ProgGen: Generating Named Entity Recognition Datasets Step-by-step with Self-Reflexive Large Language Models

Yuzhao Heng **Chunyuan Deng** **Yitong Li**
Yue Yu **Yinghao Li** **Rongzhi Zhang** **Chao Zhang**
Georgia Institute of Technology
{stefan.heng,cdeng73,yli3277,yueyu,yinghaoli,rongzhi.zhang,chaozhang}@gatech.edu

Abstract

Although Large Language Models (LLMs) exhibit remarkable adaptability across domains, these models often fall short in structured knowledge extraction tasks such as named entity recognition (NER). This paper explores an innovative, cost-efficient strategy to harness LLMs with modest NER capabilities for producing superior NER datasets. Our approach diverges from the basic class-conditional prompts by instructing LLMs to self-reflect on the specific domain, thereby generating domain-relevant attributes (such as category and emotions for movie reviews), which are utilized for creating attribute-rich training data. Furthermore, we preemptively generate entity terms and then develop NER context data around these entities, effectively bypassing the LLMs' challenges with complex structures. Our experiments across both general and niche domains reveal significant performance enhancements over conventional data generation methods while being more cost-effective than existing alternatives.[1]

1 Introduction

Recently, Large Language Models (LLMs) have showcased their impressive capabilities across various NLP downstream tasks, including question answering, mathematical reasoning, and code generation (Touvron et al., 2023; Anil et al., 2023; OpenAI, 2023). Beyond these tasks, there is also a growing interest in exploiting LLMs' strong capabilities and extensive parametric knowledge for synthetic data generation. While previous research has predominantly focused on generating datasets for instruction tuning (Wang et al., 2023c; Xu et al., 2023b; Li et al., 2023b) or text classification tasks (Meng et al., 2022, 2023; Ye et al., 2022a,b; Gupta et al., 2023; Li et al., 2023c), there has been limited focus on producing high-quality datasets for information extraction (IE). This gap can be attributed to the deficiency of LLMs in handling the complex formats and relationships between entities (Li et al., 2023a; Wei et al., 2023).

Stemming from the limited capabilities of IE, LLMs also demonstrate inadequate performance in the domain of Named Entity Recognition (NER) (Xie et al., 2023; Zhou et al., 2024; Li et al., 2024). Achieving high performance in NER typically necessitates a supervised model trained on in-domain data, which is a requirement hard to meet in real-world scenarios. Leveraging the vast knowledge of LLMs to enhance the capabilities of small supervised models remains a challenging task. Several previous works (Ma et al., 2022; Zhou et al., 2022) augment training data through the masked language modeling (MLM) task. For example, Kim et al. (2022) employ a retrieval system to generate NER samples from open-domain QA datasets; Zhou et al. (2024) utilize LLMs to *annotate* a pre-existing corpus for supervised fine-tuning, a process that necessitates domain-specific corpora for application.

We introduce ProgGen, a new method that leverages the self-reflexive capabilities of LLMs to progressively synthesize NER datasets. Our method, in contrast to traditional approaches that directly generate synthetic NER datasets, operates through a sequential process involving multiple steps. To overcome the challenges posed by LLMs' limited ability to handle complex structures, we initially focus on generating entity terms, which are then employed for constructing the NER dataset. This step-by-step generation process allows for more effective dataset creation in new domains.

In addition to progressively generating NER datasets, we follow several recent studies to leverage LLMs' ability to self-reflect (Shinn et al., 2023; Madaan et al., 2023) from two perspectives. First, instead of employing simple class-conditional prompts, we leverage LLMs to generate

[1] Our code, prompts, and generated datasets are available at https://github.com/StefanHeng/ProgGen for reproducing our results.

Findings of the Association for Computational Linguistics ACL 2024, pages 15992–16030
August 11-16, 2024 ©2024 Association for Computational Linguistics

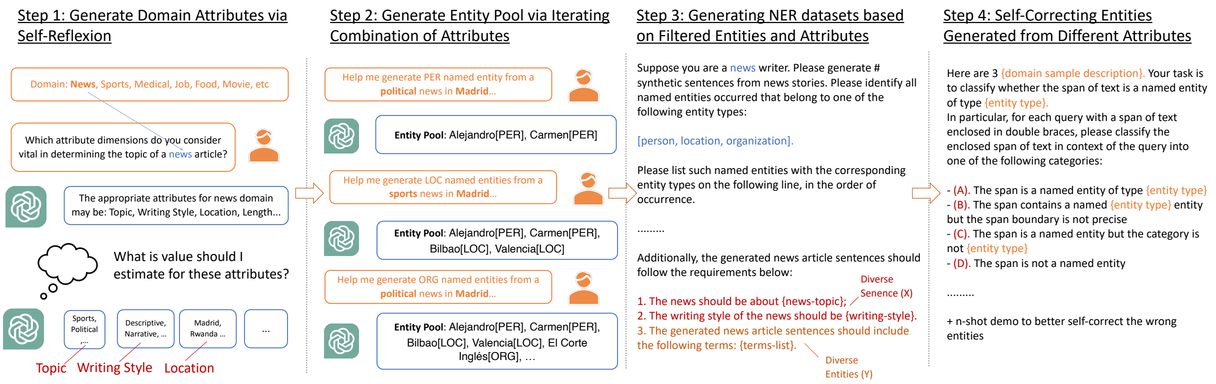


Figure 1: ProgGen NER data generation pipeline. Given a dataset domain, a set of interested entity classes with definitions, and a few demo samples, we prompt an LLM step-by-step to generate diverse NER samples. We leverage the generated samples to train a small model for NER.

domain-related attributes. For each attribute, we enumerate several values, iterating through combinations of these values to potentially generate a diverse and attributed dataset. Second, we adopt the self-reflection capability to examine the correctness of the entities it generates. These methods expand our usage from solely relying on the parametric knowledge of LLMs to leveraging their cognitive abilities to generate high-quality NER data.

Our findings reveal that a small supervised model, fine-tuned on our dataset, demonstrates robust performance across both general-domain and new-domain datasets in a low-resource context. Additionally, our approach exhibits cost-efficiency advantages over traditional prompt methods or LLMs employing in-context learning. We summarize our key contributions as follows:

1. *Novel Progressive Generation Method* - We deploy a step-wise method starting from generating entities for subsequent NER sample generation, where we also leverage LLMs' cognitive ability to self-flex on attributes and self-correct wrong NER samples.
2. *Low-Resource Utilization* - Our approach minimizes reliance on human-annotated samples, substantially smaller compared to the datasets required by prior models and are cheap in terms of inference cost, making our approach particularly advantageous in scenarios where large annotated datasets are unavailable or impractical to compile.
3. *Enhanced Efficiency and Performance* - Our downstream model are efficient compared to LLM inference and exhibits superior performance in specialized domains, surpassing the standard data generation baseline by 8% in F1 score in the best case.

2 Related Work

LLMs have opened new avenues for generating synthetic data, as highlighted by recent studies (Meng et al., 2022, 2023; Ye et al., 2022a,b; Yu et al., 2023b; Gupta et al., 2023; Li et al., 2023c), mainly for text classification tasks. However, the use of overly simplistic and generic prompts in these methods often yields suboptimal results, as the representativeness and the diversity of the synthetic data. To overcome this, some studies have adopted interactive learning strategies to enhance instance generation (Liu et al., 2022a; Zhang et al., 2022a; Yu et al., 2023a; Chung et al., 2023), albeit at the expense of increased human intervention.

Generating synthetic data for IE is challenging due to the need for detailed consideration of entities and their relationships. To address this, some studies incorporate additional rules (Zhang et al., 2022b) or knowledge graphs (Chia et al., 2022; Josifoski et al., 2023) to assist IE synthetic data generation. Xu et al. (2023a) propose a two-stage training pipeline to alternately learn from synthetic and golden data together. For the NER task studied in this work, Liu et al. (2022b); Zhou et al. (2022) used masked language modeling to augment the training data, GeNER (Kim et al., 2022) leverages an entity retrieval model to create entity labels. Recently, several works studied to generate synthetic text with LLMs, but focus on general-domain or biomedical NER with fixed entity type (Guo and Roth, 2021; Hiebel et al., 2023; Tang et al., 2023; Xu et al., 2023c). These approaches, however, do not fully accommodate the diversity of entity types encountered in broader, open-world scenar-

Template	Prompt
Simple Prompt	*Instruction*: Suppose you are a news writer. Please generate `{#generate}` synthetic sentences from news stories. Please identify all named entities occurred that belong to one of the following entity types: [person, location, organization]. Please list such named entities with the corresponding entity types on the following line, in the order of occurrence. If no entity is found in the generated sentence, leave the brackets empty. *Examples*: Here are some example sentences and annotations for your reference. Please follow this format.\n Examples:\n Sentence: "Saudi Arabia executes Pakistani man." Named Entities: [Saudi Arabia (location)]\n ...
Diversify X	`{Simple Prompt}`\n Additionally, the generated news article sentences should follow the requirements below: 1. The news should be about `{news-topic}`; 2. The writing style of the news should be `{writing-style}`.
Diversify Y	`{Simple Prompt}`\n Additionally, in the generated sentences, include the following terms: [`{terms-list}`].
Diversify X + Y	`{Simple Prompt}`\n Additionally, the generated news article sentences should follow the requirements below: 1. The news should be about `{news-topic}`; 2. The writing style of the news should be `{writing-style}`. 3. The generated news article sentences should include the following terms: [`{terms-list}`].

Table 1: NER Sample Generation prompt templates for the CoNLL-2003 dataset. We append explicit diversity requirements to the standard sample-generation prompt.

ios. Zhou et al. (2024) also studied creating NER instruction tuning data, but they focused on using LLMs to *label* the corpus to create data for supervised finetuning, which diverges significantly from the synthetic data generation focus of our study.

3 Method

Our workflow is succinctly depicted in Figure 1. At the outset, we engage LLMs in a sequential process to generate a variety of NER samples. Unlike traditional methods that rely on domain-specific, unlabeled corpora, our approach is more pragmatic and resource-efficient. We utilize only a handful of examples to prompt LLMs to produce sentences along with their corresponding named entity annotations. This method is exemplified in Table 1, which showcases a standard few-shot prompt designed for NER sample generation[2]. Our procedure involves requesting LLMs to produce multiple samples within a single completion to enhance efficiency. Subsequently, we carefully select a subset of entity annotations identified by LLMs as particularly challenging. These annotations are re-annotated by the same LLM, this time with the aid of additional, manually crafted annotation guidelines and a selection of representative examples specifically chosen for this task. To perform NER tasks, we utilize DeBERTa (He et al., 2021) trained on the generated dataset, demonstrating the practicality and effectiveness of our low-resource, LLM-based approach for NER sample generation.[3]

3.1 Problem Setup

Our setup is formalized as follows. We are given a brief task domain description d, target named entity classes $\mathcal{Y}$ with their definitions, and a few demo samples $\mathcal{D}_{\text{demo}} = \{(x_i, y_i)\}_{i=1}^{n}$ where $n < 10$. Each sample consists of a sentence x_i paired with an entity list $y_i = (e, c)^{n_i}$, consisting of contiguous text spans in x_i and their corresponding entity classes. Utilizing these information, we prompt LLMs in a step-wise fashion and process the LLMs' responses into NER samples $\mathcal{D}_{\text{gen}}$. The aggregated dataset $\mathcal{D}_{\text{gen}}$, combined with the demo samples $\mathcal{D}_{\text{demo}}$, is used to train a small model for NER. The model is then evaluated on an unseen test set.

3.2 Sample Diversity

To enhance the diversity of our generated NER samples, we augment a standard NER-sample-generation prompt (Simple Prompt) by incorporating explicit diversity requirements for the generated samples through configuration sampling. We propose 4 variants aiming to enrich sentence semantics and entity variety (examples in Table 1). Processed diversity requirement pool with statistics are available at Appendix E.1.

3.2.1 Diversify Sentence

We study imposing explicit requirements on various semantic aspects of the generated sentences (Diversify X). Inspired by several previous work to generate diverse data (Yu et al., 2023a; Chung et al., 2023), we generate attribute dimensions and values given the dataset domain in a *human-AI collaboration* fashion (illustrated in Figure 2).

[2]NER sample representation were searched over 4 choices as detailed in Appendix B.1.

[3]See Appendix F for additional prompts and generated datasets used in our study.

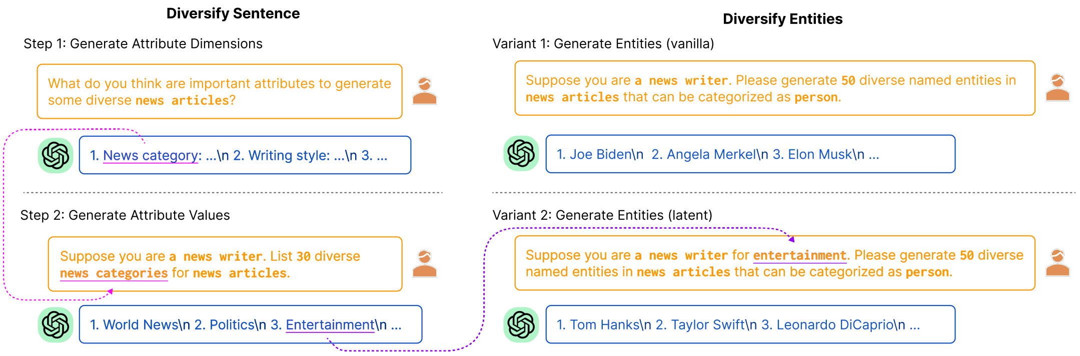


Figure 2: Diversity Requirement Generation workflow. **(Left)** Diversify Sentence: LLMs are prompted to generate attribute dimensions first and then attribute values given each dimension. **(Right)** Diversify Entities: LLMs are prompted to generate named entities for entity class, optionally conditioned on a domain-specific "topic" category.

Generate Attribute Dimension We first prompt LLMs to generate diverse attribute dimensions given the dataset domain. Given that this step consumes relatively less tokens and thus less cost, we leverage the more capable GPT-4 model. We manually review and select suitable attribute dimensions for subsequent steps.

For each dataset, we also include a "sample topic/category" dimension , such as "news topic" and "movie query category", leveraged for subsequent diversity variants (§3.2.2 & §3.2.3).

Generate Attribute Values In this step, LLMs are prompted to generate potential values for each selected attribute dimension, where we examine and filter out low-quality values. We utilize GPT-3.5 for its cost-effectiveness.

Construct Diversity Requirement We sample each generated attribute dimension at a predetermined probability such that the expected number of requirements in a sample-generation prompt R_x is small ($\in [1, 2]$). We then sample an attribute value for each sampled attribute dimension. The diversity requirement templated from the sampled dimension-value pairs.

3.2.2 Diversify Entities

We study explicit requirements on inclusion of named entities in the generated sentences (workflow illustrated in Figure 2).

Generate Entities We generate named entities by prompting GPT3.5 given dataset domain (Diversify Y). Similarly to the Diversify Sentence process, we filter out low-quality generated entities based on manual inspection.

We explore two variants, (1) *vanilla* where we generate M entities for each entity class, and (2) *latent* where we generate M entities conditioned on each entity class and additionally on each domain-specific topic value (via the dataset-independent "topic" attribute).

Construct Diversity Requirement We sample from the generated entity pool with an expected number of exampled entity R_y. We then construct the diversity requirement from the sampled named entities via a template. We omit the entity type corresponding to each sampled entity so that LLMs have more freedom in entity type annotations[4].

3.2.3 Diversify Both Sentence and Entities

We provide explicit requirements on both sentence semantic and entity inclusion (Diversify X + Y). We first construct the sentence requirements, then we use the sampled "topic" attribute value to sample named entities from the corresponding *latent* entity pool such that the diversity requirements are semantically consistent.

3.3 LLM Self-Correction

Creating NER datasets critically involves defining entities and providing annotation guidelines for edge cases to ensure clarity (Zhou et al., 2024). These guidelines often vary across datasets, even for the same entity types (e.g., Location in news vs in restaurant queries). Given LLMs' ability to learn from feedback and correct their own mistakes (Shinn et al., 2023; Madaan et al., 2023), we tailor instructions and examples for each dataset and entity class, directing LLMs to refine their annotations, aiming to align more closely with dataset-specific nuances. Further, to enhance cost efficiency, only a subset of annotations that LLMs

[4] We found that this lead to performance gains in preliminary experiments. Additional discussion in Appendix D.2.2.

Instruction: Here are {#correct} {domain sample description}. Your task is to classify whether the span of text is a named entity of type {entity type}.
In particular, for each query with a span of text enclosed in double braces, please classify the enclosed span of text in context of the query into one of the following categories:
- (A). The span is a named entity of type {entity type}
- (B). The span contains a named {entity type} entity but the span boundary is not precise
- (C). The span is a named entity but the category is not {entity type}
- (D). The span is not a named entity
If the label is (B), you should provide the correct span boundary.
If the label is (C), you should provide the correct entity type. Pick an entity type from the following list:
[{remaining entity types in the dataset}, other].\n

{entity-type-specific annotation instruction}\n

Examples: Here are some example span classifications for your reference. Please follow this format.\n
Examples:\n
{(sentence, span, correction) demo examples}\n

Please classify the following {#correct} spans of text:\n
{(sentence, entity span) samples to correct}

Table 2: LLM Self-Correction prompt template. For each entity annotation prompted, LLMs are instructed to optionally (1) correct the entity span, (2) correct the entity type, or 3) drop the annotation.

find "challenging" are selected for self-correction.

3.3.1 Annotation Uncertainty Ranking

We identify challenging samples using a scoring function that averages the log-probability of tokens within each entity annotation, i.e. `entity span (entity type)`, where a lower score means higher uncertainty and thus more challenging. Preliminary experiment compares this function with averaged token-wise loss values from a downstream model trained on the generated data (Appendix B.3), and shows that self-correction with the former scoring function yields better downstream performance.

3.3.2 Prompt Construction

We stratify all annotations to self-correct by the entity type annotated, allowing for more precise annotation instructions given the same token budget. Conditioned on the entity class, an LLMs are given (context, entity) pairs and are instructed to optionally improve the annotation by (1) correcting the entity span boundary, (2) correcting the entity type, and (3) dropping the entity annotation (template shown in Table 2). We batch correction samples for cost efficiency.

We construct the entity-type-specific prompts manually by alternating between the following 2 steps. (1) We examine top-uncertain entity annotations for each generated dataset to identify common error cases in LLMs' annotations. We write correction instructions including common error cases. We hand-picked diverse demo samples from the generated datasets that represent the most common failure cases. (2) In case of uncertain annotation or class definition ambiguities (especially a problem for specific-domain datasets), since we were not able to find detailed annotation schemas for the datasets we studied, we inspected the original samples in the dataset to infer the annotation guidelines. We then manually write the entity definitions and annotation instructions.

Dataset	Domain	#sample	C	D
CoNLL-2003*	News	15K / 3.5K / 3.7K	3	4
WikiGold*	Wikipedia	1.7K	3	3
MIT Movie	Movie	9.7K / 2.4K	12	7
MIT Restaurant	Restaurant	7.7K / 1.5K	8	5

Table 3: Dataset Statistics. #sample is provided for the original train/dev/test or train/test splits respectively if available. C and D refer to the number of entity classes and the number of demo samples we use respectively.

Further analysis of ProgGen design choices are available in Appendix B.

4 Experiments

We assess the effectiveness of our framework via downstream model performance on generated NER datasets.

4.1 Datasets

We examine datasets from various domains and with different number of entity classes, including:

- **CoNLL-2003** (Tjong Kim Sang and De Meulder, 2003) features news articles from Reuters with three broadly-defined entity types: person, location and organization.
- **WikiGold** (Balasuriya et al., 2009) comprises Wikipedia articles. It's a small dataset with the same set of entity types as CoNLL-2003.
- **MIT-Movie** and **MIT-Restaurant** (Liu et al., 2013) contain dialog system queries related to movies and restaurants, respectively. Both datasets incorporate a wide range of entity types, such as movie titles, genres, characters, and names of restaurants and cuisines.

The dataset statistics[5] are summarized in Table

[5]Following prior studies (Li et al., 2021; Kim et al., 2022),

	Method	CoNLL-2003		WikiGold		MIT-Movie		MIT-Restaurant		Average	
		F1	Price	F1	Price	F1	Price	F1	Price	F1	Price
Few-Shot	Few Shot-FT (Huang et al., 2021)	61.4	—	64.0	—	38.0	—	44.1	—	51.9	—
	SpanNER (Wang et al., 2021)	71.1	—	—	—	65.4	—	49.1	—	—	—
	Ent-LM (Ma et al., 2022)	66.8	—	—	—	59.4	—	—	—	—	—
Data Gen.	GeNER (Kim et al., 2022)	71.0	—	72.5	—	—	—	—	—	—	—
	LLM-DA (Ye et al., 2024)	54.6	—	—	—	58.9	—	—	—	—	—
	Simple Prompt (ours)	72.0	0.1	75.8	0.16	64.8	0.13	58.4	0.13	67.8	0.13
ProgGen (ours)	Diversify X	69.2 -2.8	0.24	75.5 -0.3	0.33	65.0 +0.2	0.36	54.7 -3.7	0.31	66.1 -1.7	0.31
	+ Self-Correction	71.9 -0.1	0.33	77.3 +1.5	0.44	63.5 -1.3	0.50	56.1 -2.3	0.55	67.2 -0.6	0.46
	Diversify Y (vanilla)	71.2 -0.8	0.24	76.2 +0.4	0.31	73.0 +8.2	0.35	61.5 +3.1	0.30	70.5 +2.7	0.30
	+ Self-Correction	71.4 -0.6	0.32	**78.2** +2.4	0.47	**73.3** +8.5	0.51	**62.8** +4.4	0.52	71.4 +3.6	0.46
	Diversify Y (latent)	75.6 +3.6	0.24	75.7 -0.1	0.32	71.8 +7.0	0.35	60.9 +2.5	0.30	71.0 +3.2	0.30
	+ Self-Correction	**77.5** +5.5	0.33	76.4 +0.6	0.47	72.9 +8.1	0.52	61.4 +3.0	0.52	**72.1** +4.3	0.46
	Diversify X + Y	73.7 +1.7	0.27	76.1 +0.3	0.35	70.6 +5.8	0.39	59.5 +1.1	0.33	70.0 +2.2	0.34
	+ Self-Correction	74.3 +2.3	0.35	74.6 -1.2	0.48	72.2 +7.4	0.54	59.8 +1.4	0.59	70.2 +2.4	0.50
	Few-shot ICL	71.5	0.39	74.5	0.49	74.2	0.61	67.0	0.48	71.8	0.49

Table 4: ProgGen Performance Comparison. Price refers to the API querying cost ($). The prices are for 1) Data Generation: Generating around 1.5K samples added by the cost for self-correction, and 2) Inference: few-shot learning prediction for 1.5K samples. F1 gains/drops are w.r.t Simple Prompt. Relative performance drops after Self-Correction are colored.

3. For datasets that originally contain over 1% negative samples, i.e. samples with no entity annotations, we include a negative sample in the set of demo samples to enhance LLMs learning what's *not* a relevant named entity.

4.2 Experiment Setup

Data Generation We use 1-shot demo as described in Ma et al. (2022), wherein we include one entity annotation for each entity class. In the attribute dimension generation step for *Diversify X*, we prompted the ChatGPT4 web App 3 times. All other generations are from ChatGPT3.5 through OpenAI API[6]. We generate ~1.5K training samples for each dataset and for each method. We drop duplicates and filter out malformed samples (details in Appendix D.1). See Appendix E and F for generated datasets samples and statistics. We select all generated entity annotations with log-probability below $P_c = -2e^{-2}$ for LLM self-correction, capped at $T_c = 20\%$ of all entity annotations.

Downstream Model Training For all setups, we train a DeBERTaV3 (He et al., 2023) model for 16 epochs with batch size 24, utilizing the `microsoft/deberta-v3-base` checkpoint from HuggingFace. We train with the AdamW optimizer (Loshchilov and Hutter, 2019), a learning rate of $4e^{-5}$, a weight decay of $1e^{-4}$, and a linear scheduler with warmup over the first 200 steps. The 1-shot demo samples are integrated into training with a weight of 5. Following prior work (Ma et al., 2022; Yu et al., 2023a), we don't use a validation set and directly evaluate the final trained model with micro-average F1. For our approaches, we also report precision and recall.

Baselines We compare our approach with the existing few-shot baselines: (1) Few-shot FT (Huang et al., 2021), which directly finetunes a pretrained language model on few-shot examples. (2) SpanNER (Wang et al., 2021), which learns from natural language supervision to identify entity classes. (3) Ent-LM (Ma et al., 2022), which leverages masked language modeling to augment the few-shot examples. We also consider the following data generation baselines: (4) GeNER (Kim et al., 2022), which uses an entity retrieval module to retrieve from unlabeled corpus to create NER training data. (5) LLM-DA (Ye et al., 2024), which prompts GPT-3.5 to rewrite the context and entities of few-shot samples to augment training data. (6) Our standard data generation prompt (Simple Prompt). Besides, we provide few-shot in-context learning (ICL) on the same set of demonstration samples for reference. We use reported results from prior work.

See Appendix D for additional implementation details.

4.3 Results

Tables 4 and 5 display our main results. We first note that the Simple Prompt baseline is strong and consistently out-perform prior works in few-shot NER, stemmed from the strong base knowledge

the miscellaneous (MISC) class is omitted due to its lack of a precise definition. Datasets originally containing the MISC entity class are denoted with asterisks*.

[6]We used the `gpt-3.5-turbo-1106` checkpoint.

Method		CoNLL-2003	WikiGold	MIT-Movie	MIT-Restaurant	Average
Simple Prompt		67.2 / 77.4	69.2 / 83.7	62.3 / 67.4	55.4 / 61.8	63.5 / 72.6
ProgGen	Diversify X	63.8 -3.4 / 75.6 -2.8	71.0 +1.8 / 80.6 -3.1	63.4 +1.1 / 66.6 -0.8	52.1 -3.3 / 57.6 -4.2	62.6 -0.9 / 70.1 -2.5
	+ Self-Correction	67.5 +0.3 / 77.0 -0.4	73.6 +4.4 / 81.4 -2.3	62.4 +0.1 / 64.5 -2.9	53.9 -1.5 / 58.5 -3.3	64.4 +0.9 / 70.4 -2.2
	Diversify Y (vanilla)	68.2 +1.0 / 74.5 -2.9	73.8 +4.6 / 78.7 -5.0	71.7 +9.4 / **74.4 +7.0**	60.4 +5.0 / 62.7 +0.9	68.5 +5.0 / 72.6 +0.0
	+ Self-Correction	69.2 +2.0 / 73.7 -3.7	**77.9 +8.7** / 78.5 -5.2	**73.0 +10.7** / 73.7 +6.3	**61.9 +6.5 / 63.4 +1.6**	70.5 +7.0 / 72.3 -0.3
	Diversify Y (latent)	72.7 +5.5 / 78.8 +1.4	74.8 +5.6 / 76.7 -7.0	70.0 +7.7 / 73.7 +6.3	59.9 +4.5 / 61.9 +0.1	69.4 +5.9 / 72.8 +0.2
	+ Self-Correction	**75.0 +7.8 / 80.2 +2.8**	75.8 +6.6 / 77.1 -6.6	71.9 +9.6 / 74.1 +6.7	60.5 +5.1 / 62.2 +0.4	**70.8 +7.3 / 73.4 +0.8**
	Diversify X + Y	69.2 +2.0 / 78.8 +1.4	75.0 +5.8 / 77.3 -6.4	69.3 +7.0 / 71.9 +4.5	58.5 +3.1 / 60.4 -1.4	68.0 +4.5 / 72.1 -0.5
	+ Self-Correction	71.8 +4.6 / 76.9 -0.5	74.0 +4.8 / 75.2 -8.5	72.0 +9.7 / 72.4 +5.0	59.3 +3.9 / 60.3 -1.5	69.8 +5.3 / 71.2 -1.4
Few-shot ICL		65.8 / 78.3	70.2 / 79.3	73.2 / 75.3	64.8 / 69.4	68.5 / 75.6

Table 5: ProgGen Performance Comparison. We show precision and recall separated by slash (precision / recall).

of LLMs. Also, few-shot ICL have recall values higher than most data generation approaches, reflecting loss of world knowledge when distilling LLMs to generated data.

We further note that (1) our best-performing setup matches or outperforms ICL in 3 out of 4 datasets we study. In general domains (News, Wikipedia), even Simple Prompt out-performs ICL, indicating a small pretrained BERT model can learn from noisy LLM annotations. (2) Batching sample- and correction-generation achieves roughly the same cost w.r.t. ICL inference. Data generation cost is one-time (see additional cost analysis in Appendix A.4), whereas few-shot ICL incurs inference API querying cost for each new prediction sample. Combined with the fact that (3) a small, task-specific downstream model has negligible inference overhead (c.f. LLMs), we highlight the efficacy of data generation for NER tasks.

4.3.1 Sample Diversity

Incorporating diversity notably improves model performance, outperforming Simple Prompt across all datasets in most cases. Diversify Y variants are the best-performing, yielding at least 2% F1 gains on average and up to 8% improvement for nuanced domains. The *latent* variant is more robust with higher average performance and more consistent gains over Simple Prompt, which may benefit from the implicit diversity in sentence semantics rooted from the latent "topic" dimension. Conversely, Diversify X tends to lower performance, a sign that sentence diversity may not benefit NER tasks, relative to entity diversity. Further manual inspection finds that data generated from Diversify X variants have lower-quality entity annotations (Appendix A.5.3), which may be the root cause of low performance. In particular, such annotations are more frequently incorrect and the wrong annotations encompass more disparate failure types. We hypothesize that explicit semantic requirements in Diversify X result in less-frequent sentences and smoother token distributions, thus creating room for additional LLM annotation errors. Combining sentence and entity diversity (Diversify X+Y) yields intermediate gains, likely still affected by the shortcomings of Diversify X.

Diversity have diminishing gains. Comparing the *vanilla* and *latent* variants of Diversify Y, while the performances in general domain are mixed, vanilla Diversify Y wins in specific domains. We hypothesize that specific-domain entity classes are already narrowly focused (e.g. Director in movie queries vs. Person in news) such that the *vanilla* setting generates sufficiently diverse datasets. The additional "topic" attribute value for the latent setting may only marginally increase diversity and at times introduce noise (e.g. generate *Genre* entities for movie queries about *Cast and crew*; Table 35; Appendix F.4.5).

4.3.2 Self-Correction

LLM self-correction leads to additional 1-2% F1 improvements for most cases. Manual inspection on LLM self-correction responses further show that most corrections are valid (i.e., turns originally wrong entity annotations into correct ones), indicating LLMs can learn and generalize from annotation feedback. LLM self-correction mostly preserves the relative trend among diversity variants as opposed to closing the gap, underscoring the critical role of sample diversity and the initial quality of annotations. Precision scores generally lag behind recall, suggesting a propensity for over-identification of named entities by LLMs, leading to more false positives (Wang et al., 2023a). Further, self-correction tends to enhance precision more than recall, correlating with the distribution of LLM correction kinds (Appendix A.5.3).

Below, we discuss and show examples for two most common classes of LLMs' systematic biases that yield false positives, which are the focus of annotation instructions in self-correction prompts. See Appendix A.6 for more error classes and representative samples.

LLMs are overly confident in what is a named entity. LLMs tend to falsely annotate generic, non-named references and categorical references as named entities.

Span: `The` **`CEO`** `of Apple Inc. announces a new product launch event.`
Entity Type: person
Span: **`global organization`** `announces plans for climate change summit in Reykjavik.`
Entity Type: organization

LLMs have overly broad definitions of entity classes. We found that LLMs may associate named entities of irrelevant types into one of the entity classes in prompt, such as counting events and awards as organization entities.

Span: `Zhang Yimou, a renowned` **`Chinese`** `film director, has won numerous awards...`
Entity Type: Location (Correct type: Demonym)
Span: `Show me a trailer for the movie that has a song performed by` **`Elton John`**`.`
Entity Type: Song (Correct type: Artist)

4.3.3 Specific Domain

In specific domains like Movies and Restaurants, data generation shows reduced effectiveness compared to general domains (News, Wikipedia), with most ProgGen variants lagging in F1 scores behind ICL. This shows that distilling LLMs' knowledge via data generation is especially challenging for more nuanced domains, with presumably smaller proportion of pre-training corpus on the given domain and respective entity classes. Notably, recall values are lower for specific domains, suggesting that the generated dataset might not fully capture the requisite domain-specific knowledge. However, gains for Diversify Entity variants roughly double for specific domains compared to general domains, effectively narrowing the performance gap with ICL in these challenging areas.

We further note that, in specific-domain datasets, certain entity classes are especially hard to define and separate due to ambiguity, yielding more wrong LLM annotations (detailed in Appendices A.5.3 and A.7). Manual inspection and partial F1 scores reveal that while LLMs correctly identify many entities, inaccuracies primarily arise from imprecise span boundaries and misclassified entity types (Appendices A.1 and A.6).

Method	Dataset Difference	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Rest.
Supervised Learning	—	90.3	88.0	84.8	76.5
LLM Annot.	Annot. quality	75.9 -14.4	75.0 -13.0	72.6 -12.2	64.0 -12.5
LLM Reph. & Annot.	Syntax	75.8 -0.1	77.4 +2.4	71.7 -0.9	63.0 -1.0
LLM Generated	Semantics	75.6 -0.2	76.2 -0.8	73.0 +1.3	61.5 -1.5

Table 6: Sentence distribution and LLM annotation accuracy ablation. Dataset Difference and Performance Drop values are w.r.t. the row above.

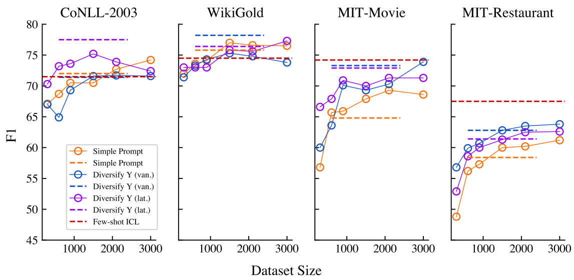


Figure 3: Sample Diversity Scaling plot. Corresponding F1 from the main results (Table 4) are shown in dashed lines.

See Appendix B for additional experiments.

5 Discussion

Scaling diversity alone plateaus Using generated dataset size as a rough proxy of sample diversity, we plot its relationship with downstream model performance in Figure 3. We compare Simple Prompt with the top-performing Diversify Y variants where we re-generate 3K valid samples. We take 6 subsets of sample sizes [300, 600, 900, 1.5K, 2.1K, 3K] where we enforce the smaller subsets are contained in the larger subsets. We then train NER models as in §4.2. We observe that substantially increasing dataset size doesn't lead to much performance gains across the board, confirming our speculation that sample diversity alone has diminishing returns (§4.3.1).

Entity annotation accuracy is performance bottleneck We conduct an ablation experiment to evaluate the impact of sentence distribution and LLM annotation accuracy on model performance, using modified datasets described below:

- *Supervised Learning*. A 90/10 train/dev split on 1.5K dataset samples as an upperbound[7].
- *LLM Annotated*. A 1.5K-sample subset of the training set containing the 1,350 training samples above, with entity annotations from ChatGPT.

[7] For WikiGold, we use the entire training dataset with just 1.1K samples.

- *LLM Rephrased & Annotated*. Sentences from the samples above, first rephrased and then annotated by ChatGPT.
- *LLM Generated*. The best-performing diversity setup using 1.5K LLMs-generated data for reference (from Table 4).

We then train NER models as in §4.2. For the *Supervised Learning* setup, we use the model with best dev loss. The results are shown in Table 6. We note that differences in sentence syntactic and semantic distributions hardly influence downstream performance, where as the largest drop in performance is attributed to the lower-quality of LLMs-generated annotations compared to the original. On the other hand, top-performing data-generation approaches achieve similar performances, indicating that with sufficient diversity, LLM-generated sentences matches a domain-relevant unlabeled corpus. Thus, annotating accurately w.r.t entity class definitions becomes the performance bottleneck.

Step-wise sample generation Considering the limited instruction following capabilities of GPT-3.5 (Ouyang et al., 2022) may influence performance, we explored generating NER samples step-wise (from sentences, entity spans, to entity types) with manually-written Chain-of-Thought (Wei et al., 2022) demos for entity annotation. We found no noticeable performance gains, despite the higher token querying costs due to re-prompting partially generated samples. See Appendix B.2 for additional discussion.

6 Conclusion

We present ProgGen, an innovative framework that leverages LLMs to generate diverse and accurate task-specific datasets via requirement sampling and self-reflection for low-resource NER. Experimental results on 4 datasets encompassing various domains demonstrate the advantages of our data generation approach compared to LLM few-shot in-context learning both in downstream model performance and cost-efficiency. Our research highlights the critical role of entity diversity in improving dataset quality and demonstrates the effectiveness of incorporating LLM self-correction alongside manual feedback to enhance annotation accuracy. Further analysis on the generated datasets reveals persistent biases in LLM entity classification and that annotating accurately is a major bottleneck to close the gap with traditional supervised learning methods.

7 Limitation & Future Work

More Comprehensive Experiments Our investigation focused on widely recognized NER benchmarks. Although we did not find signs of LLMs regurgitating training samples[8] (Carlini, 2020), experiments on less-widely-used datasets such as datasets in the medical domain or datasets released after LLMs' pretraining cutoff times would yield more reliable findings, as such datasets are less likely to be contaminated. Additionally, our work can be viewed as data augmentation which may particularly benefit lower-resource domains.

Furthermore, our study was constrained to utilizing ChatGPT3.5, primarily due to budgetary limitations. Expanding future research to a broader range of LLMs could provide more insights, as more advanced LLMs may follow instructions and annotate entities better.

Enhancing Automation Selecting representative demos for LLM Self-Correction can get labor-intensive since it involves examining large number of entity annotations for each generated dataset. It's worth assessing annotation instructions and demos shared for a given task domain. Advancing automatic feedback generation leveraging stronger LLMs such as GPT-4 can help, namely in (1) crafting annotation instructions and (2) evaluating annotations from weaker LLMs.

Further, the hyperparameter choices for data generation in this work is setup ad hoc whereas they may influence downstream performance noticeably. Future work can explore how to optimize these hyperparameters under the few-shot setup.

Better Annotations We found that LLMs may generate different/conflicting annotations on the same sentence. Such uncertainty signals can be leveraged to identify challenging instances and improve entity annotations. See Appendix C for additional discussions.

Ethics Statement

We acknowledge the potential biases of texts generated by language models which may not be factual and appropriately represent the included entities (including persons, locations, organizations, movies, restaurants) in the generated datasets. In our studies, we did not find noticeable cases of

[8]For example, in CoNLL-2003, we found that LLMs-generated news articles that are more recent, as opposed to news around 2003.

toxic language in the generated data as OpenAI API have content moderation policies in place. Nonetheless, harmful synthetic sentences may be present in the processed datasets.

Our research utilized publicly accessible NER benchmarks and was designed to be computationally efficient, with each training session on a generated dataset requiring no more than 10 minutes on a NVIDIA GeForce RTX 2080 Ti GPU.

Acknowledgements

This work was supported in part by NSF CAREER IIS-2144338, ONR MURI N00014-17-1-2656, and research grant from ADP. We appreciate the compute resources provided by Chao Zhang's group. We also thank Junmo Kang, Alan Ritter and the anonymous reviewers for their insightful feedback on the project.

References

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *ArXiv preprint*.

Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. 2009. Named entity recognition in Wikipedia. In *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources (People's Web)*, pages 10–18.

Nicholas Carlini. 2020. Privacy considerations in large language models. `https://blog.research.google/2020/12/privacy-considerations-in-large.html`.

Yew Ken Chia, Lidong Bing, Soujanya Poria, and Luo Si. 2022. RelationPrompt: Leveraging prompts to generate synthetic data for zero-shot relation triplet extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 45–57, Dublin, Ireland. Association for Computational Linguistics.

John Chung, Ece Kamar, and Saleema Amershi. 2023. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 575–593, Toronto, Canada. Association for Computational Linguistics.

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

Ruohao Guo and Dan Roth. 2021. Constrained labeled data generation for low-resource named entity recognition. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4519–4533, Online. Association for Computational Linguistics.

Himanshu Gupta, Kevin Scaria, Ujjwala Anantheswaran, Shreyas Verma, Mihir Parmar, Saurabh Arjun Sawant, Swaroop Mishra, and Chitta Baral. 2023. TarGEN: Targeted data generation with large language models. *ArXiv preprint*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Nicolas Hiebel, Olivier Ferret, Karen Fort, and Aurélie Névéol. 2023. Can synthetic text help clinical named entity recognition? a study of electronic health records in French. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2320–2338, Dubrovnik, Croatia. Association for Computational Linguistics.

Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. 2021. Few-shot named entity recognition: An empirical baseline study. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10408–10423, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting asymmetry for synthetic training data generation: SynthIE and the case of information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1555–1574.

Hyunjae Kim, Jaehyo Yoo, Seunghyun Yoon, Jinhyuk Lee, and Jaewoo Kang. 2022. Simple questions generate named entity recognition datasets. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6220–6236.

Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023a. Evaluating chatgpt's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness. *ArXiv preprint*.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023b. Camel: Communicative agents for "mind" exploration of large language model society.

Jiacheng Li, Haibo Ding, Jingbo Shang, Julian McAuley, and Zhe Feng. 2021. Weakly supervised named entity tagging with learnable logical rules. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4568–4581.

Yinghao Li, Rampi Ramprasad, and Chao Zhang. 2024. A simple but effective approach to improve structured language model output for information extraction.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023c. Synthetic data generation with large language models for text classification: Potential and limitations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10443–10461.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022a. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jian Liu, Yufeng Chen, and Jinan Xu. 2022b. Low-resource ner by data augmentation with prompting. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4252–4258. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Jingjing Liu, Panupong Pasupat, Scott Cyphers, and Jim Glass. 2013. Asgard: A portable architecture for multilingual dialogue systems. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8386–8390.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Ruotian Ma, Xin Zhou, Tao Gui, Yiding Tan, Linyang Li, Qi Zhang, and Xuanjing Huang. 2022. Template-free prompt tuning for few-shot NER. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5721–5732.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Chris Manning. 2006. Doing named entity recognition? don't optimize for f1. `https://nlpers.blogspot.com/2006/08/doing-named-entity-recognition-dont.html`. Blog post.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. 2022. Generating training data with language models: Towards zero-shot language understanding. In *Advances in Neural Information Processing Systems*.

Yu Meng, Martin Michalski, Jiaxin Huang, Yu Zhang, Tarek Abdelzaher, and Jiawei Han. 2023. Tuning language models as training data generators for augmentation-enhanced few-shot learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 24457–24477. PMLR.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Xuan Wang, Yu Zhang, Heng Ji, and Jiawei Han. 2021. Distantly-supervised named entity recognition with noise-robust learning and language model augmented self-training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10367–10378, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll 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 F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, pages 27730–27744.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. 2023. Does synthetic data generation of llms help clinical text mining? *ArXiv preprint*.

Megh Thakkar, Tolga Bolukbasi, Sriram Ganapathy, Shikhar Vashishth, Sarath Chandar, and Partha Talukdar. 2023. Self-influence guided data reweighting for language model pre-training. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2033–2045, Singapore. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *ArXiv preprint*.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023a. GPT-NER: Named entity recognition via large language models.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Yaqing Wang, Haoda Chu, Chao Zhang, and Jing Gao. 2021. Learning from language description: Low-shot named entity recognition via decomposed framework. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1618–1630, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023c. Self-instruct: Aligning language models with self-generated instructions.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

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

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical study of zero-shot NER with chatgpt. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 7935–7956. Association for Computational Linguistics.

Benfeng Xu, Quan Wang, Yajuan Lyu, Dai Dai, Yongdong Zhang, and Zhendong Mao. 2023a. S2ynRE: Two-stage self-training with synthetic data for low-resource relation extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8186–8207, Toronto, Canada. Association for Computational Linguistics.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023b. Baize: An open-source chat model with parameter-efficient tuning on self-chat data.

Ran Xu, Hejie Cui, Yue Yu, Xuan Kan, Wenqi Shi, Yuchen Zhuang, Wei Jin, Joyce Ho, and Carl Yang. 2023c. Knowledge-infused prompting: Assessing and advancing clinical text data generation with large language models. *ArXiv preprint*.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022a. ZeroGen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2022b. ProGen: Progressive zero-shot dataset generation via in-context feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3671–3683, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Junjie Ye, Nuo Xu, Yikun Wang, Jie Zhou, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. Llm-da: Data augmentation via large language models for few-shot named entity recognition.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2023a. Large language model as attributed training data generator: A tale of diversity and bias. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yue Yu, Yuchen Zhuang, Rongzhi Zhang, Yu Meng, Jiaming Shen, and Chao Zhang. 2023b. ReGen: Zero-shot text classification via training data generation with progressive dense retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11782–11805, Toronto, Canada. Association for Computational Linguistics.

Rongzhi Zhang, Rebecca West, Xiquan Cui, and Chao Zhang. 2022a. Adaptive multi-view rule discovery for weakly-supervised compatible products prediction. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4521–4529.

Rongzhi Zhang, Yue Yu, Pranav Shetty, Le Song, and Chao Zhang. 2022b. Prboost: Prompt-based rule discovery and boosting for interactive weakly-supervised learning. *arXiv preprint arXiv:2203.09735*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.

Ran Zhou, Xin Li, Ruidan He, Lidong Bing, Erik Cambria, Luo Si, and Chunyan Miao. 2022. MELM: Data augmentation with masked entity language modeling for low-resource NER. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2251–2262, Dublin, Ireland. Association for Computational Linguistics.

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

A Additional Analysis

A.1 Partial F1 Evaluation

We note that f1 only reward exact boundary matches of identified named entities (Manning, 2006) whereas partial matches may still benefit users of NER systems. As such, we also report the partial f1 scores following prior work (Zhou et al., 2024). We use the setup as described in Segura-Bedmar et al. (2013), where predicted named entities that overlaps with ground truth named entities of the same type are counted as 0.5 true positive[9]. We report partial F1 scores for the main experiments (§4.3) in Table 7. We observe that F1 and partial F1 scores share similar trends where we can make similar conclusions as in §4.3. We further note that (1) the gap between exact F1 and partial F1 scores are greater for specific domains (~10% vs ~3% for general domains), and thus (2) partial F1 closes the performance gap between general-domain- and specific-domain-datasets, indicating that annotating the precise entity span boundaries is more challenging in specific domains, corroborating with manual inspection (Appendix A.6).

A.2 Full Supervised Learning

As reference, we show standard supervised learning results using the same training pipeline as in §4.2 and our de-duplicated version of the original NER datasets (Appendix D.1) in Table 8. We report F1 on the model with best dev loss.

A.3 OpenAI API Stability

We note that data generation via OpenAI's Chat Completion API may yield unstable results, with minor data generation pipeline edits. Unfortunately, OpenAI API did not support generation output seeding until Nov. 2023, where even seeding yield different results sometimes. Earlier NER data generation results yield slightly different F1 scores compared to that we report in §4.3, as shown in Table 9. We attribute the discrepancy to OpenAI server changes (`system_fingerprint`)[10]. We can observe similar trends except that in earlier experiments, Diversify X works better more frequently. The final conclusion remains the same: (1) diversity improves performance, where entity diversity contributes the most, and (2) the *latent* variant of diverse entity yield most robust gains across datasets.

To further assess whether unstable OpenAI completions influence our findings, we re-run Simple Prompt 3 times with different random seeds (42, 43, 44) for prompt construction, OpenAI completion and downstream model training. We note that GPT3.5 frequently fail to generate $L = 50$ samples as instructed in prompt (may stop generation around 20 samples; see Appendix D.2.3), Thus, for the ease of seeding, we generated NER samples with $L \in [3, 10]$. We report the average F1 and standard deviation in Table 10. We observe that different L's influence dataset-specific performances noticeably with fluctuations, while for both L values, the average F1 scores stays around the same w.r.t. the results we report (Tables 4 and 9). This confirms our findings that that ProgGen leads to noticeable gains over Simple Prompt.

A.4 API Querying Cost

In this section, we show the token querying cost[11] for each step of data generation for the main experiments (§4.3) in Table 11. We make the following observations: (1) Simple Prompt costs 4X-5X less than ProgGen, since it contains only NER sample generation with significantly less number of prompt tokens (Appendix D.2.3). (2) The cost for Diversify X are primarily attributed to the high price of querying GPT4. (3) For a given dataset, costs for NER sample generation and entity correction are roughly the same across all diversity variants, since the only differences are the length of the diversity requirements and number of generated entity annotations, respectively.

A.5 LLM Self-Correction

A.5.1 Prompt Quality

We found that LLMs are especially biased in certain cases. For example, LLMs frequently consider demonyms (e.g. Chinese, American) as location entities. We found that more precise categorization of wrongly-annotated terms in the annotation instruction leads to more valid LLM self-corrections (e.g. "Demonyms" vs "Adjectives", "Ambiguous identifiers" vs "General references"). Instructing LLMs that certain terms are *not* named entities seems less effective than suggesting such terms are named entities of "other" type.

[9] We used the implementation from the `nervaluate` package available at `https://github.com/davidsbatista/NER-Evaluation`

[10] `https://cookbook.openai.com/examples/reproducible_outputs_with_the_seed_parameter`

[11] We used GPT3.5 input \$0.001, output \$0.002 and GPT4 input \$0.03, output \$0.06 for 1K tokens (cost at the time of experiments).

	Method	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Restaurant	Average
	Simple Prompt	72.0 / 75.6	75.8 / 77.9	64.8 / 76.7	58.4 / 73.5	67.8 / 75.9
ProgGen	Diversify X	69.2 / 72.5	75.5 / 80.2	65.0 / 73.9	54.7 / 72.3	66.1 / 74.7
	+ Self-Correction	71.9 / 74.6	77.3 / 80.3	63.5 / 73.5	56.1 / 73.1	67.2 / 75.4
	Diversify Y (vanilla)	71.2 / 73.7	76.2 / 81.4	73.0 / 80.9	61.5 / 78.2	70.5 / 78.6
	+ Self-Correction	71.4 / 74.3	**78.2 / 82.1**	**73.3 / 81.2**	**62.8 / 78.8**	71.4 / 79.1
	Diversify Y (latent)	75.6 / 78.1	75.7 / 79.2	71.8 / 80.0	60.9 / 78.2	71.0 / 78.9
	+ Self-Correction	**77.5 / 79.7**	76.4 / 80.1	72.9 / **81.2**	61.4 / 78.0	**72.1 / 79.8**
	Diversify X + Y	73.7 / 75.4	76.1 / 80.2	70.6 / 79.9	59.5 / 76.5	70.0 / 78.0
	+ Self-Correction	74.3 / 76.3	74.6 / 78.7	72.2 / 79.9	59.8 / 76.9	70.2 / 78.0
	Few-shot ICL	71.5 / 73.8	74.5 / 79.2	74.2 / 84.9	67.0 / 81.6	71.8 / 79.9

Table 7: ProgGen Performance Comparison. We show F1 and partial F1 score separated by slash (F1 / partial F1).

CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Restaurant	Average
94.2	88.0	87.8	79.9	88.6

Table 8: Supervised Learning results (F1).

Method	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Rest.	Avg.
Simple Prompt	69.5	72.2	67.0	56.1	66.2
Diversify X	73.2 +3.7	76.4 +4.2	70.2 +3.2	55.9 -0.2	68.9 +2.7
Diversify Y (vanilla)	72.1 +2.6	76.1 +3.9	**71.6** +4.6	**61.5** +5.4	70.3 +4.1
Diversify Y (latent)	73.4 +3.9	**77.9** +5.7	70.5 +3.5	60.6 +4.5	**70.6** +4.4
Diversify X + Y	**76.4** +6.9	72.9 +0.7	66.2 -0.8	58.7 +2.6	68.6 +2.4

Table 9: Early Data Generation Performance Comparison (F1).

A.5.2 Performance Drops

In the main results (Tables 4 and 5), we observe that less frequently, self-corrections leads to performance drops or insignificant performance differences. We hypothesize the reason is unstable training, as a result of 3 scenarios: (1) Since average log-probability is imperfect in identifying wrong annotations, *valid LLM corrections may not significantly influence downstream model learning* if the majority of same-class annotation errors are in the remaining 80% un-selected annotations, and thus not corrected (for example, cursory manual inspection found only 6 out of 40 occurrences of `directed (Director)` selected); (2) *LLMs may (less frequently) falsely correct previously valid entity annotations*, such as introducing noise into the training data by misclassifying entities; (3) Certain training samples have a larger influence on model learning (Thakkar et al., 2023). Changes to the annotations of these samples can lead to significant shifts in results. Appendix B.3.3 can serve as an evidence for points (2) and (3).

L	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Restaurant	Average
10	**74.4** $\pm$ 0.5	**76.7** $\pm$ 1.0	61.9 $\pm$ 3.0	**58.1** $\pm$ 1.9	**67.8** $\pm$ 1.6
3	71.2 $\pm$ 0.8	74.6 $\pm$ 0.6	**66.7** $\pm$ 0.5	57.8 $\pm$ 0.9	67.6 $\pm$ 1.0

Table 10: Simple Prompt stability comparison (F1). L refers to the number of NER samples LLMs are instructed to generate in each prompt.

Method	Gen. Kind	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Rest.	Avg.
Simple Prompt	NER	0.10	0.16	0.13	0.13	0.13
Diversify X	Attr. Dim.	0.09	0.14	0.11	0.13	0.12
	Attr. Val.	0.00	0.00	0.00	0.00	0.00
	NER	0.24	0.33	0.36	0.31	0.31
	Correct.	0.09	0.11	0.14	0.24	0.15
	Total	0.42	0.48	0.61	0.68	0.57
Diversify Y (vanilla)	Entity	0.01	0.02	0.07	0.04	0.04
	NER	0.24	0.31	0.35	0.30	0.30
	Correct.	0.08	0.16	0.16	0,22	0.16
	Total	0.33	0.49	0.58	0.56	0.49
Diversify Y (latent)	Entity	0.18	0.23	0.27	0.08	0.19
	NER	0.24	0.32	0.35	0.30	0.30
	Correct.	0.09	0.15	0.17	0.22	0.16
	Total	0.51	0.70	0.58	0.56	0.49
Diversify X + Y	Divers.	0.27	0.37	0.38	0,21	0.31
	NER	0.27	0.35	0.39	0.33	0.34
	Correct.	0.08	0.13	0.15	0.26	0.16
	Total	0.62	0.85	0.92	0.80	0.80
Few-shot ICL	Pred.	0.39	0.49	0.61	0.48	0.49

Table 11: ProgGen API Token Querying Cost Comparison. *Gen. Kind* refers to the type of data generated and are ordered by the ProgGen steps performed, where *Attr. Dim.*, *Attr. Val.*, *NER* and *Correct.* refer to generating attribute dimensions, attribute values, the entity pool, NER samples and entity corrections respectively. For *Diversify X + Y*, we sum the cost of all diversity requirement generation in *Divers.* (values taken from *Diversify X* and *Diversify Y (latent)* rows).

We notice that occasionally, LLM Self-Correction trades improved precision with lower recalls, indicating LLMs may drop valid named entity annotations.

A.5.3 Correlation with Correction Statistics

We make a *loose* assumption that the probability of LLM corrections that are invalid is about the same across diversity variants and regardless of the class of annotation error.

Downstream Performance As such, we can correlate the distribution of LLM correction types (Appendix E.3) with downstream performance gains (Tables 4 and 5).

We note that by definition of F1 score for NER evaluation, (1) valid entity span corrections would improve true positives and reduce false positives, (2) valid entity type corrections would improve true positives, and (3) valid entity drops would reduce false positives. More true positives lead to improvements in precision and recall, whereas less false positives lead to higher recall.

We make the following observations: (1) For all datasets except MIT-Restaurant, we observe high ratio of entity drops relative to entity span and type corrections, indicating high occurrence of false positives (Tables 32 and 33), which can explain mostly higher precision gains from LLM self-correction. (2) For all diversity variants with explicit semantic requirements (*Diversify X* and *Diversify X+Y*), we observe noticeably higher ratios of annotations LLM corrected (Table 32), reflecting the low-quality annotations related to sentence diversity.

Annotation Difficulty With the same assumption, we can also get insight into which classes are more challenging for LLMs to annotate. In Table 33, we observe that the difficulty for annotating entities in specific domains is greater and more varied across classes (e.g. In MIT-Movie, LLMs corrected from 10+% to ~60% selected entities).

Specifically, based on the distribution of entity type corrections (Figure 5), we can probe into how well LLMs separate relevant entity classes. Notably, (1) in CoNLL-2003 and WikiGold, Location entities and Organization entities are more interleaved. (2) In MIT-Movie, Genre and Plot entities, and Viewers' Rating and Review entities, are more mixed up. (3) LLMs especially struggle with identifying the correct type in MIT-Restaurant, with Amenity, Cuisine and Dish entities mixed together and many Amenity and Hours entities falsely labeled.

A.6 Representative Annotation Errors

In this section, we select and show representative entity annotation errors for each dataset in Tables 12 and 13 to highlight systematic biases in LLMs. See the exact and illustrative annotation instructions that covers broader classes of LLM annotation errors repository[12].

We note that, for general domain datasets (CoNLL-2003 and WikiGold), the main problem in LLM annotations is false positives, including (1) annotating generic and non-name references (e.g. CEO as person entity) and (2) categorizing irrelevant named entities (e.g. mythological figures as person entities). For specific domain datasets (MIT-Movie and MIT-Restaurant), LLMs mainly suffer from (1) imprecise spans annotated (e.g. "action movie" as opposed to "action"), and (2) entity type confusion (e.g. Viewers' Rating vs Review). In MIT-Restaurant, we also observed that GPT-3.5 may drop many valid named entities (more frequently than other datasets).

A.7 Entity Class Ambiguities

We note that, espcially for datasets in the specific domain (MIT-Movie and MIT-Restaurant), certain entity classes are hard to define just given its label (e.g. Viewers' Rating vs Review). Without access to entity annotation schemas, there can be multiple valid interpretations leading to different "correct" annotations. Further, inferring the annotation schemas is challenging even for humans since even original samples in the test set are noisy. We show three significant (highly-occurring) classes of ambiguities encountered in our study below. See Appendix A.6 for representative LLM annotation errors that reflect these challenges. In particular, we define the entities with our judgement from reviewing samples in the dataset (mostly determined via majority vote on similar entity contexts) and show representative ambiguous samples in the test set.

A.7.1 Ambiguous Class Separation

MIT Movie: Viewers' Rating vs Review We separate Viewers' Rating entities and Review entities based on level of detail, where Viewers' Rating entities are shorter such as "good" and Review entities are more detailed such as "incredible visual effects". By this definition, a large fraction of Review entities in the test set should have been Viewers' Rating entities.

MIT Restaurant: Cuisine vs Hours It's unclear whether meal times such as "breakfast" and "lunch" are Cuisine entities or Hours entities, since (1) both types share similar query contexts for the same

[12] https://github.com/StefanHeng/ProgGen/reproduce/correction

Type	Span	Issue	Desc.
CoNLL-2003			
person	Renowned economist **Dr. <u>John Smith</u>** predicts a downturn in the stock market.	Span	Starting title
	Local **woman** raises funds for homeless shelter by knitting blankets.	NA	Generic reference
	CEO of <u>General Motors</u> predicts electric vehicles will account for 40% of company's sales by 2025.	NA/Type/ Span&Type	Non-name reference / Title / Contains named org. entity
location	**Japanese** prime minister announces economic reforms to boost the country's GDP.	Type	Denonym
	Hurricane Katrina devastates New Orleans, Louisiana.	Type	Natural disaster
	Celebrities flock to **<u>Paris</u> Fashion Week** for the latest trends.	Type/ Span	Event / Contains named loc. entity
	New study shows increase in pollution levels in **major cities**.	NA	Categorical reference
	New mural depicting local history to be commissioned in **downtown arts district**.	NA	Non-name reference
organization	**Chinese** President Xi Jinping meets with Indian Prime Minister Narendra Modi to discuss trade relations.	Type	Denonym
	WHO issues new guidelines for **COVID-19** prevention.	Type	Virus
	Apple announces new **iPhone** release date.	Type	Product
	New startup **company** in Edinburgh secures funding for expansion.	NA	Generic reference
	Women starts **non-profit organization** to provide free meals for the homeless in downtown Los Angeles.	NA	Categorical reference
WikiGold			
person	**Dr. James E. Webb** was instrumental in the development and success of the Apollo lunar landing program.	Span	Starting title
	President of the <u>United States</u> is the head of state and head of government of the United States of America.	NA/Type/ Span&Type	Non-name reference / Title / Contains named org. entity
	In Greek mythology, **Apollo** was the god of music, poetry, art, oracles, archery, plague, medicine, sun, light, and knowledge.	Type	Fictional figure
	The **Mona Lisa** is a famous painting created by Leonardo da Vinci.	Type	Work of art
	She is a well-known **entrepreneur** who has founded several successful companies in the tech industry.	Type	Non-name reference
location	The **African** elephant is the largest living terrestrial animal.	Type	Denonym
	According to the National Weather Service, **Hurricane Katrina** was the costliest natural disaster in the history of the United States.	Type	Natural disaster
	The new business park, located in the heart of **the city**, aims to attract both local and international companies looking to expand their presence in the region.	NA	Non-name reference
organization	Sylvia Earle, an **American** marine biologist, has dedicated her life to the exploration and conservation of the world's oceans.	Type	Denonym
	The director, Steven Spielberg, won an **Academy Award** for Best Director for his work on the film.	Type	Award
	The film premiered at the **<u>Cannes</u> Film Festival**.	Type/ Span&Type	Event / Contains named loc. entity
	The 2020 Summer Olympics was postponed to 2021 due to the **COVID-19** pandemic.	Type	Virus
	NASA's **Artemis program** aims to return humans to the Moon by 2024, with the goal of establishing a sustainable human presence on the lunar surface.	Type	Landmark project
	The **Harry Potter** series, produced by Warner Bros., has been a massive success worldwide.	NA	Work of art
	The Battle of Helm's Deep, a fictional battle in the novel **The Two Towers** by J.R.R. Tolkien, is one of the most iconic scenes in the book and its film adaptation.	Type	Literature
	The Battle of Gettysburg was a significant turning point in the **American Civil War**.	Type	Historical period
	The **band** released a music video for their latest single on YouTube, which quickly gained millions of views.	NA	Non-name reference

Table 12: Representative LLM annotation errors for CoNLL-2003 and WikiGold. In the *Span* and *Type* columns, we show the entity spans annotated by LLMs in **bold**, within the sentences as context, and the corresponding annotated types. In *Issue*, we label the wrong annotation as one of (1) wrong span (Span), (2) wrong type (Type), and (3) not a named entity (NA). Spans that can be considered as named entities outside the relevant classes in the respective dataset are also labeled as Type. Multiple valid entity interpretations are separated by slash. Candidate entity spans are <u>underlined</u>. We summarize the class of annotation error in the *Desc.* column.

entity span and (2) the occurrence counts of the same entity span can be close to each other (e.g. 8 vs 9). Thus, we found no better solution than just declaring all such terms as Hours entities (for slightly more annotations).

We show samples in Table 14.

A.7.2 Conflicting Annotation Guideline

We note that higher-order annotation guidelines are counter-intuitively conflicting in certain cases, where annotations need to follow opposite rules for different entity classes[13].

[13] One reason can be that the tasks for MIT-Movie and MIT-Restaurant are more precisely slot filling as opposed to named entity recognition, where the relevant information to extract may differ.

MIT Movie: Trailer vs *rest* We note that annotating the Trailer class is not precisely NER, where vast majority of the annotated spans simply indicate querying about trailer information (A). However, other entity types don't following the same rule, i.e. querying about other entity types (e.g. director information) are not highlighted (B).

MIT Restaurant: Rating vs Price Rating entities frequently include "rating" in the phrase, such as "4-star rating" as opposed to "4-star" (A), where as Price entities often omit trailing "price" terms (B).

We show samples in Table 15.

Type	Span	Issue	Desc.
MIT-Movie			
Title	Can I find tickets for the **new James Bond movie**?	Span/ Span&Type	Starting & trailing desc. Character
	What is the plot of the **newest movie** starring Owen Wilson?	NA	Generic reference
Viewers' Rating	Please show me a movie with a **high** viewers' rating.	Span	Incomplete phrase
	What's the **rating** for the latest Marvel movie?	NA	Generic reference
Year	When was "The Godfather" **released**?	NA	Time indicator
	Can you show me the trailer for the **latest** Harry Potter film?	NA	Temporal descriptor
Genre	Are there any **good action movies** playing today?	Span	Starting & trailing desc.
	What movie features a **Dance competition** and compelling plot twists	Type	Movie element (Plot)
Director	Who **directed** the highest-rated horror movie of the year?	NA	Director indicator
	I'm open to watching a movie directed by a **female filmmaker** that received critical acclaim. Do you have any recommendations?	NA	Non-name reference
	Can I see a list of theaters showing the **new Christopher Nolan film**?	Span	Starting & trailing desc.
MPAA Rating	I want to see a trailer for The Wolf of Wall Street, **rated R**	Span	Starting desc.
	Is John McClane in any **parental guidance** films that are worth watching?	Type	Rating interpretation
Plot	Which director has released a new film with a **thrilling plot** that I can get a sneak peek of?	NA	Plot desc.
	could you tell me the **plot** of the latest marvel film?	NA	Generic reference
Actor	Who are the **main actors** in the latest sci-fi movie directed by Christopher Nolan?	NA	Generic reference
Trailer	Can you play a **Movie clip** from Inception starring Marion Cotillard?	Span	Starting desc.
Song	Can you recommend a movie with **spectacular music** from 1971?	NA	Song attribute
	Show me a movie from 1999 with a **soundtrack** that includes dance music.	NA	Generic reference
	I want to watch a movie with a song by **Adele**. Show me the options.	Type	Artist
Review	What movie with a mind-bending plot has the **best viewers' rating**?	Type	Viewers' Rating
	I'm looking for a **superb** movie directed by Morpheus. Can you help me find one?	Type	Viewers' Rating
Character	Show me a movie with a **strong female lead** character.	NA	Categorical reference
MIT-Restaurant			
Restaurant Name	What are the hours of operation for the **steakhouse** downtown?	Type	Cuisine
Amenity	Recommend a family-friendly diner that serves **breakfast** all day.	Type	Hours
	Where is the closest cafe for a good **Cafe** Mocha?	Type	Cuisine
Cuisine	What time does the **Italian restaurant** on Main Street close tonight?	Span	Trailing desc.
	Is there a 24-hour restaurant nearby that offers a **breakfast** buffet?	Type	Hours
Dish	I'm craving some fast food, where can I find a good **burger joint**?	Type	Cuisine
	I want to try a new seafood **dish**, any suggestions?	NA	Generic reference
Hours	What are the **operating hours** for the BBQ food truck on 5th Avenue?	NA	Hours indicator
Location	I'm looking for a well-recommended seafood restaurant in this **area**.	Span	Whole descriptive phrase
Price	Which restaurant in this area serves Italian cuisine at a **reasonable price**?	Span	Trailing desc.
	What is the **price range** for italian restaurants in this area?	NA	Price indicator
Rating	I'm looking for a high-end restaurant with a **5-star** rating and a location in downtown Manhattan	Span	Whole descriptive phrase

Table 13: Representative LLM annotation errors for MIT-Movie and MIT-Restaurant.

A.7.3 Unseen Edge Cases

We note that LLMs-generated samples are more diverse (Appendix E.2) and LLMs may generate edge-case spans unlike any cases in the original entities annotated. For example, in WikiGold, LLMs may generate places in mythology as location entities where as the original dataset samples don't contain sentences about mythology. In such ambiguous cases, we use our judgement to determine the correct annotation. For example, (1) in WikiGold, we consider places in mythology such as "underworld" as Location entities; (2) In MIT-Movie, we consider popular movie studios such as "Marvel" and "Disney" as Genre entities, and we ignore label interpretations (e.g. "Parental Guidance" as opposed to "PG") as MPAA Rating entities.

B Additional Experiments

In this section, we discuss preliminary experiments that guide the ProgGen design choices.

B.1 Sample Format Search

Cursory experiments explored 4 different formats for representing NER samples (examples in Table 16) in two directions: (1) generating complete sentences (*natural language*), and (2) generating token by token (*token-wise*):

1. *Natural Pair*. Generate a sentence and the list of (entity name, entity type) pairs.
2. *Natural Inline*. Generate a sentence and then generate the sentence again with inline entity class annotations.
3. *Token List*. Generate a sentence, then generate

Correct?	Span	Type
MIT-Movie		
✓	what is the **best** viewer rated vampire film what are some **good** kids movies starring adrian pasdar list the romance films directed by james cameron rated **must see**	Viewers' Rating
✗	what was the **best** rated stanley kubrick film what are some **good** animated films are there any meg ryan romantic comedy movies that are considered **must see**	Review
MIT-Restaurant		
✓	is there a **breakfast** place that has valet parking find me a southwestern restaurant that serves **breakfast** and is located nearby can you help me get to a restaurant where i can get **lunch** for under 10	Hours
✗	id like to find a **breakfast** place i need to know of a place that serves **breakfast** beginning as early as 5 30 is there anywhere near here that is open 24 hours and serves **breakfast**	Cuisine
✗	can you help me find a high end restaurant where i can have **lunch**	Amenity

Table 14: Selected original NER samples that showcase ambiguous class separation in MIT-Movie and MIT-Restaurant. For each group (same expected entity type), we show correct (✓) annotations on top and incorrect annotations (✗) on bottom (w.r.t. our inferred entity definitions). All samples are original samples in the test sets with entity spans in query contexts highlighted in **bold** (*Span*) and the original type annotations (*Type*).

Span	Type (A)	Type (B)
MIT-Movie		
is there a **trailer** out for advengers yet	Trailer	*NA*
is billy wilder the director of inception	Director	*NA*
show me the latest **trailer** for the avengers	Trailer	*NA*
what is the mpaa rating for star wars episode 5	MPAA Rating	*NA*
what is the vuewers rating for heidi	Viewers' Rating	*NA*
MIT-Restaurant		
is there a restaurant that has a **5 star rating** for it that people like	Rating	Rating
find a clean place to eat that has **reasonable** prices	Price	Price
find me a thai restaurant with a **great rating**	Rating	Rating
any good place to get a pie at an **affordable** price	Price	Price

Table 15: Selected original samples that showcase conflicting annotation guidelines in MIT-Movie and MIT-Restaurant. In each group (similar context), we show original samples in the test sets with different high-level guidelines (A and B). We highlight originally annotated entity spans in **bold** and underline alternative entity spans under the opposing guideline (*Span*). In *Type*, we show the corresponding entity type under each guideline.

the tokens in the sentence, and finally the BIO tags corresponding to each token.

4. *Token Lines*. Generate (token, BIO tag) line by line, in CoNLL-2003 style.

We note that each sample formatting are equally expressive. Namely, LLMs have the option to annotate multi-occurring text spans with potentially different types.

We conduct experiments on the CoNLL-2003 dataset and select the top-performing sample format for the rest of our experiments in this study.

We generate around 1K NER samples[14][15] using a prompt template like Simple Prompt (Table 1). We use the first 5 samples in the CoNLL-2003 training set as demo examples. we observed that the *Token Lines* format had repetition issues and generated exact samples in the original CoNLL-2003 dataset. Thus, we did not investigate this sample format further. A BERT (Devlin et al., 2019) model was trained on other sample formats with learning rate $2e^{-5}$. Other training hyperparameters remain the same as the main experiments (§4.2 and Appendix D). We used the `bert-base-cased` checkpoint from HuggingFace. We compare the best test set F1 of the BERT model from epoch-wise evaluation (results in Table 16). The model trained on datasets generated via the *Natural Pair* format performs the best.

[14]We used the GPT3.5 `gpt-3.5-turbo-0613` checkpoint.

[15]Experiments were done between Sep. 12th - 17th, 2023.

B.2 Step-wise Generation

The instruction-following capability of ChatGPT3.5 may be limited. The complexity of simultaneously generating diverse samples and accurately annotating entities in a single prompt may present a considerable challenge. Consequently,

Sample Format	Example	F1
Natural language		
Pair	Sentence: "EU rejects German call to boycott British lamb." Named Entities: [EU (organization), German (miscellaneous), British (miscellaneous)]	**59.8**
Inline	Sentence: "EU rejects German call to boycott British lamb." Annotated Sentence: "**EU** (organization) rejects **German** (miscellaneous) call to boycott **British** (miscellaneous) lamb."	58.4
Token-wise		
List	Sentence: "EU rejects German call to boycott British lamb." Tokens: [EU, rejects, German, call, to, boycott, British, lamb, .] NER tags: [B-organization, O, B-miscellaneous, O, O, O, B-miscellaneous, O, O]	52.1
Lines	sentence: EU, B-organization rejects, O German, B-miscellaneous call, O to, O boycott, O British, B-miscellaneous lamb, O ., O	-

Table 16: NER Sample Format examples and downstream performance on CoNLL-2003.

Method	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Restaurant	Average
Simple Prompt	71.3	75.6	65.5	54.2	66.7
Diversify X	72.7 +1.4	76.8 +1.2	62.7 -2.8	57.2 +3.0	67.4 +0.7
Diversify Y (vanilla)	73.6 +2.3	**77.4** +1.8	**70.6** +5.1	**61.3** +7.1	**70.7** +4.0
Diversify Y (latent)	**74.9** +3.6	76.7 +1.1	65.9 +0.4	59.2 +5.0	69.2 +2.5
Diversify X + Y	**74.9** +3.6	76.7 +1.1	65.8 +0.3	57.9 +3.7	68.8 +2.1

Table 17: 2-Stage Data Generation Performance Comparison (F1).

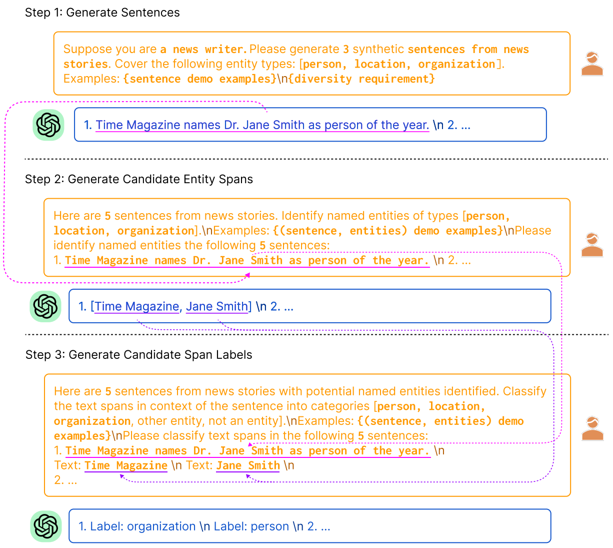
we investigate whether separate prompts for each aspect (sentence generation & sentence annotation) leads to improved instruction following and thus improved downstream performance. In this section, we explore multiple variants of step-wise sample generation, all starting from sentence generation.

B.2.1 2-Stage Generation

Our 2-Stage NER sample generation pipeline is illustrated in Figure 4a. We first prompt LLMs to generate diverse sentences. Then, the generated sentences are annotated by LLM. In the first step (sentence generation), we add explicit diversity requirements just like in default NER sample generation (§3.2; note the diversity requirements are for the sentences). Then, for all diversity setups, we use the same annotation prompt with temperature $t = 0$.

We run experiments with setup just like in §4.2.

(a) 2-Stage NER Sample Generation pipeline. We first prompt LLMs to generate sentences. The LLMs are prompted again to annotate the generated sentences.



(b) 3-Stage NER Sample Generation pipeline. We first prompt LLMs to generate sentences. Then, LLMs are prompted again to identify potential named entities in the generated sentences. Finally, LLMs are prompted to classify the type of each named entity identified in context of the generated sentences, with the option to drop the named entity.

Figure 4: Step-wise NER Sample Generation pipeline.

The results are shown in Table 17[16]. We observe that, compared to the vanilla (1-stage) data generation (Appendix A.3 and Table 9), 2-stage doesn't yield higher F1 scores, even at the expense of additional token querying cost (for prompting the generated sentences again), indicating simply re-annotating in an isolated prompt doesn't improve annotation accuracy.

Chain-of-Thought We study whether chain-of-thought (CoT) prompting (Wei et al., 2022) can help LLMs annotate better where we manually write annotation reasoning for each demo sample. We compare generating entity annotations (step 2) naively (*vanilla*) or with manually written chain-

[16]Experiments were done between Nov. 28th - Dec. 28th, 2023.

Instruction: Here are 5 sentences from news stories. Please analyze each sentence and identify all named entities occurred that belong to one of the following entity types: [person, location, organization]. Show your reasoning in bullet points. Please list such named entities with the corresponding entity types on the following line, in the order of occurrence. If no entity is found in the generated sentence, leave the brackets empty.
Examples: Here are some example reasonings and annotations for your reference. Please follow this format.\n Examples:\n 1. Sentence: "Saudi Arabia executes Pakistani man."\n\nReasoning: - "Saudi Arabia" is a general country name and hence a location. - "Pakistani man" refers to an individual, but it's not the name of a specific person, so it's not a person entity.\n Named Entities: [Saudi Arabia (location)]\n 2. Sentence: "BASEBALL - DODGERS WIN FIFTH STRAIGHT."\n\nReasoning: - "WIN FIFTH STRAIGHT" likely describes the record of a baseball team, so "DODGERS" should be the name of a sports team. Thus, it's an organization. - "DODGERS" is a known baseball team name, which qualifies it as an organization entity.\n Named Entities: [DODGERS (organization)]\n 3. Sentence: "With 2.5 percent gross domestic product growth expected for 1996, new job growth should slowly lower the unemployment rate over the rest of the year."\n\nReasoning: - "2.5 percent" is a percentage. "1996" is a year. The sentence does not mention specific persons, geographical locations, or named organizations.\n Named Entities: []\n 4. Sentence: "A.de Silva not out 49"\n\nReasoning: - "A.de Silva" appears to be an individual's name, hence it's categorized as a person. - "not out 49" seems to describe a player's score/status in a sports game. Thus, "A.de Silva" is the name of a player, thus a person entity.\n Named Entities: [A.de Silva (person)]\n\n ---\n
Input: Please analyze and annotate the following 5 sentences:\n 1. Sentence: "Remote village in the Himalayas cut off from aid after heavy snowfall blocks mountain passes."\n 2. Sentence: "The CEO of Amazon, Jeff Bezos, has announced his resignation from the company."\n 3. Sentence: "The United Nations has launched a new initiative to promote gender equality and empower women and girls around the world."\n 4. Sentence: "Russian President Vladimir Putin visits China to discuss economic cooperation."\n 5. Sentence: "Megan Markle and Prince Harry speak out against racism and discrimination in the media."\n ---\n Show your reasoning. Let's think step by step.

Table 18: 2-Stage Data Generation step 2: Generate Entity Annotations with CoT prompt example for CoNLL-2003.

of-thought reasoning (*CoT*). We experiment on the CoNLL-2003 dataset with Simple Prompt[17]. An example entity annotation prompt is shown in Table 18. We found that CoT results in worse downstream performance (69.6% vs 72.0% F1). Manual examination reveals that many LLM-generated reasoning chains are not factual, as if copying the formatting from the demo samples.

B.2.2 3-Stage Generation

We also experimented with 3-stage generation (pipeline illustrated in Figure 4b), further splitting the entity-generation step into candidate entity span generation (step 2) and entity span classification (step 3). In the last step, LLMs can also optionally drop the entity candidate entirely to address the common false-positive issues (Detailed discussion in Appendix A.5.3). Intuitively, since step 3 cannot add entity spans, LLMs are instructed to generate more candidate entities in step 2 where we also add non-entities as demos. In the last step, we group all candidate spans for the same sentence together

[17]Experiments were done between Nov. 19th - 21st, 2023.

Instruction: Here are 5 sentences from news stories. For each sentence, please identify all named entities occurred using the format [named entity 1; ...]. Especially identify named entities that belong to one of the following entity types: [person, location, organization]. Try to include all possible named entity spans at the expense of potential overlaps. If no entity is found in the generated sentence, leave the brackets empty.
Examples: Here are some example entity identifications for your reference. Please follow this format.\n Examples:\n 1. Sentence: "With 2.5 percent gross domestic product growth expected for 1996, new job growth should slowly lower the unemployment rate over the rest of the year." Named Entities: []\n 2. Sentence: "BASEBALL - DODGERS WIN FIFTH STRAIGHT." Named Entities: [DODGERS]\n 3. Sentence: "Saudi Arabia executes Pakistani man." Named Entities: [Saudi Arabia; Pakistani; Pakistani man]\n 4. Sentence: "A.de Silva not out 49" Named Entities: [A.de Silva]\n\n ---
Input: Please identify named entities in the following 5 sentences: 1. Sentence: "Mary Barra, the CEO of General Motors, announced plans to open a new production facility in Tokyo."\n 2. Sentence: "Munich-born CEO of Volkswagen announces plans to expand electric vehicle production in Dubai, United Arab Emirates."\n 3. Sentence: "Renowned scientist, Dr. Jane Smith, awarded prestigious Nobel Prize for groundbreaking research in medical science."\n 4. Sentence: "The United Nations reports a surge in humanitarian aid to Afghanistan as winter approaches."\n 5. Sentence: "Renowned climber John Smith reaches the summit of Mount Everest for the fifth time."\n

Table 19: 3-Stage Data Generation step 2: Generate Candidate Entity Spans prompt example for CoNLL-2003.

to reduce number of prompt tokens and thus token querying costs (c.f. classifying each distinct (sentence, entity) pair).

We experimented with CoT prompting for both step 2 and 3 on CoNLL-2003 and MIT-Movie to improve annotation accuracy[18]. We show example prompts in in Tables 19, 20, 21 and 22. For MIT-Movie, to provide additional annotation instructions, we further added a manually crafted NER sample to showcase representative annotation errors. Performance comparisons are shown in Table 23. We found that 3-stage generation shows gains for CoNLL-2003, but makes no difference in F1 score for MIT-Movie. Manual inspection shows that the main problem for MIT-Movie is incorrect span boundaries as a result of step 2, where as step 3 can only correct type errors. Thus, we proposed LLM self-correction (§3.3) with more explicit feedback and correction mechanisms.

B.3 Entity Annotation Uncertainty Ranking

As discussed in §3.3 and §3.3.1, we compare two scoring functions to select entity annotations for LLM Self-Correction (Appendices B.3.1 & B.3.2).

B.3.1 LLM Generation Log-Prob

We average the log-probabilities for all tokens w.r.t the entity annotation (i.e. log-probs for `span (entity type)`), denoted *log-prob*. As such, a lower average log-prob means more uncertainty.

[18]Experiments were done between Dec. 27th - Jan. 15th, 2023.

Instruction: Here are some sentences from news stories with potential named entities identified. Your task is to classify whether the spans of text are named entities of the following entity types:
[person, location, organization].
In particular, for each sentence with a span of text enclosed in double curly braces, please analyze and classify the enclosed span of text in context of the sentence into one of the following categories:
[named person entity, named location entity, named organization entity, named entity of other type, not a named entity]

Examples: Here are some example span analyses and classifications for your reference. Please follow this format.
Examples:
1. Sentence: "A.de Silva not out 49"
Text Span: "A.de Silva"\nLabel: "de Silva" is a first name initial followed by a common last name, so "A.de Silva" should be an individual's name. Additionally, "not out 49" describes a player's score in a sports game. Based on context, "A.de Silva" is the name of a player. A player is a person. A specific person's name is a person entity. The span is a named person entity.\n
2. Sentence: "With 2.5 percent gross domestic product growth expected for 1996, new job growth should slowly lower the unemployment rate over the rest of the year."
Text Span: "2.5 percent"\nLabel: "2.5 percent" is a percentage and a named entity. However, a percentage is not a person, location, or organization. The span is a named entity of other type.
Text Span: "1996"\nLabel: "1996" is a year and a named entity. However, a year is not a person, location, or organization. The span is a named entity of other type.\n
3. Sentence: "BASEBALL - DODGERS WIN FIFTH STRAIGHT."
Text Span: "DODGERS"\nLabel: "WIN FIFTH STRAIGHT" describes the record of a baseball team, so "DODGERS" refers to a sports team by its name. Additionally, "DODGERS" is a known baseball team name. A sports team is an organization. The span is a named organization entity.\n
4. Sentence: "Saudi Arabia executes Pakistani man."
Text Span: "Saudi Arabia"\nLabel: "Saudi Arabia" is the name of a specific country. The name of a country is a location entity. The span is a named location entity.
Text Span: "Pakistani"\nLabel: "Pakistani" is a adjective describing where the "man" is from. The span "Pakistani" in itself is a nationality, not a country name. A adjective describing a nationality is not a named entity. The span is not a named entity.
Text Span: "Pakistani man"\nLabel: "Pakistani man" refers to someone but does not provide the individual's name. A reference to a person not by the name is not a named entity. The span is not a named entity.\n
---

Input: Please analyze and classify spans of text in the following 5 sentences:
1. Sentence: "Organization for Economic Cooperation and Development predicts global economic growth to slow down in 2021."
Text Span: "Organization for Economic Cooperation and Development"\nText Span: "2021"\n
2. Sentence: "Sahara Desert experiences record high temperatures, posing challenges for local communities and wildlife."
Text Span: "Sahara Desert"\n
3. Sentence: "Johnny Depp to star in new film directed by Tim Burton."
Text Span: "Johnny Depp"\nText Span: "Tim Burton"\n
4. Sentence: "Stanford University appoints former UK prime minister as guest lecturer for political science course."
Text Span: "Stanford University"\nText Span: "UK"\nText Span: "prime minister"\n
5. Sentence: "Department of Justice announces new initiative to combat cybercrime in partnership with global organization."
Text Span: "Department of Justice"\nText Span: "global organization"

Table 20: 3-Stage Data Generation step 3: Generate Candidate Span Labels with CoT prompt example for CoNLL-2003.

B.3.2 Downstream BERT Loss

We first train a BERT-class model on the generated dataset and compare its prediction signals with the LLM annotations (i.e. the dataset the model was trained on), where lower prediction discrepancy means more uncertainty (denoted *loss*). Intuitively, we can leverage feedback from the trained downstream model since it learns from a larger pool of NER samples and learns to resolve noises in the original LLM annotations in the training process. We run inference with the downstream model on the generated sentences to get token-wise cross entropy (CE) loss w.r.t the original LLM annotations[19], where higher loss means higher discrepancy between BERT and LLMs' annotations. To get the ranking score for each LLM entity annotation, we average the negated loss values for the subset of tokens in the sentence that correspond to the entity span. We further include one more token on both sides of the entity span if possible to better capture discrepancy in span boundaries. For example, given a generated NER sample with tokens [`Bob`, `is`, `born`, `in`, `Athens`, `.`] and BIO tags [`B-person`, `O`, `O`, `O`, `B-location`, `O`], to get the uncertainty score for "Athens", we average the negated loss for the tokens [`in`, `Athens`, `.`].

We note that conceptually, *loss* ranking can be considered as a log-probability and hence comparable with *log-prob* on the same scale. As shown in equation 1, the negated CE loss for token position i with entity class label y is effectively the log of a probability in range $[0, 1]$:

$$\begin{aligned} -\text{CE Loss}(i, y) &= \log \frac{\exp(\text{logits}(i))}{\text{sum}(\exp(\text{logits}(i)))} \\ &= \log \text{softmax}(\text{logits}(i))[y] \\ &= \log P(y) \end{aligned} \tag{1}$$

[19]In terms of implementation, we simply use CrossEntropyLoss from PyTorch treating the LLM annotations as "ground truth".

Instruction: Here are 5 spoken queries to a dialogue system about movies. Please analyze each query and identify and extract informative keywords in the queries using the format [keyword 1; ...].
Especially identify informative keyword spans that belong to one of the following entity types:
[Title, Viewers' Rating, Year, Genre, Director, MPAA Rating, Plot, Actor, Trailer, Song, Review, Character].
Try to include all possible keyword spans at the expense of potential overlaps.
Genre keywords should not end in "movie" or "film".

Examples: Here are some example keyword analyses and identifications for your reference. Please follow this format.\n
Examples:\n
1. Query: "what type of movie genre is the perfect weapon"
Likely Keywords: "the perfect weapon" is a movie name. Likely keywords are [the perfect weapon].\n
2. Query: "show me a movie with the song a whole new world"
Likely Keywords: "a whole new world" is a song name. Likely keywords are [a whole new world].\n
3. Query: "how many movies came out in 2004"
Likely Keywords: "2004" refers to a time period. Likely keywords are [2004].\n
4. Query: "when did mark joffe direct the bounty hunter film that is rated pg"
Likely Keywords: "mark joffe" is a person's name. "bounty hunter" describes a film element. "pg" is a movie rating. Likely keywords are [mark joffe; bounty hunter; pg].\n
5. Query: "is cary grant in any historical films that are a must see"
Likely Keywords: "cary grant" is the name of a person. "historical" defines a genre. "must see" describes a film. Likely keywords are [cary grant; historical; must see].\n
6. Query: "could you show me some part of the new indiana jones movie"
Likely Keywords: "some part" indicates a trailer. "indiana jones" refers to a person. Likely keywords are [some part; indiana jones].\n
7. Query: "what movie is considered the funniest of all time"
Likely Keywords: "funniest of all time" describes a movie. Likely keywords are [funniest of all time].\n
\n---\n

Input: Please analyze and identify potential spans in the following 5 queries:\n
1. Query: "Can you recommend a high-rated sci-fi movie released in the 21st century?"\n
2. Query: "Could you show me the trailer for Sing 2 and tell me the main plot of the movie?"\n
3. Query: "Could you recommend a good robot movie with a strong female character in a damsel in distress situation?"\n
4. Query: "Find me a movie directed by Quentin Tarantino with a strong female lead character"\n
5. Query: "I'm looking for the highest rated horror film from the 2000s, preferably directed by a female director like The Babadook."

Table 21: 3-Stage Data Generation step 2: Generate Candidate Entity Spans with CoT prompt example for MIT-Movie.

Instruction: Here are 5 spoken queries to a dialogue system about movies. Your task is to analyze each query and classify whether the spans of text are named entities of the following entity types:
[Title, Viewers' Rating, Year, Genre, Director, MPAA Rating, Plot, Actor, Trailer, Song, Review, Character].
In particular, for each query with a span of text, please analyze and classify the span of text in context of the query into one of the following categories:
[named Title entity, named Viewers' Rating entity, named Year entity, named Genre entity, named Director entity, named MPAA Rating entity, named Plot entity, named Actor entity, named Trailer entity, named Song entity, named Review entity, named Character entity, named entity of other type, not a named entity].

Examples: Here are some example span analyses and classifications for your reference. Please follow this format.\n
Examples:\n
1. Query: "could you show me some part of the new indiana jones movie"
Text Span: "some part"\nLabel: "some part" indicates a segment of a movie. General request to a segment or clip from a movie counts as a Trailer entity. The span is a named Trailer entity.
Text Span: "indiana jones"\nLabel: "Indiana Jones" is a well-known character. He's the protagonist in the "Indiana Jones" franchise first appearing in the film "Raiders of the Lost Ark". The span is a named Character entity.\n
2. Query: "what type of movie genre is the perfect weapon"
Text Span: "the perfect weapon"\nLabel: "the perfect weapon" is a concrete movie name. That is, "The perfect weapon" is a specific title of a movie. The span is a named Title entity.\n
3. Query: "when did mark joffe direct the bounty hunter film that is rated pg"
Text Span: "mark joffe"\nLabel: "Mark Joffe" is a person's name and he directed a film. Thus, "Mark Joffe" is a director's name. A specific director's name is a Director entity. The span is a named Director entity.
Text Span: "direct"\nLabel: "direct" is a verb describing the action of the director. It is the action of a director. The span is a named entity of other type.
Text Span: "bounty hunter"\nLabel: The span "the bounty hunter film" indicates "bounty hunter" is not a movie name. Thus, "bounty hunter" is a specific plot element for movies. A specific movie theme or storyline counts as a Plot entity. The span is a named Plot entity.
Text Span: "pg"\nLabel: "pg" stands for "Parental Guidance Suggested." This rating is a part of the Motion Picture Association of America (MPAA) film rating system. The span is a named MPAA Rating entity.\n
4. Query: "is cary grant in any historical films that are a must see"
Text Span: "cary grant"\nLabel: "Cary Grant" is a person's name and he is in films. Thus, "Cary Grant" is an actor's name. A specific actor's name is a named Actor entity. The span is a named Actor entity.
Text Span: "historical"\nLabel: "historical" is a specific category or style of film. A specific movie category or style is a Genre entity. The span is a named Genre entity.
Text Span: "historical film"\nLabel: "historical" is suffice to serve as a Genre keyword. The addition of film is redundant and the entire span "historical film" is not a named entity. The span is not a named entity.
Text Span: "must see"\nLabel: "must see" indicates a specific level of popularity or recommendation. "must see" is a specific assessment of a movie by its audience. A specific movie assessment is a Viewers' Rating entity. The span is a named Viewers' Rating entity.\n
5. Query: "show me a movie with the song a whole new world"
Text Span: "a whole new world"\nLabel: "A whole new world" is the name of a song. A song name featured in a movie is a Song entity. The span is a named Song entity.\n
6. Query: "show me viewers' rating and plot summary for a good funny movie from a female filmmaker."
Text Span: "viewers' rating"\nLabel: "viewers' rating" is not an actual movie assessment from viewers. "viewers' rating" merely requests rating information. A flexible request for a movie assessment doesn't count as a Viewers' Rating entity. The span is not a named entity.
Text Span: "plot summary"\nLabel: "plot summary" simply requests storyline information. "plot summary" is not an actual movie theme or plot element. A flexible request for a movie plot doesn't count as a Plot entity. The span is not a named entity.
Text Span: "good"\nLabel: "good" is an adjective that describes "movie". It implies a high quality movie perceived by viewers. The span is a named Viewers' Rating entity.
Text Span: "funny"\nLabel: "movie" is right after "funny", so the span "funny" directly specifies a movie characteristic. It implies a movie in the comedy genre. The span is a named Genre entity.
Text Span: "female"\nLabel: "female" is a adjective describing a characteristic of "filmmaker". It is a preference, not a female filmmaker's name. The span is not a named entity.
Text Span: "female filmmaker"\nLabel: "female filmmaker" is a category of directors. It is not the name of a specific director. The span is not a named entity.\n
7. Query: "how many movies came out in 2004"
Text Span: "2004"\nLabel: "2004" is a year. It specifies a particular time duration when the movies were released. A time duration of movie release counts as a Year entity. The span is a named Year entity.\n
8. Query: "what movie is considered the funniest of all time"
Text Span: "funniest of all time"\nLabel: "funniest of all time" is a detailed critique or opinion about a movie. A detailed movie comment from viewers or critics is a Review entity. The span is a named Review entity.\n
\n---\n

Input: Please analyze and classify spans of text in the following 5 queries:\n
1. Query: "Show me the trailer of the animated movie Toy Story released in 1995"\nText Span: "Toy Story"\nText Span: "1995"
2. Query: "Show me a trailer for the highest-rated comedy movie of 2020."\nText Span: "highest-rated"\nText Span: "comedy"\nText Span: "2020"
3. Query: "What is the viewers' rating for Manchester by the Sea?"\nText Span: "Manchester by the Sea"\n
4. Query: "Show me a trailer for the movie Titanic, which features the song My Heart Will Go On, and provide a cinema evaluation of the film."\nText Span: "trailer"\nText Span: "Titanic"\nText Span: "My Heart Will Go On"\nText Span: "cinema evaluation"\n
5. Query: "Who directed the supernatural thriller movie 'The Others' that was released in 2001?"\nText Span: "supernatural thriller"\nText Span: "The Others"\nText Span: "2001"

Table 22: 3-Stage Data Generation step 3: Generate Candidate Span Labels with CoT prompt example for MIT-Movie.

Dataset	Approach	F1
CoNLL-2003	2-Stage (step 2 vanilla)	74.9
	3-Stage (Step 2 vanilla, Step 3 CoT)	**76.8** +1.9
MIT-Movie	2-Stage (step 2 vanilla)	**70.6**
	3-Stage (Step 2 cot, Step 3 CoT)	70.5 -0.1

Table 23: 3-Stage Data Generation with CoT Performance Comparison. 2-Stage and 3-Stage generation uses the same pool of sentences (i.e. same step 1 output). The 2-Stage reference setups don't use CoT prompting. *vanilla* and *CoT* refers to without and with chain-of-thought prompting respectively.

Approach		F1
Correction Source	**Ranking Function**	
Diversify Y (vanilla)		72.7
LLM	log-prob	**74.1** +1.4
	loss	70.7 -2.0
Manual	log-prob	**74.9** +2.2
	loss	73.2 +0.5

Table 24: Entity Annotation Uncertainty Ranking Function performance comparison on MIT-Movie. F1 gains/drops are w.r.t before annotation corrections.

B.3.3 Experiment

Compare the quality of entity annotations selected via *log-prob* and *loss*, we conducted preliminary experiment on data generated for MIT-Movie via *Diversify Y (vanilla)*. The experiment setup including annotation-selection hyperparameters for *log-prob* remain the same as in the main experiments (§4.2). For *loss*, we do not perform any filtering ($P_c = 0$) and simply select the top-ranking $T_c = 20\%$ for each entity class normalized by entity proportions. In addition to LLM self-correction,

#Annot. examined	#Invalid LLM Correct.	Manual Correction Type Counts			
		Wrong Span	Wrong Type	NA	*total*
707	**136**	**59**	**82**	137	278
707	131	52	72	**254**	**378**

Table 25: Entity Annotation Uncertainty Ranking Function statistics on MIT-Movie. *total* refers to the total number of annotations corrected.

CoNLL-2003
President Biden announces new infrastructure plan President Biden signs new infrastructure bill into law. President Biden announces new infrastructure plan to rebuild roads and bridges.
Former President Obama to release new memoir. Former President Obama to release memoir this fall.
Taylor Swift releases new album Taylor Swift released a new album, causing a frenzy among fans.
WikiGold
The Golden Gate Bridge is a suspension bridge spanning the Golden Gate strait, the mile-wide, three-mile-long channel between San Francisco Bay and the Pacific Ocean. The Golden Gate Bridge is a suspension bridge spanning the Golden Gate, the one-mile-wide strait connecting San Francisco Bay and the Pacific Ocean.

Table 26: Example groups of LLMs-generated sentences that are syntactically and semantically similar.

one of the authors manually annotated the selected entity annotations. The results are shown in Table 24. We observe that whether entity annotations are corrected via LLMs or manually, *log-prob*-selected annotations noticeably yield better downstream F1.

Treating manual corrections as ground truths, we further examine the corrections as shown in Table 25. We observe that, comparing entities annotations selected via *log-prob* and *loss*, (1) the ratio of invalid annotation corrections that LLMs make is the same, and (2) *loss* selects more wrong entity annotations and identifies significantly more non-named-entities. Thus, we conclude that in this case, *loss* scoring selects entity annotations that less significantly contribute to performance gains. We hypothesize that the downstream model might identify more errors that it already resolved during the training process (Appendix A.5.2). We look forward to more comprehensive investigations on selecting influential entity annotations in future work.

C Future Work

More Analysis Given the initial success of ProgGen, it would be interesting to explore how does sample diversity contribute to performance. Notably, we hypothesized that distilling entity knowledge is more challenging in specific domains (§4.3.3), experiments on how scaling the number of training samples influence downstream performance can provide more insights.

Cost analysis trading downstream performance with token querying cost is also interesting to investigate. For example, how would hyperparameters such as batch size (for sample generation and correction generation) influence the quality of the generated dataset?

ProgGen Extensions Cursory manual examinations on the generated datasets found several cases where LLMs generate syntactically and semantically similar sentences (examples in Table 26). This implies that (1) the number of generated samples may *not correlate linearly* with sample diversity and (2) the downstream training process is biased towards the "duplicated" groups of samples. Future work can explore how to select a diverse subset of generated sentences for annotations and downstream training.

In our main experiments (§4.3), we found that Diversify Y variants work well, where the *vanilla* variants perform the best across almost all datasets and the *latent* variant leads to more robust performances. One extension can explore merging these variants via sampling to get the best of both worlds and further, generating entity pools conditioned on additional attribute dimensions from *Diversify X*.

Our current self-correction pipeline only handles the most common cases of entity annotation errors. More more nuanced annotation errors can be incorporated, including (ordered by occurrence from cursory inspection): (1) both wrong span and wrong type, and (2) two valid annotations. We show example wrong entity annotations of such case in Table 27. Our work mainly focused on reducing false positives since we found this is a more prevalent issue, future work can also explore looking at existing annotations given a sample and add additional annotations (reducing false negative).

Better Annotations We note (1) the strong performance of Simple Prompt (Table 4) and that (2) Simple Prompt have relative less number of generated samples in downstream training due to deduplication (Appendix E.2). Thus, we hypothesize that Simple-Prompt-generated NER samples may have more accurate annotations, despite less diverse, possibly due to an implicit self-consistency-like (Wang et al., 2023b) "check" where LLMs generate the same sentence multiple times. In particular, one of two cases will happen: (1) all gen-

Span	Type ✗	Type ✓
CoNLL-2003		
The **Prime Minister of India** has announced plans to invest in infrastructure to boost economic growth.	Person	Location
California governor signs bill to address homelessness crisis.	Person	Location
The **Secretary-General of the United Nations** is calling for an immediate ceasefire in the region.	Person	Organization
Former **CEO of Google** to testify before Congress.	Person	Organization
WikiGold		
The Moscow Kremlin serves as the official residence of the **President of the Russian Federation**.	Person	Organization
Ankara-based organization launches new initiative to combat poverty in the region.	Organization	Location
MIT-Movie		
Who directed the **latest paranormal thriller** film to hit theaters?	Title	Genre
Can I watch a movie snippet of the **newest Marvel** film?	Title	Genre
Could you provide me with a trailer for the **new Tarantino** film?	Title	Director
Could you recommend a popular **Sports underdog** story movie from the 1990s?	Genre	Plot
What is the **highest** Viewers' Rating movie with Olaf in it?	Review	Viewers' Rating
Who directed the **classic film** about unrequited love	Viewers' Rating	Genre
I heard there's a new film coming out with **Zendaya and Tom Holland**, do you know if it's a PG-13 movie?	Actor	Actor, Actor
what are some top-rated **M/PG** movies from the 90s	MPAA Rating	MPAA Rating, MPAA Rating
MIT-Restaurant		
What time does the **sushi place** close?	Restaurant Name	Cuisine
Tell me a place where I can have a **nice brunch** on weekends.	Amenity	Hours
What are the operating hours for the **upscale Italian restaurant** in the downtown area?	Restaurant Name	Price, Cuisine
What are the opening hours of the **nearest Thai restaurant**?	Location	Location, Cuisine

Table 27: Example LLM entity annotation samples that showcase annotation errors that are more malformed/current correction pipeline don't handle. We **boldface** LLMs-annotated and wrong entity spans and underline the expected and correct entity spans (*Span*). In *Type*, we show the corresponding entity types. The 2 different annotations are also color coded (**LLMs/wrong** (✗) vs expected/correct (✓)).

erated sentences have the same annotations and thus is part of the processed dataset, or (2) some of the generated annotations are conflicting and thus such samples are dropped from the processed dataset. We also observed that despite many wrong entity annotations GPT-3.5 make, GPT-3.5 may have a higher accuracy for easier tasks such as simply checking if terms like "Nigerian President" is a named entity. Thus, despite incorporating CoT and stage-wise annotation are shown non-trivial (Appendix B.2), future directions include exploring more variants, including (1) combining re-annotation (Appendix B.2) with self-consistency-like (Wang et al., 2023b) majority voting to reduce noises (in entity class, demo & input ordering) in LLM annotations and (2) least-to-most-promoting-like (Zhou et al., 2023) step-wise annotation correction to reduce bias.

Additionally, the task of selecting potentially *all* wrong entity annotations in the generated dataset in a *single trial* may be too challenging. In light of the hypothesis that LLMs may not correct enough wrong annotations of the same failure class to significantly influence training (Appendix A.5.2), future work can explore propagating "confident" LLM corrections to the non-selected pool of entity annotations, leading to better annotated datasets.

	CoNLL -2003	Wiki Gold	MIT-Movie	MIT-Restaurant
Duplicate				
#	1,349 / 179 / 147	7	31 / 1	1 / 0
%	9.8 / 5.5 / 4.2	0.4	0.3 / 0.0	0.0 / 0.0
Conflicting				
#	2 / 2 / 4	0	16 / 0	1 / 0
%	0.0 / 0.1 / 0.1	0.0	0.2 / 0.0	0.0 / 0.0

Table 28: Samples dropped from the original datasets due to *duplicate* or *conflicting*. Separated by slash, we show the number of samples dropped for the original train/dev/test or train/test splits respectively if available.

D Additional Implementation Details

D.1 NER Sample & Dataset Processing

This section details how LLMs' free-form responses are parsed into structured NER samples for training.

NER Sample Parsing We use regular expression (`re`) patterns to match and extract the generated sentences and list of entity name-type pairs. We drop malformed NER generations. Primary reasons for invalid NER samples include (1) overlapping entity spans annotated, (2) unseen entity type annotated, and (3) annotated entity spans not found in the generated sentence. For the few-shot ICL baseline, to ensure a prediction is made for each sample in the original test set, we filter out the subset of malformed entity annotations in the generated annotation list (as opposed to filtering out the test sample entirely).

Despite that LLMs are instructed to (1) annotate each multi-occurring named entities multiple times (Appendix B.1) and (2) order the annotated spans by occurrence, we found LLMs sometimes don't follow the instructions. We show one rare case where LLMs annotate a multi-occurring span with different annotations below:

> Sentence: `Can you tell me about the character Harry Potter from the Harry Potter series?`
> Named Entities: `[Harry Potter (Character), Harry Potter (Title)]`

Thus, we pragmatically (1) duplicate single annotations for multi-occurring entities as much as it occurs (as a result, they have the same entity type), and (2) re-order the entity annotations to match the occurrence order in the generated sentence as needed.

BIO Format Mapping For downstream model training, we tokenize the sentences and entity spans by splitting on any whitespace and punctuations (and then process NER tags into BIO format). Conversely, since the original datasets we studied are in the BIO format and we need the demo samples in a natural-language format in prompts (Appendix B.1), we detokenize the original samples via the `sacremoses` package.

Deduplication We observed that the original datasets may contain duplicate samples or conflicting annotations. Thus, we filter out these samples. In particular, we (1) keep only 1 copy of duplicate samples (same sentence and same annotations; *duplicate*), and (2) drop all conflicting samples, i.e. samples with the same sentence but different annotations (*conflicting*). The number of dropped samples are summarized in Table 28.

D.2 Data Generation

D.2.1 Diversify X

Generate Attribute Values Since ChatGPT generates long-form responses, the responses from attribute dimension generation also include additional attribute descriptions and example values. We may select high-quality generated descriptions and examples and include them in the attribute value generation prompt (example in Appendix F.1.1).

Construct Diversity Requirement The dataset-independent "topic" dimension is always included/sampled in the diversity requirements, for simplicity of incorporating with the *Diversify X + Y* variant, passing the sampled topic value to the named entity pool.

During attribute dimension sampling, we note that certain dimensions may have similar or conflicting values. For example, for MIT-Restaurant, attribute dimensions "dietary restriction" and "special offering" may both contain dietary needs. Thus, we group such "conflicting" attributes together and ensure at most one attribute dimension in the conflicting set is sampled. The conflicting sets for each dataset are shown in Table 29.

Another reason for keeping a small R_x ($\in [1, 2]$; §3.2.1) is that we expect limited instruction-following for GPT3.5 given a larger number of diversity requirements, where too many instructions presumably lead to lower-quality samples (Ouyang et al., 2022).

D.2.2 Diversify Y

Generate Entities We generally use a smaller M for *latent* Diversify Y compared to the *vanilla* variant. Intuitively, this is because with a more detailed requirement (conditioned on both an entity class and a domain-specific topic), there tend to be less relevant named entities.

Construct Diversity Requirement For each entity class, we sample a few entities $\{e_i\}_1^{P_c}$ for each entity class c, where P_c is sampled uniformly in $[0, 3]$ to ensure entity class balance. We then union all named entities sampled and takes a subset on the named entities as if in Bernoulli trials where we enforce the expected number of remaining entities to be R_y.

In preliminary experiments, we found that including the entity type corresponding to each sampled entity in the entity inclusion requirement consistently yield worse performance (~3%). We hypothesize that (1) LLM-generated entity pool may not be accurate, i.e. named entities sampled may not belong to the given type, leading to incorrect requirements, and (2) specifying the entity type for spans that may belong to multiple entity types based on context (e.g. Burger as a Dish or a Cuisine entity) may limit the LLMs' capability in following the instruction. In both cases, LLMs may generate samples with wrong annotations, leading to worse downstream performance.

D.2.3 NER Sample Generation

In our API calls, we use temperature $t = 1$ and parameter top $p = 1.0$ unless otherwise specified. For a fair comparison, for each approach, we ensure LLMs generated 1.8K samples. We filter out malformed samples LLMs generated and de-duplicate the remaining, valid samples (Appendix D.1), resulting in different number of NER samples in the NER dataset for training. For Simple Prompt, we request ($L = 50$) samples in each prompt (thus ~36 API calls) to further save token querying cost (since less prompt tokens). We found that GPT-3.5 frequently fails to generate 50 samples requested so we made additional API calls until 1.8K samples are found (without checking for validity and duplication). For all diversity setups, we request 3 samples in each prompt to sample more diverse configurations, making ~600 configurations total for each variant. We use $\mathrm{E}\left[R_y\right] = 4.5$ for MIT-Movie and $\mathrm{E}\left[R_y\right] = 1.5$ for other datasets in our experiments, which empirically performs well in early experiments.

Few-shot ICL experiments are greedily decoded ($t = 0$).

D.2.4 Self-Correction

For LLM self-correction, each prompt includes 3 samples to correct and we use greedy decoding ($t = 0$). When crafting the self-correction annotation instructions and demos, we generally follow the rule below: 1) if a failure case appears more than 5 times, we include a demo correction sample, and 2) if a failure case appears more than 3 times, we summarize the error case in the annotation instruction. We provide no more than 6 error cases in the instruction and no more than 6 demos. The typical number of wrong-annotation cases in instruction and demos are about 3.

Less common failure cases are ignored to prevent overly tuning for each generated dataset. We further note that (1) ChatGPT3.5 can correct many "easier" wrong entity annotations 0-shot, i.e. without any instruction and demo, and ChatGPT3.5 is biased sometimes and its capacity and instruction following capabilities are limited, where (2) a fraction of LLM Self-Corrections are invalid and (3) additional demos may not lead to additional valid corrections.

D.3 Downstream Model Training

The maximum token length for each sequence is 144. We use the standard training practice and cross entropy loss in PyTorch. All models are trained with NVIDIA GeForce RTX 2080 Ti GPUs.

We use DeBERTa and learning rate of $4e^{-5}$ due to cursory hyperparameter search via supervised learning. We searched over BERT, RoBERTa and DeBERTa (Devlin et al., 2019; Liu et al., 2019; He et al., 2023) and learning rate in [$1e^{-5}$, $2e^{-5}$, $3e^{-5}$, $4e^{-5}$, $5e^{-5}$, $5e^{-6}$] and found this setup consistently yield relatively high F1 scores across all 4 datasets studied.

D.4 Special Handling

CoNLL-2003 and WikiGold both contain over 1% negative samples where as MIT-Movie and MIT-Restaurant don't. Thus, we include a negative demo sample for CoNLL-2003 and WikiGold (§4.1). We also add an explicit instruction in the data generation prompt for these datasets in case no named entities is found in the generated sentences (as shown in Table 1). For WikiGold, we used the same train/dev/test split as in Meng et al. (2021).

For MIT-Movie and MIT-Restaurant, we lowercase all generated NER samples before (de-duplication and) downstream training, since (1) the original dataset samples are all in lowercase, and (2) lowercasing leads to noticeable gains (~10%). For MIT-Movie entity pool generation, we found that GPT-3.5-generated entities don't belong to the respective entity classes for several classes, so we additionally provided entity definitions and some demos in the prompt as necessary.

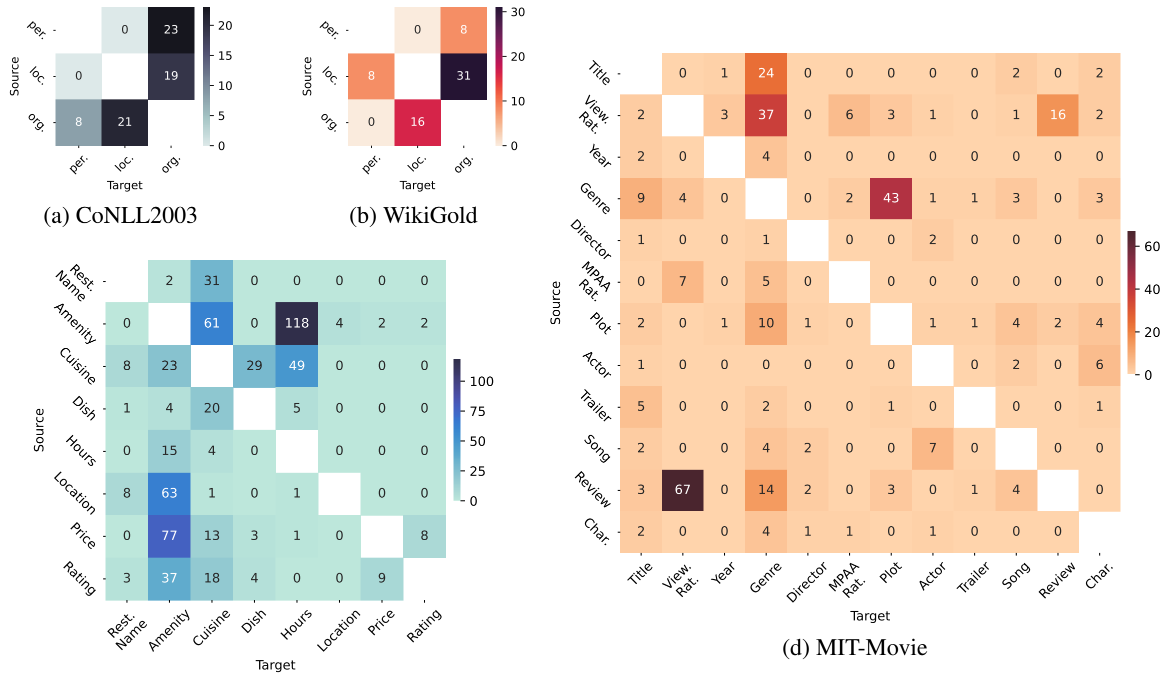
E Additional Data Statistics

E.1 Diversity Requirements

We show summary statistics of the diversity requirement configurations for *Diversify X* and *Diversify Y* respectively in Tables 29 and 30.

E.2 Sample Generation

We show summary statistics of generated datasets (before LLM Self-Correction) for Simple Prompt and our diversity approaches in Table 31. We observe consistent trends where we make the following (unsurprising) observations: (1) Diversify X leads to longer generated sentences, indicating more tokens are needed to satisfy the semantic requirements, (2) Diversify Y shows increased number of distinct named entities, (3) Simple Prompt in general domain have less samples processed, mostly due to de-duplication, and (4) Explicit diversity requirements yield improved diversity where



(a) CoNLL2003

(b) WikiGold

(c) MIT-Restaurant

(d) MIT-Movie

Figure 5: LLM Entity Type Self-Correction counts by dataset.

Diversify X+Y yields the highest diversity as expected.

E.3 LLM Self-Correction

For each dataset, we show summary statistics for distribution of LLM correction types, across different diversity setups and across entity classes, in Tables 32 and 33. Entity type corrections to "other" are considered as entity annotation drops. For entity type corrections to relevant entity types within the dataset, we show the count of corrections in Figure 5.

F Additional Prompts and Samples

In this section, we show (1) prompt templates in F.1, (2) some prompt and response samples for the CoNLL-2003 dataset for space constraints in F.2, (3) a small subset of the generated diversity configurations for each dataset in F.3, and (4) selected 1-shot demo samples and a few generated NER samples for each dataset and each diversity variant in F.4. The full prompts and LLM responses for (1) diverse requirement generation, (2) NER sample generation and (3) entity correction generation, and the corresponding processed data are all available in our repository[20].

[20] `https://github.com/StefanHeng/ProgGen/reproduce`

F.1 Prompt Templates

F.1.1 Diversify Sentence

Generate Attribute Dimensions What do you think are important attributes to generate some diverse `{domain sample description}`? Examples: `{example attributes}`... Let's think step by step.

Generate Attribute Values Suppose you are `{domain description}`. List `{#generate}` different `{attribute dimension}` for `{domain sample description}`. Some examples are `{example attribute values}`.

F.1.2 Diversify Entities

Generate Entities (vanilla) Suppose you are `{domain description}`. Please generate `{#generate}` diverse named entities in `{domain sample description}` that can be categorized as `{entity class}`.

Generate Entities (latent) Suppose you are domain + latent topic description. Please generate `{#generate}` diverse named entities in `{domain sample description}` that can be categorized as `{entity class}`.

F.1.3 Few-Shot ICL baseline

Suppose you are `{domain description}`. You are given `{domain sample description}`. Please identify all named entities occurred that belong to one of the following entity types:
`{entity types in the dataset}`.
Please list such named entities with the corresponding entity types on the following line, in the order of occurrence.\n
Here are some example annotations for your reference. Please follow this format.\n
Examples:\n
`{(sentence, entity annotations) demo samples}`\n
---\n
Please annotate the following `{domain sample prefix}`:\n
`{domain sample prefix}`: "`{sample}`"

Dataset	Attr. Dim.	V	P	T
CoNLL -2003	News Topic*	44	-	1.0K
	Writing Style	22	0.4	
Wiki Gold	Topic*	46	-	1.2K
	Writing Style	26	0.4	
MIT-Movie	Query Category*	10	-	450
	User Demographic	18	0.3 }	
	User Emotion	17	0.3 }	
	Query Language	9	0.3 }	
MIT-Restaurant	Meal Category*	10	-	12K
	Price Range	5	0.2 }	
	Ambiance	20	0.2 }	
	Service Mode	4	0.1 }	
	User Demographic	20	0.1 }	
	Dietary Restriction	10	0.1 }	
	Special Offering	8	0.1 }	

Table 29: Diversify X Requirement Configurations Summary Statistics. V and P and refer to the number of unique attribute values and sampling probability for each attribute dimension respectively. T refers to the total number of unique diversity configurations. The dataset-independent "topic" attribute dimensions have trailing asterisks* and are always included in the diversity requirements. Conflicting attribute dimensions are grouped with braces} (Appendix D.2.1).

Var.	#unique entities by class	N
CoNLL-2003		
van.	{person: 132; location: 86; organization: 132}	350
lat.	{person: 45.1, location: 46.7, organization: 43.5}	6.0K
WikiGold		
van.	{person: 119; location: 310; organization: 282}	711
lat.	{person: 18.1; location: 26.8; organization: 22.0}	3.0K
MIT-Movie		
van.	{Title: 184; Viewers' Rating: 257; Year: 98; Genre: 133; Director: 120; MPAA Rating: 40; Plot: 168; Actor: 110; Trailer: 66; Song: 125; Review: 249; Character: 170}	1.7K
lat.	{Title: 100.9; Viewers' Rating: 127.2; Year: 98.5; Genre: 70.2; Director: 92.1; MPAA Rating: 13.3; Plot: 90.0; Actor: 84.3; Trailer: 30.0; Song: 78.4; Review: 123.7; Character: 124.1}	10K
MIT-Restaurant		
van.	{Restaurant Name: 199; Amenity: 170; Cuisine: 96; Dish: 188; Hours: 221; Location: 221; Price: 284; Rating: 308}	1.7K
lat.	{Restaurant Name: 35.5; Amenity: 33.6; Cuisine: 19.8; Dish: 32.5; Hours: 37.8; Location: 26.9; Price: 33.8; Rating: 34.4}	3.9K

Table 30: Diversify Y Requirement Configurations Summary Statistics. *Var.* refers to the Diversify Y variants, one of *van.* or *lat.* for *vanilla* and *latent* respectively. N refers to the total number of entities in the entity pool. For the latent variant of Diversity Y, we show number of entities values averaged by the number of "topic" dimension values.

F.2 Prompt Examples for CoNLL-2003

F.2.1 Diversify Sentence

Generate Attribute Dimensions

Prompt to ChatGPT

What do you think are important attributes to generate some diverse news from Reuters? Examples: topic, writing style...
Let's think step by step.

Example Response from ChatGPT

Creating diverse news content, especially for a global news organization like Reuters, involves several key attributes. Let's break them down step by step:

1. **Topic Selection**: Covering a wide range of topics is crucial. This includes politics, economics, technology, health, environment, culture, sports, and human interest stories. The goal is to cater to varied interests and provide a holistic view of global events.
2. **Geographical Diversity**: Ensuring that news covers different regions of the world is important. This involves not just focusing on major powers like the US, China, or the EU, but also giving attention to less covered areas like Africa, South America, and smaller Asian and European countries.
3. **Writing Style**: Reuters is known for its straightforward, factual reporting style. However, incorporating different writing styles like in-depth analysis, opinion pieces, and feature stories can enrich the content.
4. **Source Variety**: Utilizing a wide array of sources, from government officials to local activists, experts in various fields, and ordinary citizens, can provide a more comprehensive perspective on issues.
5. **Multimedia Content**: Incorporating videos, podcasts, and interactive graphics can enhance the storytelling and appeal to a broader audience.
6. **Inclusivity and Representation**: Reflecting diverse voices in terms of ethnicity, gender, age, socio-economic status, and cultural backgrounds in both the news stories and among the journalists.
7. **Fact-Checking and Accuracy**: Maintaining high standards of journalism by rigorously fact-checking information and presenting unbiased, accurate news.
8. **Language Accessibility**: Offering news in multiple languages can reach a broader global audience.
9. **Investigative Journalism**: Going beyond surface-level reporting to uncover deeper truths, especially in areas like corruption, human rights abuses, and environmental issues.
10. **Audience Engagement**: Encouraging reader interaction through comments, social media, and possibly reader polls or Q&A sessions.
11. **Ethical Reporting**: Upholding journalistic ethics by respecting privacy, avoiding sensationalism, and reporting responsibly on sensitive issues.
12. **Technology and Innovation**: Leveraging new technologies like AI, VR, or AR for more immersive and interactive news experiences.

By focusing on these attributes, Reuters or any other news organization can produce diverse, engaging, and high-quality news content that appeals to a global audience.

Generate Attribute Values

Prompt to ChatGPT

Setup	N	$\|V\|$	$\overline{\|L\|}$	$\|E\|$	$\overline{E}$
CoNLL-2003					
Simple Prompt	1.6K	2.5K	8.7	549	1.3
Diversify X	1.7K	3.6K	11.2	972	1.5
Diversify Y (vanilla)	1.7K	3.0K	10.5	689	1.5
Diversify Y (latent)	1.7K	3.6K	11.0	940	1.5
Diversify X+Y	1.7K	4.2K	13.2	1.0K	1.5
WikiGold					
Simple Prompt	1.3K	3.1K	15.2	894	2.2
Diversify X	1.6K	5.3K	20.9	975	1.7
Diversify Y (vanilla)	1.6K	5.1K	17.6	1.4K	1.9
Diversify Y (latent)	1.7K	5.7K	18.4	1.6K	1.9
Diversify X+Y	1.7K	6.3K	22.7	1.4K	1.7
MIT-Movie					
Simple Prompt	1.7K	618	9.7	896	1.8
Diverse-X	1.4K	1.3K	14.1	940	2.0
Diverse-Y (vanilla)	1.7K	1.9K	12.3	1.5K	2.1
Diverse-Y (seeded)	1.7K	2.0K	12.1	1.5K	2.1
Diverse-X+Y	1.5K	2.2K	16.0	1.5K	2.1
MIT-Restaurant					
Simple Prompt	1.7K	704	112	10.2	2.2
Diverse-X	1.6K	638	191	12.6	2.8
Diverse-Y (vanilla)	1.7K	1.1K	90	12.4	2.4
Diverse-Y (seeded)	1.7K	1.0K	86	12.5	2.5
Diverse-X+Y	1.6K	1.0K	145	13.6	2.9

Table 31: Generated NER Dataset Statistics. N refers to the number of processed NER samples. $|V|$ refers to the vocabulary size. $\overline{L}$ refers to the average token length of generated sentences. $|E|$ refers to number of distinct named entities. $\overline{E}$ refers to the average number of named entities for all sentences with at least 1 named entity annotated. Tokenization via splitting on any whitespace and punctuation.

Setup	Correction Type			%	#
	NA	Span	Type		
CoNLL-2003					
Simple Prompt	41	14	12	17.4	386
Diversify X	107	5	15	27.3	466
Diversify Y (vanilla)	42	8	12	15.8	392
Diversify Y (latent)	53	10	21	19.1	439
Diversify X+Y	83	9	11	28.1	366
WikiGold					
Simple Prompt	172	8	21	36.8	546
Diversify X	179	8	4	44.0	434
Diversify Y (vanilla)	95	9	17	20.1	601
Diversify Y (latent)	88	8	18	20.6	553
Diversify X+Y	92	4	3	20.7	479
MIT-Movie					
Simple Prompt	125	18	59	35.0	577
Diverse-X	127	34	72	40.5	576
Diverse-Y (vanilla)	131	47	85	36.3	725
Diverse-Y (seeded)	156	37	70	37.5	701
Diverse-X+Y	169	27	64	42.1	618
MIT-Restaurant					
Simple Prompt	41	44	112	26.5	744
Diverse-X	29	62	191	31.0	909
Diverse-Y (vanilla)	49	67	90	25.0	824
Diverse-Y (seeded)	43	59	86	22.7	829
Diverse-X+Y	36	50	145	24.7	935

Table 32: Distribution of LLM Correction Types grouped by diversity variant. NA, Span, Type refers to dropping entity annotation, correcting entity span, and correcting entity type respectively. % refers to the percentage of entity annotations LLMs corrected, among total number of entity annotations prompted. # refers to the total number of entity annotations prompted.

Suppose you are a news writer for Reuters. List 30 diverse news categories for news articles. Some examples are [health, local stories, international events, environment, business, sports, politics, economics, breaking news, culture, science, technology, human interest stories].

Example Response from ChatGPT

1. World News
2. Politics
3. Technology
4. Sports
5. Entertainment
6. Health
7. Business
8. Environment
9. Breaking News
10. Human Interest Stories
11. Local Stories
12. Economics
13. International Events
14. Science
15. Education
16. Travel
17. Crime
18. Fashion
19. Food
20. Religion
21. Art
22. Music
23. Film
24. Literature
25. Automotive
26. Lifestyle
27. Weather
28. Wildlife
29. Education
30. Immigrant Stories

Prompt to ChatGPT

Suppose you are a news writer for Reuters. List 8 different writing styles for news articles. Some examples are [op-eds, opinion pieces, feature stories, in-depth analysis].

Example Response from ChatGPT

Entity Type	Correction Type			%	#
	NA	Span	Type		
CoNLL-2003					
person	102	30	23	30.8	504
location	97	12	19	18.5	693
organization	127	4	29	18.8	852
all	326	46	71	21.6	2,049
WikiGold					
person	81	14	8	21.5	479
location	286	17	39	28.5	1,199
organization	259	6	16	30.1	935
all	626	37	63	27.8	2,613
MIT-Movie					
Title	29	14	29	16.1	446
Viewers' Rating	105	6	71	52.4	347
Year	87	6	6	43.6	227
Genre	26	69	66	27.4	588
Director	50	1	4	34.8	158
MPAA Rating	54	7	12	63.5	115
Plot	151	10	26	52.5	356
Actor	22	0	9	27.4	113
Trailer	29	34	9	34.3	210
Song	80	4	15	61.1	162
Review	24	11	94	39.4	327
Character	51	1	9	41.2	148
all	708	163	350	38.2	3,197
MIT-Restaurant					
Restaurant Name	3	4	33	25.2	159
Amenity	85	92	187	28.1	1,297
Cuisine	20	72	109	24.2	830
Dish	3	21	30	21.0	257
Hours	32	4	19	13.5	408
Location	6	70	73	30.2	494
Price	16	7	102	45.6	274
Rating	33	12	71	22.2	522
all	198	282	624	26.0	4,241

Table 33: Distribution of LLM Correction Types grouped by entity type.

1. Hard news reporting
2. Investigative journalism
3. Informative news articles
4. Human interest stories
5. Broadcast news
6. Long-form journalism
7. Photojournalism
8. Data-driven journalism

F.2.2 Diversify Entities

Generate Entities (vanilla)

Prompt to ChatGPT

Suppose you are a news writer. Please generate 50 diverse named entities in news articles that can be categorized as location.

Example Response from ChatGPT

1. New York City
2. Eiffel Tower
3. Sahara Desert
4. Mount Everest
5. Great Wall of China
6. Rio de Janeiro
7. Sydney Opera House
8. Taj Mahal
9. Amazon Rainforest
10. Grand Canyon
11. Tokyo
12. Moscow
13. London
14. Paris
15. Rome
16. Antarctica
17. Hawaii
18. Dubai
19. Bali
20. Barcelona
21. African Savannah
22. Las Vegas
23. Machu Picchu
24. Venice
25. Yellowstone National Park
26. Cairo
27. Galapagos Islands
28. Himalayas
29. Easter Island
30. The Bahamas
31. Mount Kilimanjaro
32. Norwegian Fjords
33. Great Barrier Reef
34. Prague
35. Istanbul
36. Vatican City
37. San Francisco
38. Cape Town
39. Athens
40. Sydney
41. Bangkok
42. Buenos Aires
43. Montreal
44. Vancouver
45. Toronto
46. New Delhi
47. Cairo
48. Singapore
49. Sydney
50. Shanghai

Generate Entities (latent)

Prompt to ChatGPT

Suppose you are a news writer for Entertainment. Please generate 15 diverse named entities in news articles that can be categorized as person.

Example Response from ChatGPT

1. Tom Hanks
2. Beyoncé
3. Brad Pitt
4. Rihanna

5. Will Smith
6. Taylor Swift
7. Leonardo DiCaprio
8. Oprah Winfrey
9. Jennifer Lopez
10. Chris Hemsworth
11. Meryl Streep
12. Dwayne "The Rock" Johnson
13. Kim Kardashian
14. Hugh Jackman
15. Lady Gaga

F.2.3 Few-Shot ICL baseline

Suppose you are a news writer. You are given a sentence from a news story. Please identify all named entities occurred that belong to one of the following entity types:
[person, location, organization].
Please list such named entities with the corresponding entity types on the following line, in the order of occurrence.\n

---

Here are some example annotations for your reference. Please follow this format.\n
Examples:\n
1. Sentence: "BASEBALL - DODGERS WIN FIFTH STRAIGHT."
Named Entities: [DODGERS (organization)]\n
2. Sentence: "With 2.5 percent gross domestic product growth expected for 1996, new job growth should slowly lower the unemployment rate over the rest of the year."
Named Entities: []\n
3. Sentence: "A.de Silva not out 49"
Named Entities: [A.de Silva (person)]\n
4. Sentence: "Saudi Arabia executes Pakistani man."
Named Entities: [Saudi Arabia (location)]\n

Please annotate the following sentence:\n
Sentence: "Japan began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday."

F.2.4 LLM Self-Correction

We show annotation instructions and demo pools for the 3 entity classes: Person, Location and Organization. These are for illustrative purposes, encompassing annotation errors in all diversity variants. Please refer to our repository for the illustrative prompts for all datasets and the exact prompts used for each generated dataset.

Person

Here are 3 sentences from news articles. Your task is to classify whether the span of text is a named entity of type person.
In particular, for each query with a span of text enclosed in double braces, please classify the enclosed span of text in context of the query into one of the following categories:
- (A). The span is a named entity of type person
- (B). The span contains a named person entity but the span boundary is not precise
- (C). The span is a named entity but the category is not person
- (D). The span is not a named entity

If the label is (B), you should provide the correct span boundary.
If the label is (C), you should provide the correct entity type. Pick an entity type from the following list:
[location, organization, other].\n

A named person entity must be the name of a person. Only first names, last names, and full names are considered named person entities.
A named person entity should not have any starting titles such as "President", "Prime Minister", "Mayor", "Professor" and "Dr".
(Governmental, political or executive) titles such as "Prime Minister of Australia", "President of Iceland" and "CEO of Google" are not relevant named entities.
General reference to a person or people such as "actress", "chef", "CEO", "woman", "high school student" are not named entities.\n

---

Here are some example span classifications for your reference. Please follow this format.\n
Examples:\n
1. Sentence: "Canadian {{Prime Minister Justin Trudeau}} visits Indigenous communities."
Text Span: "Prime Minister Justin Trudeau"
Label: (B). Wrong Boundary. The correct span boundary is "Justin Trudeau".\n
2. Sentence: "Newly elected {{president}} pledges to address climate change."
Text Span: "president"
Label: (D). Not a Named Entity.\n
3. Sentence: "{{CEO of Amazon}} steps down from role."
Text Span: "CEO of Amazon"
Label: (D). Not a Named Entity.\n
4. Sentence: "{{CEO}} of Dubai-based company arrested for fraud."
Text Span: "CEO"
Label: (D). Not a Named Entity.\n

Please classify the following 3 spans of text:\n
1. Sentence: "{{Cindy Gruden}} appointed as the new CEO of a major automotive company."
Text Span: "Cindy Gruden"\n
2. Sentence: "{{Justin Trudeau}} meets with CEOs of major tech companies to discuss investment opportunities in Canada."
Text Span: "Justin Trudeau"\n
3. Sentence: "{{Angela Merkel}} visits Silicon Valley to discuss collaboration with automotive tech companies."
Text Span: "Angela Merkel"\n

Location

Here are 3 sentences from news articles. Your task is to classify whether the span of text is a named entity of type location.
In particular, for each query with a span of text enclosed in double braces, please classify the enclosed span of text in context of the query into one of the following categories:
- (A). The span is a named entity of type location
- (B). The span contains a named location entity but the span boundary is not precise
- (C). The span is a named entity but the category is not location
- (D). The span is not a named entity

If the label is (B), you should provide the correct span boundary.
If the label is (C), you should provide the correct entity type. Pick an entity type from the following list:
[person, organization, other].\n

A named location entity must be the name of a location.
Generic location references such as "city hall", "community center", "downtown", "major cities" and "bank" are not named

entities. Categorical location descriptors such as "major airports" and "arts district" are also not named entities.
Demonyms such as "American", "Chinese" and "Russian" are not relevant named entities. Events like "Olympics" and "Fashion Week" are also not relevant named entities. Hurricanes and other natural disasters are not relevant named entities.\n

---

Here are some example span classifications for your reference. Please follow this format.\n
Examples:\n
1. Sentence: "The Prime Minister of {{Japan}} meets with world leaders to discuss economic cooperation."
Text Span: "Japan"
Label: (A). Correct Entity Annotation.\n
2. Sentence: "The European Union imposes sanctions on {{Russian}} officials over human rights abuses."
Text Span: "Russian"
Label: (C). Wrong Type. The correct entity type is other.\n
3. Sentence: "German Chancellor Angela Merkel meets with {{French}} President Macron."
Text Span: "French"
Label: (C). Wrong Type. The correct entity type is other.\n
4. Sentence: "New restaurant to open in {{downtown area}}, bringing jobs and economic growth."
Text Span: "downtown area"
Label: (D). Not a Named Entity.\n
5. Sentence: "Celebrities flock to {{Paris Fashion Week}} for the latest trends."
Text Span: "Paris Fashion Week"
Label: (B). Wrong Boundary. The correct span boundary is "Paris".\n
---

Please classify the following 3 spans of text:\n
1. Sentence: "Gang-related violence on the rise in {{New York City}}."
Text Span: "New York City"\n
2. Sentence: "The {{Tokyo}} Marathon, one of the largest and most prestigious races in the world, has been postponed due to the ongoing pandemic."
Text Span: "Tokyo"\n
3. Sentence: "{{Detroit}} police arrest three suspects in connection with a series of robberies in the downtown area."
Text Span: "Detroit"\n

Organization

Here are 3 sentences from news articles. Your task is to classify whether the span of text is a named entity of type organization.
In particular, for each query with a span of text enclosed in double braces, please classify the enclosed span of text in context of the query into one of the following categories:
- (A). The span is a named entity of type organization
- (B). The span contains a named organization entity but the span boundary is not precise
- (C). The span is a named entity but the category is not organization
- (D). The span is not a named entity
If the label is (B), you should provide the correct span boundary.
If the label is (C), you should provide the correct entity type. Pick an entity type from the following list:
[person, location, other].\n

A named organization entity must be the name of an organization.
Generic organization categories such as "high school", "city council", "local community", "government", "company" are not named entities. Descriptive organization types such as "non-profit organization", "foreign government" and "global technology company" are also not named entities.
Demonyms such as "Chinese", "Russian", "Indian", "Australian" and "European" are not relevant named entities. Viruses such as "COVID-19" are also not relevant named entities. Products such as "iPhone", "Windows" and "Instagram" are not relevant named entities.\n

---

Here are some example span classifications for your reference. Please follow this format.\n
Examples:\n
1. Sentence: "Indian government bans {{TikTok}} and 58 other Chinese apps."
Text Span: "TikTok"
Label: (A). Correct Entity Annotation.\n
2. Sentence: "{{CEO}} of Tesla Elon Musk Denies Securities Fraud Allegations"
Text Span: "CEO"
Label: (D). Not a Named Entity.\n
3. Sentence: "{{Japanese}} automaker Toyota recalls millions of vehicles."
Text Span: "Japanese"
Label: (D). Not a Named Entity.\n
4. Sentence: "{{Russian}} cosmonauts and Chinese astronauts conduct joint space mission."
Text Span: "Russian"
Label: (D). Not a Named Entity.\n
5. Sentence: "Pfizer announces new vaccine efficacy data against {{COVID-19}} variants."
Text Span: "COVID-19"
Label: (D). Not a Named Entity.\n
6. Sentence: "Local {{non-profit organization}} provides free meals to homeless veterans."
Text Span: "non-profit organization"
Label: (D). Not a Named Entity.\n

Please classify the following 3 spans of text:\n
1. Sentence: "The pharmaceutical company Pfizer announces a breakthrough in the development of a potential {{HIV}} vaccine."
Text Span: "HIV"\n
2. Sentence: "Celebrities flock to {{Cape Town Fashion Week}} for the latest trends."
Text Span: "Cape Town Fashion Week"\n
3. Sentence: "{{The National Gallery of Art}} opens new exhibit featuring works by Monet and Renoir."
Text Span: "The National Gallery of Art"\n

F.3 Diversity Requirement Sample Values

In this section, we show a few of the generated diversity requirement configuration values for illustrative purposes in Tables 34, 35 and 36.

F.4 Generated NER Samples

In this section, we show the selected demo samples and a few generated NER samples from Simple Prompt and each diversity variant. We show the raw LLM-generated samples, which may not follow the diversity instructions and demo formatting. Note that the generated samples may contain wrong annotations. For diversity approaches, we additionally show the corresponding sampled configurations in the prompts.

Variant	Group	Sample Values
CoNLL-2003		
Diversify X	News Topic Writing Style	World News, Politics, Technology, ... Breaking news reports, Investigative journalism, Human interest stories, ...
Diversify Y (vanilla)	per. loc. org.	Joe Biden, Angela Merkel, Donald Trump, ... New York City, Paris, Tokyo, ... United Nations, World Health Organization, European Union, ...
Diversify Y (latent)	*World News* - per. *World News* - loc. *World News* - org.	Angela Merkel, Elon Musk, Megan Rapinoe, ... New York City, London, Sydney, ... United Nations, World Health Organization, Google, ...
	Politics - per. *Politics* - loc. *Politics* - org.	Joe Biden, Kamala Harris, Mitch McConnell, ... Washington, D.C., New York City, Beijing, ... United Nations, European Union, Democratic National Committee, ...
	Technology - per. *Technology* - loc. *Technology* - org.	Elon Musk, Tim Cook, Sundar Pichai, ... Silicon Valley, Shenzhen, China, Seoul, South Korea, ... Google, Apple, Microsoft, ...
	...	...
WikiGold		
Diversify X	Topic Writing Style	Geography, Sports, Literature, ... Academic, Persuasive, Explanatory, ...
Diversify Y (vanilla)	per. loc. org.	Barack Obama, Marilyn Monroe, Albert Einstein, ... New York City, Paris, Tokyo, ... United Nations, International Monetary Fund, World Health Organization, ...
Diversify Y (latent)	*Geography* - per. *Geography* - loc. *Geography* - org.	Ferdinand Magellan, Alexander von Humboldt, Marie Tharp, ... Mount Everest, Great Barrier Reef, Amazon Rainforest, ... National Geographic Society, National Park Service, World Wildlife Fund, ...
	Sports - per. *Sports* - loc. *Sports* - org.	Lionel Messi, Serena Williams, Michael Jordan, ... Madison Square Garden, Camp Nou, Lambeau Field, ... International Olympic Committee, Major League Baseball, National Football League, ...
	Literature - per. *Literature* - loc. *Literature* - org.	William Shakespeare, Jane Austen, F. Scott Fitzgerald, ... Stratford-upon-Avon, The Brontë Parsonage Museum, The Shakespeare's Globe, ... Poetry Foundation, PEN America, National Book Foundation, ...
	...	...

Table 34: Sample generated values for Diversity Requirement Configurations for CoNLL-2003 and WikiGold.

F.4.1 Demo Samples

CoNLL-2003

1. Sentence: "BASEBALL - DODGERS WIN FIFTH STRAIGHT."
Named Entities: [DODGERS (organization)]\n
2. Sentence: "With 2.5 percent gross domestic product growth expected for 1996, new job growth should slowly lower the unemployment rate over the rest of the year."
Named Entities: []\n
3. Sentence: "A.de Silva not out 49"
Named Entities: [A.de Silva (person)]\n
4. Sentence: "Saudi Arabia executes Pakistani man."
Named Entities: [Saudi Arabia (location)]\n

WikiGold

1. Sentence: 'His book, A Biblical Case for an Old Earth was described in a review by Law Professor David W. Opderbeck, in the American Scientific Affiliation's Perspectives on Science and Christian Faith as "succeed[ing] admirably" in "establish[ing] that the 'day-age' view is a valid alternative for Christians who hold to biblical inerrancy", but as "less persuasive" at "argu[ing] for a concordist understanding of the Genesis texts and modern science."'
Named Entities: [David W. Opderbeck (person), American Scientific Affiliation (organization)]
2. Sentence: "He scored one international goal, against Slovakia in 1994."
Named Entities: [Slovakia (location)]\n
3. Sentence: "It was built as a self-contained steamship dock facility."
Named Entities: []\n

MIT-Movie

1. Query: "what type of movie genre is the perfect weapon"
Named Entities: [the perfect weapon (Title)]\n
2. Query: "how many movies came out in 2004"
Named Entities: [2004 (Year)]\n
3. Query: "could you show me some part of the new indiana jones movie"
Named Entities: [some part (Trailer), indiana jones (Character)]\n
4. Query: "when did mark joffe direct the bounty hunter film that is rated pg"
Named Entities: [mark joffe (Director), bounty hunter (Plot), pg (MPAA Rating)]\n
5. Query: "is cary grant in any historical films that are a must see"
Named Entities: [cary grant (Actor), historical (Genre), must see (Viewers' Rating)]\n
6. Query: "show me a movie with the song a whole new world"
Named Entities: [a whole new world (Song)]\n
7. Query: "what movie is considered the funniest of all time"
Named Entities: [funniest of all time (Review)]\n

MIT-Restaurant

1. Query: "is there any place around here that has a good menu for happy hour"
Named Entities: [good (Rating), happy hour (Amenity)]\n
2. Query: "where can i get afghan cuisine before 9 a m"
Named Entities: [afghan (Cuisine), before 9 a m (Hours)]\n
3. Query: "call mcdonalds"
Named Entities: [mcdonalds (Restaurant Name)]\n
4. Query: "find us a place to eat for under 10 dollars a plate"
Named Entities: [under 10 dollars a plate (Price)]\n

Variant	Group	Sample Values
MIT-Movie		
Diversify X	Query Category	Plot summary and synopsis, Cast and crew information, Movie reviews and ratings, ...
	User Demographic	Teenage girl, Middle-aged man, Senior woman, ...
	User Emotion	Excited, Bored, Indifferent, ...
	Query Language	Vague, Formal, Indirect, ...
Diversify Y (vanilla)	Title	The Godfather, Citizen Kane, Pulp Fiction, ...
	View. Rat.	5 stars, 4.5 stars, A+, ...
	Year	1975, 2003, 1999, ...
	Genre	Action, Romance, Comedy, ...
	Director	Steven Spielberg, Martin Scorsese, Christopher Nolan, ...
	MPAA Rat.	G, PG, PG-13, ...
	Plot	Love triangle, Time travel, Revenge, ...
	Actor	Leonardo DiCaprio, Meryl Streep, Tom Hanks, ...
	Trailer	Teaser, Sneak peek, Preview, ...
	Song	Bohemian Rhapsody, Purple Haze, Thriller, ...
	Review	Gripping, Riveting, Spectacular, ...
	Char.	Harry Potter, Tony Stark, Darth Vader, ...
Diversify Y (latent)	*Plot summary and synopsis* - Title	The Lord of the Rings, The Shawshank Redemption, The Matrix, ...
	Plot summary and synopsis - View. Rat.	Gripping, Must see, Riveting, ...
	Plot summary and synopsis - Year	1980, 1995, 2001, ...
	Plot summary and synopsis - Genre	Action, Adventure, Animation, ...
	Plot summary and synopsis - Director	Steven Spielberg, Quentin Tarantino, Christopher Nolan, ...
	Plot summary and synopsis - MPAA Rat.	PG-13, R, G, ...
	Plot summary and synopsis - Plot	Love triangle, Murder mystery, Time travel, ...
	Plot summary and synopsis - Actor	Tom Hanks, Sandra Bullock, Leonardo DiCaprio, ...
	Plot summary and synopsis - Trailer	Teaser, Sneak peek, Preview, ...
	Plot summary and synopsis - Song	Purple Rain, Bohemian Rhapsody, Don't Want to Miss a Thing, ...
	Plot summary and synopsis - Review	Gripping storyline, Intriguing plot, Compelling narrative, ...
	Plot summary and synopsis - Char.	Harry Potter, Hermione Granger, Ron Weasley, ...
	Cast and crew information - Title	The Godfather, Pulp Fiction, Titanic, ...
	Cast and crew information - View. Rat.	Must-see, Excellent, Outstanding, ...
	Cast and crew information - Year	1995, 2000s, 1987, ...
	Cast and crew information - Genre	Action, Adventure, Animation, ...
	Cast and crew information - Director	Steven Spielberg, Martin Scorsese, Quentin Tarantino, ...
	Cast and crew information - MPAA Rat.	PG-13, R, G, ...
	Cast and crew information - Plot	Time travel, Artificial intelligence, Bank heist, ...
	Cast and crew information - Actor	Tom Hanks, Leonardo DiCaprio, Meryl Streep, ...
	Cast and crew information - Trailer	Teaser, Sneak peek, Preview, ...
	Cast and crew information - Song	Beyoncé, Elton John, Frank Sinatra, ...
	Cast and crew information - Review	Stellar performances, Chemistry between the actors, Supportive cast, ...
	Cast and crew information - Char.	Luke Skywalker, Harry Potter, Tony Stark, ...
	Movie reviews and ratings - Title	The Godfather, Avatar, The Shawshank Redemption, ...
	Movie reviews and ratings - View. Rat.	Must-see, Highly recommended, Excellent, ...
	Movie reviews and ratings - Year	2020, 1995, 1982, ...
	Movie reviews and ratings - Genre	Action, Adventure, Comedy, ...
	Movie reviews and ratings - Director	Steven Spielberg, Martin Scorsese, Quentin Tarantino, ...
	Movie reviews and ratings - MPAA Rat.	G, PG, PG-13, ...
	Movie reviews and ratings - Plot	Serial killer, Time travel, Alien invasion, ...
	Movie reviews and ratings - Actor	Tom Hanks, Meryl Streep, Leonardo DiCaprio, ...
	Movie reviews and ratings - Trailer	Teaser, Preview, Sneak peek, ...
	Movie reviews and ratings - Song	Moon River, My Heart Will Go On, Let It Go, ...
	Movie reviews and ratings - Review	Gripping, Captivating, Stunning, ...
	Movie reviews and ratings - Char.	Luke Skywalker, Harry Potter, James Bond, ...
	...	...

Table 35: Sample generated values for Diversity Requirement Configurations for MIT-Movie.

5. Query: "im thirsty take me to the closest slurpee"
Named Entities: [closest (Location), slurpee (Dish)]\n

F.4.2 Simple Prompt

CoNLL-2003

1. The European Union is considering imposing tariffs on American products.
Named Entities: [European Union (organization), American (location)]\n
2. Elon Musk's SpaceX successfully launches another batch of Starlink satellites.
Named Entities: [Elon Musk (person), SpaceX (organization)]\n
3. The protests in Hong Kong continue to escalate as tensions rise between the government and demonstrators.
Named Entities: [Hong Kong (location)]\n
...

WikiGold

1. The Eiffel Tower is a wrought iron lattice tower on the Champ de Mars in Paris, France.
Named Entities: [Eiffel Tower (location), Champ de Mars (location), Paris (location), France (location)]
2. John F. Kennedy International Airport is a major international airport in Queens, New York City.
Named Entities: [John F. Kennedy International Airport (location), Queens (location), New York City (location)]
3. The World Health Organization is a specialized agency of the United Nations responsible for international public health.
Named Entities: [World Health Organization (organization), United Nations (organization)]
...

MIT-Movie

1. Can you tell me the plot of the movie Inception?
Named Entities: [Inception (Title)]\n

Variant	Group	Sample Values
MIT-Restaurant		
Diversify X	Meal Category	fine dining, buffet, late-night dining, ...
	User Demographic	25-year-old female student, 40-year-old female accountant, 55-year-old male retiree, ...
	Ambiance	Casual, Outdoor seating, Family-friendly, ...
	Price Range	Affordable, Budget-friendly, Mid-range, ...
	Dietary Restriction	Vegetarian, Vegan, Gluten-free, ...
	Special Offering	Chef's specials, Private Dining, Tasting Menus, ...
	Service Mode	takeout, delivery, dine-in, ...
Diversify Y (vanilla)	Rest. Name	La Cucina Italiana, The Steakhouse, Sushi Park, ...
	Amenity	WiFi, Outdoor seating, Parking, ...
	Cuisine	Italian, Thai, Chinese, ...
	Dish	Spaghetti carbonara, Pad Thai, Sushi, ...
	Hours	Opening hours, Closing time, Operating hours, ...
	Location	Near me, Close by, Nearby, ...
	Price	McDonald's, Ruth's Chris Steak House, Olive Garden, ...
	Rating	Yelp, Google reviews, TripAdvisor, ...
Diversify Y (latent)	*fine dining* - Rest. Name	The French Laundry, Per Se, Alinea, ...
	fine dining - Amenity	Wine cellar, Private dining room, Outdoor terrace, ...
	fine dining - Cuisine	Italian, French, Japanese, ...
	fine dining - Dish	Filet mignon, Lobster bisque, Chicken parmesan, ...
	fine dining - Hours	Lunchtime, Dinner, Brunch, ...
	fine dining - Location	Near me, City, Downtown, ...
	fine dining - Price	Le Bernardin, Per Se, Eleven Madison Park, ...
	fine dining - Rating	Best, Top, Highest, ...
	buffet - Rest. Name	Chez Panisse, The French Laundry, Nobu, ...
	buffet - Amenity	WiFi availability, Parking options, Outdoor seating, ...
	buffet - Cuisine	Indian, Italian, Chinese, ...
	buffet - Dish	Sushi, Pizza, Tacos, ...
	buffet - Hours	Sunday Brunch, Breakfast Buffet, Lunch Buffet, ...
	buffet - Location	Nearby, Close, Local, ...
	buffet - Price	Budget-friendly, Affordable, Expensive, ...
	buffet - Rating	Best buffet in town, Top-rated buffet restaurants, Highly recommended buffets, ...
	late-night dining - Rest. Name	The Midnight Diner, After Hours Bistro, The Night Owl Tavern, ...
	late-night dining - Amenity	Outdoor seating, Private dining, Bar service, ...
	late-night dining - Cuisine	Italian, Mexican, Chinese, ...
	late-night dining - Dish	Pizza, Sushi, Burger, ...
	late-night dining - Hours	Late-night dining, Open late, Closing time, ...
	late-night dining - Location	Near me, Close by, Nearby, ...
	late-night dining - Price	Budget-friendly, Affordable, Cheap eats, ...
	late-night dining - Rating	Top rated, Highly recommended, 5-star, ...
	...	...

Table 36: Sample generated values for Diversity Requirement Configurations for MIT-Restaurant.

2. What is the viewers' rating for The Shawshank Redemption?
Named Entities: [The Shawshank Redemption (Title), viewers' rating (Viewers' Rating)]\n
3. Show me movies from the 1980s.
Named Entities: [1980s (Year)]\n
...

MIT-Restaurant

1. "Can you recommend a good sushi restaurant nearby?"
Named Entities: [sushi (Cuisine), nearby (Location)]\n
2. "What Italian restaurant has the best reviews?"
Named Entities: [Italian (Cuisine), best (Rating)]\n
3. "I want to go to a steakhouse for dinner tonight."
Named Entities: [steakhouse (Cuisine), tonight (Hours)]\n
...

F.4.3 Diversify X

CoNLL-2003

Prompt to ChatGPT

Additionally, the generated news article sentences should follow the requirements below:
1. The news should be about Education;
2. The writing style of the news should be human interest stories.

Example Response from ChatGPT

1. "Local high school student overcomes homelessness to graduate with honors."
Named Entities: [high school student (person)]\n
2. "Non-profit organization partners with school district to provide free tutoring services for low-income students."
Named Entities: [non-profit organization (organization), school district (organization)]\n
3. "Former teacher starts scholarship fund to help underprivileged students pursue higher education."
Named Entities: [teacher (person), scholarship fund (organization)]\n

WikiGold

Prompt to ChatGPT

Additionally, the generated Wikipedia article sentences should follow the requirements below:
1. The topic of the Wikipedia article sentences should be Mathematics;
2. The writing style of the Wikipedia article sentences should be biographical.

Example Response from ChatGPT

1. In 1931, mathematician Kurt Gödel published his famous incompleteness theorems, which revolutionized the study of mathematical logic.
Named Entities: [Kurt Gödel (person)]\n

2. As a pioneer in the field of algebraic geometry, David Mumford was awarded the Fields Medal in 1974 for his groundbreaking work on geometric invariant theory.
Named Entities: [David Mumford (person)]\n
3. The Clay Mathematics Institute offers a prize of one million dollars for a correct proof of the Riemann hypothesis, one of the most famous unsolved problems in mathematics.
Named Entities: [Clay Mathematics Institute (organization), Riemann (person)]\n

MIT-Movie
Prompt to ChatGPT
Additionally, the generated spoken movie queries should follow the requirements below:
1. The query should inquire about Movie showtimes and ticket availability;
2. The user should be adventurous.

Example Response from ChatGPT
1. "Can I buy tickets for the latest James Bond movie?"
Named Entities: [latest James Bond (Title)]\n
2. "Is there a sci-fi movie with a soundtrack by Hans Zimmer playing at the theater today?"
Named Entities: [sci-fi (Genre), Hans Zimmer (Song)]\n
3. "I'm looking for a movie directed by Christopher Nolan and rated R. Do you have any suggestions?"
Named Entities: [Christopher Nolan (Director), R (MPAA Rating)]\n

MIT-Restaurant
Prompt to ChatGPT
Additionally, the generated restaurant queries should follow the requirements below:
1. The meal category should be fine dining;
2. The restaurant's price range should be affordable.

Example Response from ChatGPT
Query: "can you recommend a fine dining restaurant with affordable prices"
Named Entities: [fine dining (Amenity), affordable prices (Price)]\n
Query: "what are the top-rated fine dining restaurants in the area"
Named Entities: [top-rated (Rating), fine dining (Amenity)]\n
Query: "where can I find a fine dining restaurant with budget-friendly options"
Named Entities: [fine dining (Amenity), budget-friendly options (Price)]\n

F.4.4 Diversify Y (vanilla)

CoNLL-2003
Prompt to ChatGPT
Additionally, in the generated sentences, include the following terms: [United Nations Development Programme, Steven Spielberg].

Example Response from ChatGPT
1. "Steven Spielberg to direct new film about the United Nations Development Programme."
Named Entities: [Steven Spielberg (person), United Nations Development Programme (organization)]\n
2. "United Nations Development Programme launches new initiative to combat climate change in developing countries."
Named Entities: [United Nations Development Programme (organization)]\n
3. "Steven Spielberg's latest movie receives funding from the United Nations Development Programme."
Named Entities: [Steven Spielberg (person), United Nations Development Programme (organization)]\n

WikiGold
Prompt to ChatGPT
Additionally, in the generated sentences, include the following terms: [Alexander the Great, Angela Merkel].

Example Response from ChatGPT
1. "Alexander the Great is often considered one of the most successful military commanders in history for the remarkable achievements of his campaigns."
Named Entities: [Alexander the Great (person)]\n
2. "Angela Merkel, the former Chancellor of Germany, is known for her leadership during the European financial crisis and her role in the European Union."
Named Entities: [Angela Merkel (person), Germany (location), European Union (organization)]\n
3. "The Louvre, located in Paris, is the world's largest art museum and a historic monument in France."
Named Entities: [Louvre (location), Paris (location), France (location)]\n

MIT-Movie
Prompt to ChatGPT
Additionally, in the generated queries, include the following terms: [Zooey Deschanel, Cyberpunk, TV-Y, Take Me Out to the Ball Game, 1968].

Example Response from ChatGPT
1. "Can you tell me if Zooey Deschanel is starring in any new comedy films"
Named Entities: [Zooey Deschanel (Actor), comedy (Genre)]\n
2. "I'm looking for a cyberpunk movie released in 1968"
Named Entities: [cyberpunk (Genre), 1968 (Year)]\n
3. "Is there a family-friendly movie with a TV-Y rating featuring the song 'Take Me Out to the Ball Game'"
Named Entities: [family-friendly (Genre), TV-Y (MPAA Rating), Take Me Out to the Ball Game (Song)]\n

MIT-Restaurant
Prompt to ChatGPT
Additionally, in the generated queries, include the following terms: [Value meal, The Firehouse Grill].

Example Response from ChatGPT
1. "I'm hungry, can you recommend a restaurant with a good value meal?"
Named Entities: [good (Rating), Value meal (Amenity)]\n
2. "What's the address and hours of operation for The Firehouse Grill?"
Named Entities: [The Firehouse Grill (Restaurant Name), address (Location), hours of operation (Hours)]\n
3. "I'm craving Italian cuisine, do you know a restaurant nearby that serves it?"
Named Entities: [Italian (Cuisine), nearby (Location)]\n

F.4.5 Diversify Y (latent)

CoNLL-2003
Prompt to ChatGPT
Additionally, in the generated sentences, include the following terms: [Pandora Media, Sony Music Entertainment, Sony Music].

Example Response from ChatGPT

1. "Pandora Media announces partnership with Sony Music Entertainment."
Named Entities: [Pandora Media (organization), Sony Music Entertainment (organization)]\n
2. "Sony Music to release new album in collaboration with a popular artist."
Named Entities: [Sony Music (organization)]\n
3. "Pandora Media's CEO discusses future plans for the company at annual conference."
Named Entities: [Pandora Media (organization)]\n

WikiGold

Prompt to ChatGPT
Additionally, in the generated sentences, include the following terms: [Doctors Without Borders, Centers for Disease Control and Prevention].

Example Response from ChatGPT
1. Doctors Without Borders provides medical aid to over 70 countries around the world.
Named Entities: [Doctors Without Borders (organization)]\n
2. The Centers for Disease Control and Prevention is responsible for monitoring and preventing the spread of infectious diseases in the United States.
Named Entities: [Centers for Disease Control and Prevention (organization)]\n
3. In 2018, the organization partnered with the local government to provide healthcare services in rural areas.
Named Entities: []\n

MIT-Movie

Prompt to ChatGPT
Additionally, in the generated queries, include the following terms: [Taylor Swift, X, Jessica Chastain, 2011, Patty Jenkins, Sam Mendes, Standout performances].

Example Response from ChatGPT
1. "What 2011 movie directed by Patty Jenkins received standout performances by Jessica Chastain?"
Named Entities: [2011 (Year), Patty Jenkins (Director), Jessica Chastain (Actor), Standout performances]\n
2. "Could you recommend a film with a Taylor Swift song in the trailer?"
Named Entities: [Taylor Swift (Song), Trailer]\n
3. "Show me a Sam Mendes movie with a suspenseful plot."
Named Entities: [Sam Mendes (Director), suspenseful (Plot)]\n

MIT-Restaurant

Prompt to ChatGPT
Additionally, in the generated queries, include the following terms: [Cheeseburger, Mexican, Discounted items].

Example Response from ChatGPT
1. "Can you recommend a place that serves a good cheeseburger?"
Named Entities: [cheeseburger (Dish)]\n
2. "I'm craving Mexican food, where can I find a Mexican restaurant nearby?"
Named Entities: [Mexican (Cuisine), nearby (Location)]\n
3. "Do you know of any restaurants that offer discounted items during happy hour?"
Named Entities: [discounted items (Amenity), happy hour (Amenity)]\n

F.4.6 Diversify X+Y

CoNLL-2003

Prompt to ChatGPT
Additionally, the generated news article sentences should follow the requirements below:
1. The generated news article sentences should be about Art;
2. The generated news article sentences should include the following terms: [Colosseum, Rome, Public Art Fund, Art Institute of Chicago].

Example Response from ChatGPT
1. The Colosseum in Rome to host a new public art exhibition.
Named Entities: [Colosseum (location), Rome (location), Public Art Fund (organization)]\n
2. Art Institute of Chicago unveils new digital art collection.
Named Entities: [Art Institute of Chicago (organization)]\n
3. Renowned artist to create a new mural for the Public Art Fund in Rome.
Named Entities: [Public Art Fund (organization), Rome (location)]\n

WikiGold

Prompt to ChatGPT
Additionally, the generated Wikipedia article sentences should follow the requirements below:
1. The topic of the Wikipedia article sentences should be Music;
2. The generated Wikipedia article sentences should include the following terms: [Bob Dylan, Rock and Roll Hall of Fame, Berklee College of Music].

Example Response from ChatGPT
1. Her rendition of "Like a Rolling Stone" by Bob Dylan earned her a standing ovation from the audience.
Named Entities: [Bob Dylan (person)]\n
2. The band was inducted into the Rock and Roll Hall of Fame in 2006, in recognition of their influence on the music industry.
Named Entities: [Rock and Roll Hall of Fame (organization)]\n
3. Many successful musicians and artists, including Quincy Jones and Melissa Etheridge, have attended the prestigious Berklee College of Music.
Named Entities: [Quincy Jones (person), Melissa Etheridge (person), Berklee College of Music (organization)]\n

MIT-Movie

Prompt to ChatGPT
Additionally, the generated spoken movie queries should follow the requirements below:
1. The query should inquire about Soundtrack and music information;
2. The user should be ambivalent;
3. The generated spoken movie queries should include the following terms: [Thrilling musical arrangements, EDM, Rock and roll legend, Captivating theme music, I Wanna Dance with Somebody, Ska].

Example Response from ChatGPT
1. "I'm looking for some thrilling musical arrangements in a movie. Can you recommend anything with EDM or rock and roll legend in the soundtrack?"
Named Entities: [thrilling musical arrangements (Song)]\n
2. "I'm in the mood for some captivating theme music. Do you know any movies featuring 'I Wanna Dance with Somebody' or Ska?"
Named Entities: [captivating theme music (Song), I Wanna Dance with Somebody (Song), Ska (Genre)]\n

3. "I'm not sure what I'm in the mood for. Can you show me a movie with an awesome EDM soundtrack?"
Named Entities: [EDM (Song)]\n

MIT-Restaurant

Prompt to ChatGPT

Additionally, the generated restaurant queries should follow the requirements below:
1. The meal category should be fine dining;
2. The restaurant's price range should be affordable;
3. The generated restaurant queries should include the following terms: [Near me, Valet parking, Blue Hill at Stone Barns, High-end].

Example Response from ChatGPT

Query: "Can you recommend a fine dining restaurant near me with valet parking?"
Named Entities: [fine dining (Amenity), near me (Location), valet parking (Amenity)]\n
Query: "I'm looking for an affordable high-end restaurant with valet parking, maybe something like Blue Hill at Stone Barns"
Named Entities: [affordable (Price), high-end (Price), valet parking (Amenity), Blue Hill at Stone Barns (Restaurant Name)]\n
Query: "Where can I find a fine dining restaurant around here with valet parking and a price range that's affordable?"
Named Entities: [fine dining (Amenity), around here (Location), valet parking (Amenity), affordable (Price)]\n