes.

3.1.1 Mention Extent

The mention extent consists of the entire nominal phrase, including its modifiers (e.g., relative clauses and prepositional phrases). In cases where ambiguities are irresolvable (e.g., concerning the attachment of modifiers), the maximal extents should be annotated. For discontinuous constituents, the extent must include (i.e., span) the entire constituent, even if thus added tokens are not members of the constituent.

Examples:

- "I met [some people yesterday who love chess]."
- "She admired [the old house by the river]."

3.1.2 Mention Head

The head of an NP is its central (core) noun or pronoun. If the head of an NP consists of multiple tokens, only the final token is marked as the head. For proper names, however, the name's whole extent is marked as the head. **Note:** The following examples use underlining to mark the heads of entity mentions.

Examples:

- "The hurricane destroyed [the new glass-clad skyscraper]."
- "[Fred Smith] became [the new prime minister]."
- "The job fell to [Abraham Abercrombie III]."

3.1.3 Modified Conjoined Mentions

For conjoined mentions with common modifiers (pre- or post-), each head in the conjunction is considered a separate head mention.

Examples:

- "[20 angry men and women]"
- "[Bill Clinton and Jimmy Carter who are both former presidents]"
- "[The movers and shakers in Washington]"

3.1.4 Named Mentions (NAM)

Proper nouns and nicknames (not quantifiable).

Examples:

- "[Defense Secretary William Cohen]"
- "[The Jeluzoon Refugee Camp near [Ramallah]]"
- "[The U.S. Army 101st Airborne Division]"

3.1.5 Quantified Nominal Constructions (NOM)

Nouns which are modified or specified by a determiner, a quantifier, or a possessive.

Examples:

- "[Some American executives]"
- "[Thousands of corporations]"
- "[This year's Miss America]"

3.1.6 Unquantified (Bare) Nominal Mentions (BAR)

Unquantified nominal constructions.

Examples:

- "[Americans] were eagerly awaiting ..."
- "[corporations] in [foreign countries]"

3.1.7 Pronouns (PRO)

Pronouns; excluded are WH-question words and the specifier 'that'.

Examples:

- "[They] claimed that ..."
- "[These] can't drive fast."
- "[That]'s not the building." (demonstrative 'that')

3.1.8 WH-Question Words and Specifiers (WHQ)

WH-question words and the specifier 'that'.

Examples:

- "[Who] is the president of Brazil?"
- "Germany, [where] they were taken."
- "The Russian navy, [which] joined ..."

3.1.9 Headless Mentions (HLS)

Headless mentions are constructions without a nominal head. In these cases, we treat as head the right-most modifier that comes directly before the position where the head would normally be.

Examples:

- "[more than 30] on both sides"
- "[the toughest] were injured"
- "[sixty percent] said that ..."

3.1.10 Partitive Constructions (PTV)

A 'partitive' construction has two parts. The first part is headless and specifies the second part. Here, we treat as head the right most modifier that comes before the first part.

Examples:

- "[some of the lawyers]"
- "[sixty percent of them]"
- "the bottom of the ocean" (non-partitive)
- "a stockpile of weapons" (non-partitive)

3.2 Pre-Modifier Mentions

A pre-modifier mention is a taggable mention that premodifies another word or words, which (itself) may or may not be taggable. Such mentions are almost always constructed identically to NAM, NOM, or BAR constructions; exceptions are made for variations on NAM adjectives of Locations, Geo-Political Entities, Organizations, and Person-Groups (e.g., ethnicities and religious affiliations) in a pre-modifier position.

Examples:

- "[mountain] regions"
- "[Alpine] villages"
- "[Islamic] militants" (NAM adjective)
- "[democratic] senators" (NAM adjective)
- "Stalinist nations" (not taggable)
- "mountainous regions" (not taggable)

3.2.1 Nested Pre-Modifiers

The pre-modifier(s) of an NP can appear in nested form, most frequently when people (NAM or NOM) are modified by their titles and professions. The extent of a modified pre-modifier should capture the nested construction.

Examples:

- "[[[White House] [press] secretary] Scott McClellan]"

3.2.2 Post-Modifiers

A post-modifier mention is a taggable mention that postmodifies another word or words, which (itself) may or may not be taggable.

Examples:

- "All things [Seattle]"
- "[The two villages, [Arab] and [Jewish]]"

3.3 Complex Constructions

3.3.1 Appositive Constructions (APP)

Two or more taggable mentions that reference (or predicate on) the *same* entity or group of entities. The occurring mentions are tagged as appropriate, heads included. However, no head noun is assigned to the overarching APP-construction.

Examples:

- "[[Sauache County], [home of the Watchtower]]"
- "[[Mr. Black, 58], [a victim of the terrorist assault]]"
- "[[Egypt], [O Great, land of grace], [cradle of civilization]]"
- "[[Iliescu] and [Nastase], [ex-president] and [ex-prime minister]]"

3.3.2 Complex Constructions with Relative Clauses (ARC)

An ARC construction is an appositional construction with adjacent relative clause(s) that reference the initial, referent (SPC) mention instead of an attributive (ATR) mention. All occurring entity mentions and WH-mentions (e.g., which, who, whom) are tagged as appropriate, heads included.

Examples:

- "[[The former CEO], [John Fisher], [who] fired the board of directors] ..."
- "[[John Richards], [the party leader] [who] attended the DNC convention] said ..."

Figure 5: Section §3 from our complete guideline document (i.e., summarized AGE), introducing *Mention Type*. The contents highlighted in green are removed for Litbank (due to annotation differences). However, we replaced the contents in 3.1.7 and 3.1.8 with the following statement: ***Note:** {PRO, WHQ}-mentions should **not** be annotated!*

§5. Entity Types

This section provides further information and instructions on taggable entity types.

5.1 Person (PER)

PER entities involve a distinct person or groups of people which are mentioned by name, occupation, family relation, or pronoun, for example. Further, an expression that clearly describes or references a PER should be annotated as such. For example, "She is the brains of the family" contains the PER mention 'the brains of the family'.

5.1.1 Subtypes

A PER mention is classified via the following subtypes.

- **Individual** – The PER mention only involves a single person.
- **Group** – The PER mention involves more than a single person (not an ORG or GPE), including family names, and ethnic or religious groups.
- **Indefinite** – The PER mention is neither clearly an Individual nor a Group (lack of context).

5.1.2 Additional Guidelines

- Title (including religious ones) and honorifics are not considered to be an element of a person's name, and can, therefore, be annotated as individual PER mentions if they reference a taggable entity.
- A position (job title) should be annotated as PER if it indicates the person holding that position. For example, "US Secretary of State Madeline Albright" contains the PER mentions 'Secretary of State' and 'Secretary of State Madeline Albright'.
- A fictional character (if human) is taggable as PER, provided the mention does not occur as (part of) a title (e.g., TV show).
- Body parts are taggable as PER only if it represents the entire person (e.g., a corpse or severed head).

5.2 Organizations (ORG)

An ORG must have a (formally) established association, such as businesses, government units, sports teams, organized music groups, industries and industrial sectors.

5.2.1 Subtypes

An ORG mention is classified via the following subtypes. If none of these types match, the ORG mention in question cannot be annotated.

- **Government** – A government ORG is involved in governmental, political or state-related structures or processes (includes military ORGs of GPEs).
- **Commercial** – A commercial ORG is primarily focused on providing products or services for financial profit.
- **Educational** – An educational ORG is primarily focused on furthering, promulgating or promoting learning and education.
- **Entertainment** – An entertainment ORG is primarily focused on producing or delivering entertainment.
- **Non-Governmental Organizations** – A non-governmental ORG is primarily focused on advocacy, charity or politics.
- **Media** – A media ORG is primarily focused on distributing news or publications.
- **Religious** – A religious ORG is primarily focused on and devoted to aspects of religious worship.
- **Medical-Science** – A medical-science ORG is primarily focused on providing medical treatment or pursuing scientific research.
- **Sports** – A sports ORG is primarily focused on partaking in or governing organized sporting or gaming events at professional, amateur, or scholastic levels.

5.2.2 Additional Guidelines

- An entire GPE government should be annotated as GPE, not ORG.
- An ORG entity taking action should not be interpreted as a metonymical PER entity.
- ORG entities are sometimes mentioned using a (first person) pronoun, as in "our top story" when expressed by a representative of a broadcasting organization. In these cases, 'our' should be tagged as an ORG mention.

5.3 Geo-Political Entities (GPE)

GPE mentions are composite entities comprised of a population, a government, a physical location, and a jurisdiction (e.g., a nation, a state, a province); all mentions of these aspects of GPEs shall be annotated as GPE.

5.3.1 Subtypes

A GPE mention is classified via the following subtypes. If none of these types match, the GPE mention in question cannot be annotated.

- **Continent** – The GPE mention indicates an entire continent.
- **Nation** – The GPE mention indicates an entire nation.
- **State-or-Province** – The GPE mention indicates an entire nation's canton, state, or province.
- **County-or-District** – The GPE mention indicates an entire district, county, or prefecture.
- **Population-Center** – The GPE mention indicates an entity below counties or districts, such as cities, towns, villages, and boroughs.
- **GPE-Cluster** – Any named groups of GPEs that function as a (single) political entity.
- **Special** – This subtype is restricted to mentions of Palestine, the Palestinian Authority, Mainland China, Taiwan, and Native American reservations.

5.3.2 Additional Guidelines

- A reference to portions or segments of a government (e.g., "the Clinton Administration") is considered an ORG mention.
- Incidental, non-political clusters of GPEs should be marked LOC.
- Coalitions of governments, as well as the UN, should be marked ORG.
- A series of nested region names evoke one entity per region. For example, "Dallas, Texas" contains two mentions: 'Texas' and 'Dallas, Texas'.
- GPE mentions can assume or express various roles, such as representing a government (as if ORG), a population (as if PER), a location (as if LOC), or broader nations as a whole (as if GPE), thus capturing the GPE's function invoked via the text. However, such GPE mentions are annotated as GPE regardless.

5.4 Locations (LOC)

LOC entities are geographically or astronomically defined places, areas, points, or positions, that are not considered to be political entities, such as Mars, the solar system, the Hudson River, Mt. Everest, and Death Valley.

5.4.1 Subtypes

A LOC mention is classified via the following subtypes. If none of these types match, the LOC mention in question cannot be annotated.

- **Address** – The LOC is denoted as a fixed point (e.g., postal system, coordinates), or referred to by name in a postal system.
- **Boundary** – A boundary (one-dimensional) that separates GPEs or other LOCs, such as a nation's borders.
- **Celestial** – Any otherworldly or entire-world-inclusive locations.
- **Water-Body** – Any bodies of water, natural or man-made (artificial).
- **Land-Region-Natural** – A geologically or ecologically designated, non-artificial location.
- **Region-International** – A taggable LOC that crosses (or exceeds) national borders.
- **Region-General** – A taggable LOC that crosses (or exceeds) no national borders.

5.4.2 Additional Guidelines

- Places which are distinguished only by (the occurrence of) an event, such as "the scene of the murder" and "the site of the rocket launching", are not annotated as LOC.
- Any locative phrases, such as bottom, center, corner, southern, outskirts, periphery, etc., that pinpoint portions of taggable LOCs (or GPEs) are, likewise, taggable LOCs.
- Most location modifiers, location-ish expressions, and adjective or directional compass-points are not taggable as LOC. For example, "hiding behind a rock", "the lamp is back in place", and "they are going south" contain no taggable LOCs.

5.5 Facilities (FAC)

FAC entities are functional, primarily artificial (man-made) structures from the domains of architecture and civil engineering, such as facilities designed for (human) habitation, objects of similar size designed for storage, and elements of transportation infrastructure.

5.5.1 Subtypes

A FAC mention is classified via the following subtypes. If none of these types match, the FAC mention in question cannot be annotated.

- **Airport** – A taggable FAC whose primary purpose is serving as an airport.
- **Plant** – A building (or buildings) primarily designed for industrial purposes, such as power generation or manufacturing.
- **Building-or-Grounds** – A man-made (or -maintained) building or outdoor spaces, such as hotels, tents, ranches, or amusement parks.
- **Subarea-Facility** – Any taggable portions of FACs with an area that could contain a normal person, such as individual rooms.
- **Path** – A FAC facilitating the movement of people, signals, energy, fluids, or vehicles, such as streets, canals, pipes, cables, or bridges.

5.6 Vehicles (VEH)

VEHs are physical devices primarily designed to transport people or objects, e.g., by pushing, pulling, or carrying the transported object.

5.6.1 Subtypes

A VEH mention is classified via the following subtypes. If none of these types match, the VEH mention in question cannot be annotated.

- **Air** – A VEH designed to locomote primarily through the air, not touching land or water.
- **Land** – A VEH designed to locomote primarily on land.
- **Water** – A VEH designed to locomote primarily on (or submerged in) water.
- **Subarea-Vehicle** – Any taggable portions of VEHs with an area that could contain a normal person, such as boat decks, trunks, cockpits, or train wagons.
- **Underspecified** – A VEH whose subtype remains unspecified (lack of context), or VEH groupings with different subtypes.

5.7 Weapons (WEA)

WEAs are physical devices primarily designed (or used) to physically harm or destroy entities (taggable or not).

5.7.1 Subtypes

A WEA mention is classified via the following subtypes. If none of these types match, the WEA mention in question cannot be annotated.

- **Blunt** – A WEA primarily designed or used as bludgeoning instruments.
- **Exploding** – A WEA primarily designed or used to accomplish damage through explosions.
- **Sharp** – A WEA primarily designed or used to hack, cut, jab, or slash.
- **Chemical** – A WEA (substance or device) primarily designed or used for causing death or serious injury via poisonous or toxic chemicals, including their immediate precursors.
- **Biological** – A WEA (viruses, fungi, toxins, bacteria, or devices or means of dispersal) primarily designed or used to spread disease among humans, plants, or animals.
- **Shooting** – A WEA primarily designed or used for causing death or harm by launching projectiles at great speed.
- **Projectile** – A WEA primarily designed or used to be projected at great speed for causing harm.
- **Nuclear** – A WEA primarily designed or used for causing damage, harm, and death through the expenditure of radiological or nuclear energies.
- **Underspecified** – A WEA whose subtype remains unspecified (lack of context), or WEA groupings with different subtypes.

Figure 6: Section §5 from our guideline document G, presenting *Entity Types*. The contents (including the complete Section 5.7) marked in green are meaningfully excluded for Litbank. Furthermore, considering the literary content in Litbank, additional guidelines on *fictional characters* (see 5.1.2) are likewise omitted.