Major Entity Identification: A Generalizable Alternative to Coreference Resolution

Kawshik Manikantan[1], **Shubham Toshniwal**[2], **Makarand Tapaswi**[1], **Vineet Gandhi**[1]
[1]CVIT, IIIT Hyderabad [2]NVIDIA
kawshik.manikantan@research.iiit.ac.in, stoshniwal@nvidia.com, {makarand.tapaswi, vgandhi}@iiit.ac.in

Abstract

The limited generalization of coreference resolution (CR) models has been a major bottleneck in the task's broad application. Prior work has identified annotation differences, especially for mention detection, as one of the main reasons for the generalization gap and proposed using additional annotated target domain data. Rather than relying on this additional annotation, we propose an alternative referential task, **M**ajor **E**ntity **I**dentification (MEI), where we: (a) assume the target entities to be specified in the input, and (b) limit the task to only the frequent entities. Through extensive experiments, we demonstrate that MEI models generalize well across domains on multiple datasets with supervised models and LLM-based few-shot prompting. Additionally, MEI fits the classification framework, which enables the use of robust and intuitive classification-based metrics. Finally, MEI is also of practical use as it allows a user to search for all mentions of a particular entity or a group of entities of interest. [1]

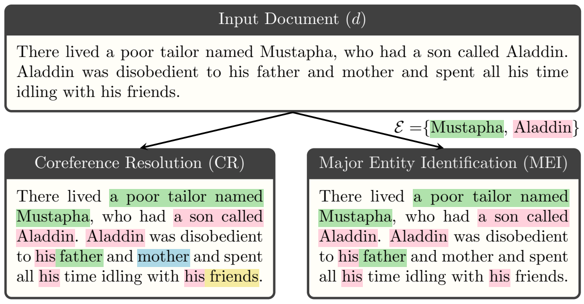

Figure 1: CR *vs*. MEI. The CR task aims to detect and cluster all mentions into different entities, shown in various colors. MEI takes major entities as additional input and aims to detect and classify the mentions that refer only to these entities.

1 Introduction

Coreference resolution (CR) is the task of finding text spans that refer to the same entity. CR is a fundamental language understanding task relevant to various downstream NLP applications, such as question-answering (Dhingra et al., 2018), building knowledge graphs (Koncel-Kedziorski et al., 2019), and summarization (Sharma et al., 2019). Despite the importance of CR and the progress made by neural coreference models (Dobrovolskii, 2021; Bohnet et al., 2023; Zhang et al., 2023), domain generalization remains an issue even with the best-performing supervised models (Xia and Van Durme, 2021; Toshniwal et al., 2021).

The lack of domain generalization in CR models can largely be attributed to differences in annotation guidelines of popular CR benchmarks, specifically annotation guidelines about what constitutes a mention (Porada et al., 2023). For example, OntoNotes (Pradhan et al., 2013) does not annotate singletons, confounding mention identity with being referential. Thus, models trained on OntoNotes generalize poorly (Toshniwal et al., 2021). The importance of mention detection for CR generalization is further highlighted by Gandhi et al. (2023), showing that solely annotating mentions is sufficient and more efficient for adapting pre-trained CR models to new domains (in comparison to annotating coreference chains). Similarly, GPT-4 struggles with zero-/few-shot mention prediction, but with ground-truth mentions, its CR performance (Le and Ritter, 2023) is competitive with that of supervised models (Toshniwal et al., 2021).

Given these observations, we hypothesize that current CR models, including large language models, generalize well at *mention clustering* but struggle to generalize on *mention detection* due to idiosyncrasies of different domains/benchmarks. We put forth an alternative referential task where the entities of interest are known and provided as additional input. Assuming entities to be part of the input offloads the required domain adaptation from training to inference. Specifically, we propose the

[1]Code for the paper is available at `https://github.com/KawshikManikantan/MEI`

Statistics	LitBank CR	LitBank MEI	FantasyCoref CR	FantasyCoref MEI
# of Mentions	29103	16985	56968	35938
# of Non singletons	23340	16985	56968	35938
Mean ant. dist.	55.31	36.95	57.58	30.24
# of Clusters	7927	490	5829	942
Avg. cluster size	3.67	34.66	9.77	38.15

Table 1: Comparing CR and MEI. MEI has fewer but larger clusters, and a smaller mean antecedent distance (Mean ant. dist.). Our formulation's frequency-based criterion for deciding major entities means that singleton mentions are typically not a part of MEI.

task of Major Entity Identification (MEI), where we assume the major entities of the narrative, to be provided as input along with the text (see Fig. 1). We focus on major entities for the following reasons: (a) Specifying major entities of a narrative is intuitively easier. (b) A handful of major entities often dominate any discourse. Table 1 shows that in FantasyCoref roughly 16% of entities (942 of 5829) contribute to 63% of the mentions (35938 of 56968).

In this work, we adapt two literary CR benchmarks, namely LitBank (Bamman et al., 2020) and FantasyCoref (Han et al., 2021) by identifying frequently occurring entities as major entities and customizing a state-of-the-art coreference model (Toshniwal et al., 2021) to MEI. Our tests for generalizability reveal that while there is a big gap in CR performance between in- and out-of-domain models (Toshniwal et al., 2021), this performance gap is much smaller for MEI (Section 5.1). To test this hypothesis further, we evaluate large language models (LLMs) for MEI in a few-shot learning setup. On CR, LLMs are shown to struggle with mention detection and perform worse than supervised models (Le and Ritter, 2023). Contrary to this, on MEI, top LLMs (*e.g.* GPT-4) are only slightly behind supervised models (Section 5.2). These experiments in the supervised setting and the few-shot setting demonstrate that the MEI task is more generalizable than CR.

Additionally, we argue that MEI is easier to evaluate than CR. The MEI task can be viewed as a classification task in which any text span either refers to one of the input entities or the null class (*minor* entities and other non-mention spans). The classification metrics maintain consistent granularity, proportionally penalize perturbations, and exhibit high discriminatory power while intuitively meeting multiple desired specifications (Moosavi and Strube, 2016; Recasens and Hovy, 2011).

Furthermore, MEI, by its definition, disregards insignificant and smaller clusters known to inflate the CR metrics (Moosavi and Strube, 2016; Lu and Ng, 2020; Kummerfeld and Klein, 2013). As an aside, formulating MEI as a classification task allows for a trivial parallelization across candidate spans (Appendix A.1).

Finally, MEI's explicit mapping of mentions to predefined entities improves its usability over CR in downstream applications that focus on mentions of specific entities. MEI effectively replaces tailored heuristics employed to extract CR cluster(s) referring to entities of choice in such applications (entity understanding (Inoue et al., 2022), sentiment and social dynamics analysis (Zahiri and Choi, 2017; Antoniak et al., 2023)).

2 Task Formulation

Notation. For a document d, let $\mathcal{E} = \{e_j\}_{j=1}^{L}$ be the set of L major entities that we wish to identify. We define $\mathcal{M}_{\text{all}}$ as the set of all mentions that could refer to any entity and subsequently $\mathcal{M}_j \subseteq \mathcal{M}_{\text{all}}$ as the set of mentions that refer to a major entity e_j. Furthermore, we denote $\mathcal{M} = \bigcup_j \mathcal{M}_j$ as the set of mentions that refer to one of the major entities while mentions that do not correspond to any major entity are designated as $\mathcal{M}_{\text{other}} = \mathcal{M}_{\text{all}} \setminus \mathcal{M}$.

Task formulation. In MEI, the input consists of the document d and designative phrases $\mathcal{P} = \{p(e_j)\}_{j=1}^{L}$ where $p(e_j)$ succinctly represents the entity e_j. For example, in Fig. 1, the phrases *"Aladdin"* and *"Mustapha"* uniquely represent Aladdin and his father who appear in *"Aladdin And The Wonderful Lamp"*. Note that in CR, the designative phrases $\mathcal{P}$ are not part of the input.

In contrast to CR's clustering foundations, MEI starts with a prior for each entity (the designative phrase) and can be formulated as an open set classification, where every mention is either classified as one of the major entities or ignored. Formally, MEI aims to assign each mention $m \in \mathcal{M}_j$ to e_j and mentions $m \in \mathcal{M}_{\text{other}}$ to $\varnothing$, a null entity.

3 Supervised MEI models

We propose MEIRa, **M**ajor **E**ntity **I**dentification via **Ra**nking, which draws inspiration from the entity ranking formulation (Xia et al., 2021; Toshniwal et al., 2020) and maintains an explicit representation for entities. The MEIRa models consist of 3 steps: encoding the document, proposing candidate mentions, and an identification (id) module that tags mentions with major entities or the null entity.

Document encoding is performed using a Longformer-Large (Beltagy et al., 2020), ϕ, that we finetune for the task. Mentions (or spans) are encoded as $\mathbf{m}_i = \phi(m_i, d)$ by concatenating the first, last, and an attention-weighted average of the token representations within the mention span. In MEI, an additional input is the set of designative phrases $\mathcal{P}$ for the major entities. Since each phrase is derived from the document itself, we also obtain its encoding using the backbone: $\mathbf{e}_j = \phi(p(e_j), d)$.

Mention detection. Similar to prior efforts (Toshniwal et al., 2021), we use a mention proposal network that predicts high-scoring candidate mentions. This step finds all mentions $\mathcal{M}_{\text{all}}$ and not just the ones corresponding to the major entities $\mathcal{M}$.Training a model to only detect mentions of major entities would confuse it leading to poor performance.

Identification module. As illustrated in Fig. 2, we initialize a working memory $\mathcal{E}^W = [\mathbf{e}_j]_{j=1}^L$ as a list of L major entities based on their designative phrase representations. Given a mention m_i, the id module computes the most likely entity as:

$$[s_i^*, e_i^*] = \max_{j=1...L} f([\mathbf{m}_i, \mathbf{e}_j, \chi(m_i, e_j)])\,, \quad (1)$$

where $f()$ is an MLP that predicts the score of tagging mention m_i with the entity e_j, and $\chi(m_i, e_j)$ encodes metadata. The output s_i^* corresponds to the highest score and e_i^* is the top-scoring entity. Based on the score, m_i is assigned to:

$$y(m_i) = \begin{cases} e_i^* & \text{if } s_i^* > \tau\,, \\ \varnothing & \text{otherwise}\,, \end{cases} \quad (2)$$

where τ is a threshold (set to 0 in practice).

The metadata $\chi(m_i, e_j)$ contains a distance (position) embedding representing the log distance between the mention m_i and the last tagged instance of the entity e_j. If no mention is yet associated with the entity, we use a special learnable embedding.

Updates to the working memory. We investigate two approaches:

(i) **MEIRa-S**tatic: As the name suggests, the working memory $\mathcal{E}^W$ of the entity representations remains constant ($\mathcal{E}^{W(0)}$) and is not updated with new mention associations. This makes the approach highly parallelizable.

(ii) **MEIRa-H**ybrid: Similar to traditional CR, this variation maintains a dynamic working memory $\mathcal{E}^W$, which is updated with every new mention-id association. Specifically, assuming m_i is assigned to e_j^*, the working memory would be updated using a weighted mean operator g as $\mathbf{e}_j \leftarrow g(\mathbf{e}_j, \mathbf{m}_i)$, similar to Toshniwal et al. (2020). To prevent error accumulation, we evaluate the mentions against $\mathcal{E}^W$ and the initial entity representations ($\mathcal{E}^{W(0)}$), then compute the average score. This hybrid approach reaps benefits from both, the initial clean designative phrases and the dynamic updates.

Following Toshniwal et al. (2020), the mention detection and identification modules are trained end-to-end using separate cross-entropy loss functions.

4 Few-shot MEI with LLMs

We propose a prompting strategy to leverage LLMs for MEI, addressing their challenges in CR.

Mention detection challenges. CR or MEI can be addressed using separate few-shot prompting strategies for mention detection and mention clustering/identification. However, Le and Ritter (2023) found that this strategy faced significant challenges with mention detection, performing worse than a deterministic mention detector. Thus, they assume access to an oracle mention detector and focus on evaluating LLMs' linking capabilities.

An alternative is to use an external supervised mention detector instead of the oracle. However, this requires annotated training data and may not align with a true few-shot LLM prompt paradigm. Additionally, supervised mention detectors often fail to generalize across CR datasets due to annotation variability (Lu and Ng, 2020).

MEI with LLMs. We demonstrate that transitioning from CR to MEI addresses this gap in mention detection and proposes an end-to-end, few-shot prompting approach for MEI. Inspired by Dobrovolskii (2021), we develop a prompting strategy that first performs MEI at word-level (rather than span), followed by a prompt to retrieve the span corresponding to the word.

In addition to the document d and the set of phrases $\mathcal{P}$, we also provide entity identifiers (*e.g.* #1, #2) to the LLM. We will use the following example:
Document: *That lady in the BMW is Alice's mom.*
Major Entities: 1. *Alice*; 2. *Alice's mother*.

Prompt 1. Word-level MEI. Mention detection with LLMs is challenging due to the frequent occurrence of nested mentions. We overcome this by prompting the LLM to tag each word. Specifically, through few-shot examples, we ask the LLM

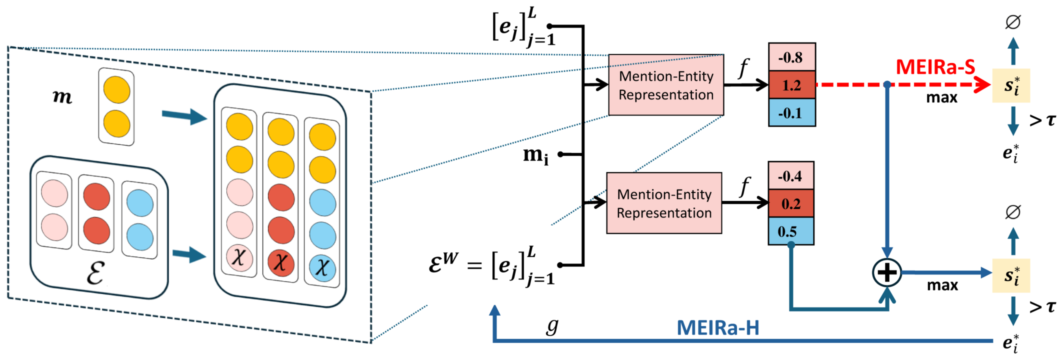

Figure 2: Identification module of MEIRa. A mention encoding $\mathbf{m}_i$ is concatenated with each entity's embedding in $\mathcal{E}^W$ and the metadata $\chi(m_i, e_j)$. Network f scores the likelihood of assigning m_i to each major entity. If the highest score s_i^* is above the threshold τ, m_i is associated with the highest scoring major entity e_i^* or discarded. In MEIRa-S, the entity memory $\mathcal{E}^W$ remains static. For MEIRa-H (blue path), the assigned entity's working memory is updated, and both the static (top half) and updated working memory (bottom half) are utilized to compute a final score.

to detect and tag the **syntactic heads**[2] (e.g., *lady*, *Alice*, *mom*) of mentions that refer to the major entities. Other words are left untagged (implicitly assigned to $\varnothing$, the null entity). To create the few-shot examples, a contiguous set of words annotated with the same entity is considered as a span and its syntactic head is extracted using `spaCy` (Honnibal et al., 2020).

The ideal output for the example above is:

"That lady#2 in the BMW is Alice#1's mom#2..".

Note that, even though the span *"BMW"* might be a valid mention, it is not annotated as it does not refer to one of the major entities. The exact prompt used for this is provided in the Appendix, Table 9.

Prompt 2. Head2Span retrieval. The entity tagged heads are passed to the Head2Span (H2S) module, along with the document to retrieve the span. The prompt consists of the document pre-annotated with the positions of the head, where each candidate head-word is followed by a "#" and is instructed to be replaced by the complete span (including any existent determiners and adjectives). For the input:

That lady# in the BMW is Alice#'s mom#.

the expected ideal output is

That lady (That lady in the BMW) in the BMW is Alice(Alice's)'s mom (Alice's mom).

Table 10 in the appendix shows the H2S prompt.

Preserving structure. We pose MEI as a structured generation task, prompting LLMs to reproduce documents and generate MEI tags at specific locations. Proprietary models like GPT-4 generally reproduce documents faithfully but for rare failures, we use the Needleman-Wunsch algorithm (Needleman and Wunsch, 1970) to align documents and extract tags In the case of open-source models, we employ regular expression-based constrained decoding with the `outlines` library (Willard and Louf, 2023)

5 Experiments

Datasets. We evaluate three literary datasets chosen for their longer length and identifiable major entities, particularly the key narrative elements such as characters or plot devices. Table 1 compares statistical aspects of MEI and CR, revealing that MEI features fewer clusters (entities) but larger cluster sizes (more mentions per cluster).

(i) *LitBank* (Bamman et al., 2020) annotates coreference in 100 literary texts, each averaging around 2000 words. Following prior work (Toshniwal et al., 2021), we utilize the initial cross-validation split, dividing the documents into training, validation, and test sets with an 80:10:10 ratio.

(ii) *FantasyCoref* (Han et al., 2021) provides OntoNotes (Pradhan et al., 2013)-style[3] coreference annotations for 211 documents from Grimm's Fairy Tales, with an average length of approximately 1700 words. The dataset includes 171 training, 20 validation, and 20 test documents.

(iii) *Additional Fantasy Text (AFT)* (Han et al., 2021) provides annotations for long narratives: (a) Aladdin (6976 words), (b) Ali Baba and the Forty Thieves (6911 words), and (c) Alice in Wonderland (13471 words).

Metrics. In contrast to CR, MEI facilitates the use of simple classification metrics. We define standard

[2]A syntactic head of a phrase is a word (*lady*) that is central to the characteristics of the phrase (*The lady in the BMW*).

[3]The exact guidelines are documented here

Model	FantasyCoref Macro-F1	FantasyCoref Micro-F1	LitBank Macro-F1	LitBank Micro-F1
Coref-ID	72.5±2.2	78.8±2.7	79.7±2.7	80.6±3.7
Coref-CM	77.7±1.8	82.4±2.2	74.1±2.5	76.0±3.0
Coref-FM	77.9±1.7	83.2±2.2	77.4±2.3	80.6±4.7
MEIRa-S	**80.7±0.6**	**84.9±0.5**	80.8±0.8	81.8±1.0
MEIRa-H	80.3±1.4	84.3±2.0	**82.3±1.2**	**83.2±2.5**

Table 2: Results for models trained jointly on FantasyCoref and LitBank.

Model	FantasyCoref Macro-F1	FantasyCoref Micro-F1	LitBank Macro-F1	LitBank Micro-F1
Coref-ID	63.4±1.8	69.5±3.6	58.0±2.4	57.7±1.0
Coref-CM	72.8±0.3	76.5±0.5	61.0±5.9	61.2±5.2
Coref-FM	71.2±1.5	75.2±1.3	66.1±2.1	67.1±3.9
MEIRa-S	**75.7±1.5**	78.5±1.2	74.6±1.1	74.7±1.6
MEIRa-H	74.7±1.0	**78.5±0.8**	**77.2±1.9**	**78.6±2.7**

Table 3: Results for models trained on OntoNotes.

precision and recall for each major entity considered as an individual class of its own.

For a dataset $\mathcal{D} = \{d_1, \ldots, d_{|\mathcal{D}|}\}$, the evaluation metrics are defined as follows:

$$\text{Macro-F1} = \frac{\sum_{d\in\mathcal{D}} \sum_{e_j\in\mathcal{E}_d} F1(e_j)}{\sum_{d\in\mathcal{D}} |\mathcal{E}_d|}, \text{ and} \quad (3)$$

$$\text{Micro-F1} = \frac{1}{|\mathcal{D}|} \sum_{d\in\mathcal{D}} \frac{\sum_{e\in\mathcal{E}} F1(e_j)\cdot|\mathcal{M}_j|}{\sum_{e\in\mathcal{E}} |\mathcal{M}_j|}. \quad (4)$$

Macro-F1 is the average F1-score of entities across the dataset, while Micro-F1 is the frequency-weighted F1-score of entities within a document, averaged across the dataset.

Major entity selection. We select as major entities, the top-k entities ranked as per the frequency of occurrences. We use k=5 for LitBank and FantasyCoref after visualizing the frequency plots of their training sets. For longer documents in AFT, we select up to 9 entities to ensure coverage of all key entities from the story. We also enforce that every entity $e_j \in \mathcal{E}$ has a mention count $|\mathcal{M}_j| \geq 5$. We derive the representative span for each selected e_j from the set of mentions $\mathcal{M}_j$ by selecting the most commonly occurring name or nominal mention.

Implementation details.
Supervised models: Model hyperparameters are derived from Toshniwal et al. (2021). To ensure consistent performance across different numbers of target entities, we randomly select a subset of major entities at each training iteration (more details in Appendix A.2). Supervised models were trained five times with random seeds, and we present aggregated results as the mean and standard deviation.
LLMs: We follow a few-shot prompting mechanism across the setups and experiments. Prompts that perform referential tasks consist of 3 examples of 6 sentences each. These 3 examples contain a mixture of narrative styles (narratives, dialogues), types of entities (major, non-major entities), categories of mentions (names, nominals, pronouns), and plurality. Additionally, before producing the MEI output, we ask the LLM to describe each major entity briefly. We find that this additional step improves performance. For the H2S prompt, we provide 9 sentences as examples, balancing the number of pre- and post-modifiers to the head. All examples were selected from LitBank's train set and kept constant throughout the experiments. We set the temperature to 0 for all the models to ensure consistent and reproducible outputs.

5.1 Experiments: Supervised Models

Baselines. We train the `longdoc` model (Toshniwal et al., 2021) for CR and perform the following three inference-time adaptations for MEI:

Coref-ID: `longdoc` uses active lists of entity representations, resolving coreference by associating mentions with existing clusters or generating new ones. During inference, we disable the cluster creation step and pre-fill the entity list with the encoded vector representations of the major entities. Hence, all the detected mentions either get mapped to one of the major entities or are discarded.

Coref-Cosine Map (Coref-CM): Since CR clusters obtained from `longdoc` lack explicit entity association, we employ the Kuhn-Munkres (KM) algorithm (Munkres, 1957) to find the optimal matching cluster for each major entity. The cost matrix uses the cosine similarity between the encoded representation of the major entities and the predicted cluster embeddings, both derived from `longdoc`.

Coref-Fuzzy Map (Coref-FM): This method uses the KM algorithm to derive optimal mappings by constructing a cost matrix from accumulated fuzzy-string matching scores between designative phrases and the predicted cluster's mention strings.

Supervised results. In this experiment, we train MEIRa and the baseline models on the joint training set of LitBank and FantasyCoref. Subsequently, we assess their performance on the individual test sets, with results summarized in Table 2. Overall, MEIRa models consistently outperform the baselines on both metrics while also exhibiting better

Model	AFT Macro-F1	AFT Micro-F1
Coref-ID	68.1±5.9	78.7±6.1
Coref-CM	71.1±2.8	82.4±4.2
Coref-FM	71.1±4.7	83.2±4.7
MEIRa-S	81.6±1.4	88.8±1.3
MEIRa-H	**82.8±1.1**	**89.5±1.0**

Table 4: Results on the AFT dataset.

stability with a lower variance. The considerable variance observed in the performance of baseline methods across all experiments underscores the non-trivial nature of identifying clusters corresponding to major entities within the output clusters provided by the CR algorithms. MEIRa-H and MEIRa-S exhibit competitive parity on FantasyCoref (children stories), while MEIRa-H edges out on LitBank dataset, showcasing its adaptability in elaborate sentence constructions.

Generalization across datasets. To evaluate the generalization capabilities of MEIRa and baseline models, we train them on the OntoNotes dataset and then test their performance on LitBank and FantasyCoref. The results are presented in Table 3. When compared with Table 2, we observe a significant performance drop across the baseline models (*e.g.* for Coref-ID, the average Micro-F1 scores drop from 80.6 to 57.7 on LitBank). The performance gap for the baseline models is more pronounced on LitBank than on FantasyCoref because LitBank's annotation strategies differ more significantly from those of OntoNotes. The observations aligns with previous work (Toshniwal et al., 2021), that showcase poor generalization of models trained for CR. In contrast, MEIRa models recover most of the underlying performance on both the datasets (MEIRa-H drops a little from 83.2 to 78.6 on LitBank Micro-F1), demonstrating MEI as a more adaptable task, bringing robustness over varying annotation strategies.

Long documents. Table 4 presents results on the AFT dataset of the models trained using a combined training set of LitBank and FantasyCoref. MEIRa models significantly outperform the baseline models, with MEIRa-H gaining 11.7% in Macro-F1 over the best baseline. The results demonstrate the efficacy of MEIRa models on resolving key entities in longer narratives.

Computational performance. MEIRa-S supports parallel batched processing since it does not update the working memory after associating mentions, *i.e.* the mentions need not be processed sequentially from left to right. Hence, post-mention detection (common to all models), MEIRa-S is about 25× faster than `longdoc` when assessed across LitBank, FantasyCoref and AFT datasets on an NVIDIA RTX 4090 (see Fig. 3 in the appendix). Additionally, with the model's small memory footprint during inference, the entire process can also be parallelized across chunks of documents making it extremely efficient. Hence, we pose MEIRa-S as a faster while competitive alternative to MEIRa-H (that requires dynamic updates and has similar computational performance as `longdoc`).

Model	FantasyCoref Macro-F1	FantasyCoref Micro-F1	LitBank Macro-F1	LitBank Micro-F1
MEIRa-H	88.5	91.0	86.1	85.4
GPT-4	**90.7**	**92.0**	**88.8**	**91.6**
GPT-3.5	69.2	74.2	74.3	75.8
Code Llama-34B	67.0	72.4	68.9	73.1
Llama3-8B	53.8	60.6	50.2	53.4
Mistral-7B	67.3	75.8	61.6	73.9

Table 5: Few-shot LLM prompting results assuming the availability of ground-truth mentions.

5.2 Experiments: Few-shot prompting

Models. We experiment with GPT-4[4] (OpenAI, 2024), GPT-3.5[5], Code Llama-34B (Rozière et al., 2024), Mistral-7B (Jiang et al., 2023), and Llama3-8B.[6] Following Le and Ritter (2023), we use the instruction-tuned versions for open-source models. These models were chosen for their ability to handle the extended context required for our benchmarks.

5.2.1 Linking Performance w/ Gold Mentions

We first evaluate all the models assuming the availability of an oracle mention detector. The experimental configuration is aligned with that of Le and Ritter (2023), albeit with the distinction that we assess them for the MEI task rather than for CR. The prompt used in our setup is provided in Table 11 of Appendix. For comparison, we also perform inference on golden mentions with MEIRa-H.

The results in Table 5 show that GPT-4 surpasses the supervised MEIRa-H model in this setup. Among LLMs, GPT-4 is easily the best-performing model. Code Llama-34B performs the best among open-source models, closely followed by Mistral-

[4]Specifically, `gpt-4-1106-preview`
[5]Specifically, `gpt-3.5-turbo-0125`
[6]https://ai.meta.com/blog/meta-llama-3/

Model	FantasyCoref Macro-F1	FantasyCoref Micro-F1	LitBank Macro-F1	LitBank Micro-F1
MEIRa-H	**80.3**	**84.3**	82.3	83.2
GPT-4 w/ Ext det	80.1	82.2	78.7	83.9
GPT-4 with varying prompting strategies				
Single prompt	51.8	57.5	61.1	70.7
Two-stage prompt	70.5	74.9	76.5	81.3
Word-level MEI + spaCy H2S				
GPT-4	77.1	79.4	**82.5**	**85.5**
GPT-3.5	50.1	54.4	60.1	63.1
Code Llama-34B	30.0	31.4	22.7	23.2
Llama3-8B	29.2	32.1	20.5	26.0
Mistral-7B	19.4	21.9	12.9	14.0

Table 6: Results on LLMs with different mention detection and linking strategies.

Error Type	**MEIRa-H**	**GPT-4**
Missing Major	162	793
Major-Major	210	154
Major-Other	243	0
Other-Major	200	516
Extra-Major	461	896
Total	1276	2359

Table 7: Breakdown of errors by MEIRa-H and GPT-4 on the combined LitBank and FantasyCoref test set.

7B. While Code Llama-34B is tailored for the code domain, surprisingly, it outperforms strong LLMs suited for natural language. This result corroborates a similar finding by Le and Ritter (2023) for CR and related evidence regarding code pretraining aiding entity tracking (Kim et al., 2024). We find that Code Llama-34B performs close to GPT-3.5 for FantasyCoref, though a sizable gap persists in the Macro-F1 metric for LitBank , potentially due to its linguistic complexity.

5.2.2 MEI Task Performance with LLMs

In this section, we present the results for the end-to-end MEI task using LLMs. We compare all the approaches from Section 4 and relevant baselines with the results summarized in Table 6. To limit the combinations of LLMs and approaches for our experiments, we first compare all the approaches in tandem with GPT-4 and then present results for the best-performing approach with other LLMs.

The first straightforward approach of using a *Single Prompt* to retrieve all the mentions of major entities in a single pass results in a significant performance drop compared to MEIRa-H (prompt in Table 12 of Appendix). The reason is that while GPT-4 outperforms MEIRa-H at mention linking, its mention detection performance, especially with nested mentions, is much worse compared to MEIRa-H.[7]

To further underscore the importance of mention detection, we compare against *GPT-4 w/ Ext det*, which utilizes an external pre-trained mention detector followed by prompt-based linking (prompt in Table 11 of Appendix). We train the mention detector on the PreCo dataset (Chen et al., 2018), which achieves a 93.8% recall and 53.1% precision on the combined FantasyCoref and LitBank validation sets. *GPT-4 w/ Ext det* performs at par with the fully supervised MEIRa-H, again highlighting the strong mention linking capabilities of GPT-4.

Next, we present the results of our proposed *Two-stage prompt*, motivated by the *Single prompt* method's failure with nested mentions. The first prompt asks GPT-4 to perform word-level MEI, by limiting the task to syntactic heads only. The second prompt then performs the task of mapping the identified syntactic heads to full mention spans. The results strongly validate our proposed approach with a relative improvement of more than 10% over the *Single prompt* method across all metrics and datasets. We also explore replacing the second step, i.e., head-to-span (H2S) retrieval, with an external tool. Specifically, we invert spaCy's span-to-head mapping to obtain a head-to-span retriever.[8]

GPT-4 significantly improves in this setup, outperforming even the supervised model on LitBank. Given the strong performance of *GPT-4 + spaCy H2S*, we evaluate the open-source LLMs in only this setting. We observe a wide gap between GPT-4 and the open-source models. Llama3-8B outperforms other open-source models on both datasets in Micro-F1 and stays competitive with the larger Code Llama-34B in Macro-F1. However, this contrasts with Llama3-8B's significant lag in the idealized golden mention setting, which solely evaluates the model's linking capabilities.

5.3 Error Analysis

We classify MEI errors into five categories: (1) *Missing Major:* Not detecting a mention $m \in \mathcal{M}$. (2) *Major-Major:* Assigning a mention $m \in \mathcal{M}_j$ to any other major entity $\mathcal{E} \setminus e_j$. (3) *Major-Other:* Assigning a mention $m \in \mathcal{M}$ to $\varnothing$.

[7]The failure to detect nested mentions is despite best efforts to provide illustrative examples in the few-shot prompt. Le and Ritter (2023) report similar findings with earlier GPT versions.

[8]For the test set gold mentions of the two datasets, there were only two cases where spans had the same head. We handled these two cases manually.

Golden Mentions	Presently [a small boy] came walking along the path – [an urchin of nine or ten] ...... [Winterbourne] had immediately perceived that [he] might have the honor of claiming [him] as a fellow countryman. "Take care [you] don't hurt [your] teeth," [he] said, paternally ...... [My] mother counted them last night, and one came out right afterwards. She said she'd slap [me] if any more came out. [I] can't help it. It's this old Europe ...... If [you] eat three lumps of sugar, [your] mother will certainly slap [you]," [he] said. "She's got to give [me] some candy, then," rejoined [[his] young interlocutor].
GPT-4 Output	Presently [a small boy] came walking along the path – [an urchin of nine or ten] ...... [Winterbourne] had immediately perceived that [he] might have the honor of claiming [him] as a fellow countryman. "Take care you don't hurt your teeth," [he] said, paternally ...... [My] mother counted them last night, and one came out right afterwards. [She] said [she]'d slap [me] if any more came out. [I] can't help it. [It]'s this old Europe ...... If you eat three lumps of sugar, [your] mother will certainly slap [you]," [he] said. "[She]'s got to give [me] some candy, then," rejoined [his] young interlocutor.
MEIRa-H Output	Presently a small boy came walking along the path – [an urchin of nine or ten] ...... [Winterbourne] had immediately perceived that [he] might have the honor of claiming [him] as a fellow countryman. "Take care [you] don't hurt [your] teeth," [he] said, paternally ...... [My] mother counted them last night, and one came out right afterwards. She said she'd slap [me] if any more came out. [I] can't help it. It's this old Europe ...... If [you] eat three lumps of sugar, [your] mother will certainly slap [you]," [he] said. "She's got to give [me] some candy, then," rejoined [[his] young interlocutor].

Table 8: Qualitative Analysis showcasing different errors made by GPT-4 and MEIRa-H. Errors are color-coded as follows: Missing Major, Others-Major, Extra-Major, Major-Major, and Major-Other.

(4) *Other-Major:* Assigning a mention $m \in \mathcal{M}_{\text{other}}$ to any major entity in $\mathcal{E}$. (5) *Extra-Major:* Detecting extra mentions $m \notin \mathcal{M}_{\text{all}}$ and assigning to any major entity in $\mathcal{E}$.

Results combined over the LitBank and FantasyCoref test sets are presented in Table 7. Missing Major and Extra-Major contribute most of the errors for GPT-4, highlighting the scope for improvement in mention detection and span retrieval. Mention detection also remains a challenge in MEIRa-H, the model making most of the mistakes in the Extra-Major category. GPT-4 distinguishes major entities more clearly than MEIRa-H but tends to over-associate other mentions with major entities, resulting in higher Other-Major and Extra-Major errors. Note that GPT-4 has zero errors in the Major-Other category due to the prompt design, which only allows annotating major entities. Examples of these errors are visualized in Table 8.

6 Related Work

Neural models for CR have become the *de facto* choice in supervised settings (Lee et al., 2017; Kantor and Globerson, 2019; Joshi et al., 2020; Otmazgin et al., 2023). Efforts to enhance model efficiency include reducing candidate mentions to word-level spans (Dobrovolskii, 2021) and using single dense representations for entity clusters (Xia et al., 2021; Toshniwal et al., 2020).

Generalization in CR remains a lingering problem (Moosavi and Strube, 2017; Zhu et al., 2021; Porada et al., 2023). Current solutions include feature addition (Aralikatte et al., 2019; Otmazgin et al., 2023), joint training (Xia and Van Durme, 2021; Toshniwal et al., 2021), and active learning (Zhao and Ng, 2014; Yuan et al., 2022; Gandhi et al., 2023). Rather than relying on additional training data, we argue for an alternative formulation where the burden of domain adaptation is offloaded from training to inference.

LLM evaluation on referential tasks has largely been conducted in limited settings, such as the sentence-level Winograd Schema Challenges (WSC) (Brown et al., 2020), clinical pronoun resolution (Agrawal et al., 2022) and instance-level Q&A (Yang et al., 2022). Le and Ritter (2023) conducted the first document-level evaluation of LLMs for CR but assumed an oracle-mention detector. In contrast, we conduct end-to-end evaluations.

Entity-centric tasks similar to MEI include character identification, where either annotations are restricted to a subset of entities (Baruah and Narayanan, 2023) or custom models are developed to extract mentions of specific characters from TV show transcripts (Chen and Choi, 2016; Zahiri and Choi, 2017). We differ from these works by adopting a generalized task formulation independent of annotation strategies and entity selection. Another task, Entity Linking (Ji et al., 2015) extracts distinct entities from a document and links them to external Knowledge Bases. In contast, MEI focuses on retrieving all mentions (including nominals and pronominals) of a specific set of key entities, extracted solely from the context of the document.

7 Conclusion

CR models are limited in their generalization capabilities owing to annotation differences and general challenges of domain adaptation. We propose MEI as an alternative to CR, where the entities relevant to the input text are provided as input along with the text. Our experiments demonstrate that MEI is more suited for generalization than CR. Additionally, MEI can be viewed as a classification task that enables the use of intuitive metrics. A trivially parallelized variation (MEIRa-S), gives a 25x speedup over a comparable CR model, making it more suitable for longer narratives. Unlike CR, the formulation of MEI allows few-shot prompted LLMs to effectively compete with trained models. Our novel two-stage prompting and robust baseline methods empower top-performing LLMs like GPT-4 to achieve this. Our analysis indicates that this task holds promise for effectively evaluating the long-context referential capabilities of LLMs in an end-to-end manner.

Potential applications of MEI include domains such as film and literature, where metadata about salient entities can be sourced from external databases like IMDb or SparkNotes. Additionally, MEI can be applied to the analysis of documents like of financial and legal reports, when the user is familiar with the relevant entities. Lastly, recent research (Lin and Zeldes, 2024) indicates that LLMs can assist or automate the extraction of salient entities, a direction we intend to explore in future work.

8 Limitations

Major Entity Identification (MEI) is proposed as a generalizable alternative to the coreference resolution (CR) task, and is not a replacement of CR. MEI limits itself to major entities and only caters to applications that are interested in a particular pre-defined set of entities. Our experiments follow certain thresholds that might not be universally applicable, and results and performance might vary slightly along this decision (refer Appendix A.2). Our current few-shot prompting evaluations are limited only to a few models that accommodate a large context window. Optimizing prompts and architecture to allow for a piece-wise aggregation of outputs across chunks of documents is left for future work.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large Language Models are Few-Shot Clinical Information Extractors. In *EMNLP*.

Maria Antoniak, Anjalie Field, Jimin Mun, Melanie Walsh, Lauren Klein, and Maarten Sap. 2023. Riveter: Measuring Power and Social Dynamics Between Entities. In *ACL (Volume 3: System Demonstrations)*.

Rahul Aralikatte, Heather Lent, Ana Valeria Gonzalez, Daniel Herschcovich, Chen Qiu, Anders Sandholm, Michael Ringaard, and Anders Søgaard. 2019. Rewarding Coreference Resolvers for Being Consistent with World Knowledge. In *EMNLP-IJCNLP*.

David Bamman, Olivia Lewke, and Anya Mansoor. 2020. An Annotated Dataset of Coreference in English Literature. In *LREC*.

Sabyasachee Baruah and Shrikanth Narayanan. 2023. Character Coreference Resolution in Movie Screenplays. In *Findings of ACL*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv preprint arXiv:2004.05150*.

Bernd Bohnet, Chris Alberti, and Michael Collins. 2023. Coreference Resolution through a seq2seq Transition-Based System. *TACL*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *NeurIPS*.

Hong Chen, Zhenhua Fan, Hao Lu, Alan Yuille, and Shu Rong. 2018. PreCo: A Large-scale Dataset in Preschool Vocabulary for Coreference Resolution. In *EMNLP*.

Yu-Hsin Chen and Jinho D. Choi. 2016. Character Identification on Multiparty Conversation: Identifying Mentions of Characters in TV Shows. In *Annual Meeting of the Special Interest Group on Discourse and Dialogue*.

Bhuwan Dhingra, Qiao Jin, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. 2018. Neural Models for Reasoning over Multiple Mentions Using Coreference. In *NAACL-HLT*.

Vladimir Dobrovolskii. 2021. Word-Level Coreference Resolution. In *EMNLP*.

Nupoor Gandhi, Anjalie Field, and Emma Strubell. 2023. Annotating Mentions Alone Enables Efficient Domain Adaptation for Coreference Resolution. In *ACL*.

Sooyoun Han, Sumin Seo, Minji Kang, Jongin Kim, Nayoung Choi, Min Song, and Jinho D. Choi. 2021. FantasyCoref: Coreference Resolution on Fantasy Literature Through Omniscient Writer's Point of View. In *Fourth Workshop on Computational Models of Reference, Anaphora and Coreference*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Naoya Inoue, Charuta Pethe, Allen Kim, and Steven Skiena. 2022. Learning and Evaluating Character Representations in Novels. In *Findings of ACL*.

Heng Ji, Joel Nothman, Ben Hachey, and Radu Florian. 2015. Overview of tac-kbp2015 tri-lingual entity discovery and linking. *Theory and Applications of Categories*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *TACL*, 8.

Ben Kantor and Amir Globerson. 2019. Coreference Resolution with Entity Equalization. In *ACL*.

Najoung Kim, Sebastian Schuster, and Shubham Toshniwal. 2024. Code Pretraining Improves Entity Tracking Abilities of Language Models.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. Text Generation from Knowledge Graphs with Graph Transformers. In *NAACL-HLT*.

Jonathan K. Kummerfeld and Dan Klein. 2013. Error-Driven Analysis of Challenges in Coreference Resolution. In *EMNLP*.

Nghia T. Le and Alan Ritter. 2023. Are Large Language Models Robust Coreference Resolvers?

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end Neural Coreference Resolution. In *EMNLP*.

Jessica Lin and Amir Zeldes. 2024. GUMsley: Evaluating entity salience in summarization for 12 English genres. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2575–2588, St. Julian's, Malta. Association for Computational Linguistics.

Jing Lu and Vincent Ng. 2020. Conundrums in Entity Coreference Resolution: Making Sense of the State of the Art. In *EMNLP*.

Nafise Sadat Moosavi and Michael Strube. 2016. Which Coreference Evaluation Metric Do You Trust? A Proposal for a Link-based Entity Aware Metric. In *ACL*.

Nafise Sadat Moosavi and Michael Strube. 2017. Lexical features in coreference resolution: To be used with caution. In *ACL*.

James Munkres. 1957. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*.

Saul B Needleman and Christian D Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*.

OpenAI. 2024. GPT-4 Technical Report.

Shon Otmazgin, Arie Cattan, and Yoav Goldberg. 2023. LingMess: Linguistically Informed Multi Expert Scorers for Coreference Resolution. In *EACL*.

Ian Porada, Alexandra Olteanu, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2023. Investigating Failures to Generalize for Coreference Resolution Models.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards Robust Linguistic Analysis using OntoNotes. In *CONLL*.

M. Recasens and E. Hovy. 2011. Blanc: Implementing the rand index for coreference evaluation. *Nat. Lang. Eng.*, 17(4):485–510.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2024. Code Llama: Open Foundation Models for Code.

Eva Sharma, Luyang Huang, Zhe Hu, and Lu Wang. 2019. An Entity-Driven Framework for Abstractive Summarization. In *EMNLP-IJCNLP*.

Shubham Toshniwal, Sam Wiseman, Allyson Ettinger, Karen Livescu, and Kevin Gimpel. 2020. Learning to Ignore: Long Document Coreference with Bounded Memory Neural Networks. In *EMNLP*.

Shubham Toshniwal, Patrick Xia, Sam Wiseman, Karen Livescu, and Kevin Gimpel. 2021. On Generalization in Coreference Resolution. In *Workshop on Computational Models of Reference, Anaphora and Coreference*.

Brandon T Willard and Rémi Louf. 2023. Efficient Guided Generation for LLMs. *arXiv preprint arXiv:2307.09702*.

Congying Xia, Wenpeng Yin, Yihao Feng, and Philip Yu. 2021. Incremental Few-shot Text Classification with Multi-round New Classes: Formulation, Dataset and System. In *NAACL-HLT*.

Patrick Xia and Benjamin Van Durme. 2021. Moving on from OntoNotes: Coreference Resolution Model Transfer. In *EMNLP*.

Xiaohan Yang, Eduardo Peynetti, Vasco Meerman, and Chris Tanner. 2022. What GPT Knows About Who is Who. In *Workshop on Insights from Negative Results in NLP*.

Michelle Yuan, Patrick Xia, Chandler May, Benjamin Van Durme, and Jordan Boyd-Graber. 2022. Adapting Coreference Resolution Models through Active Learning. In *ACL*.

Sayyed M. Zahiri and Jinho D. Choi. 2017. Emotion Detection on TV Show Transcripts with Sequence-based Convolutional Neural Networks.

Wenzheng Zhang, Sam Wiseman, and Karl Stratos. 2023. Seq2seq is All You Need for Coreference Resolution. In *EMNLP*.

Shanheng Zhao and Hwee Tou Ng. 2014. Domain Adaptation with Active Learning for Coreference Resolution. In *Workshop on Health Text Mining and Information Analysis (Louhi)*.

Yilun Zhu, Sameer Pradhan, and Amir Zeldes. 2021. OntoGUM: Evaluating Contextualized SOTA Coreference Resolution on 12 More Genres. In *ACL-IJCNLP*.

A Appendix

A.1 Linking Speed Comparison

This section compares the computational performance of `longdoc` with the proposed MEIRa-S architecture. The classification formulation and the lack of an update step in MEIRa-S makes it a more efficient alternative to MEIRa-H and CR models. Fig. 3 displays the speed-up obtained in the identification module when assessed across documents with varying numbers of mentions. MEIRa-S consistently clocks a 20x efficiency across all ranges.

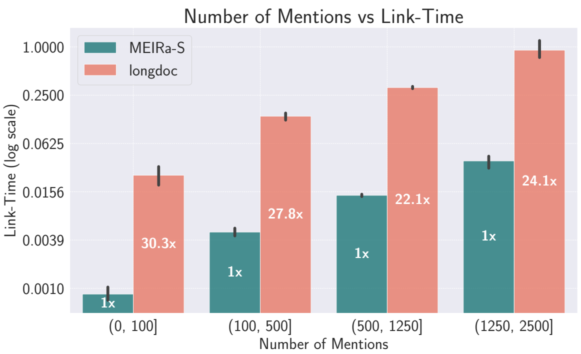

Figure 3: Linking speed comparison between MEIRa-S and `longdoc` for the combined LitBank and FantasyCoref test set. There exists 6 documents with (0, 100] mentions, 19 with (100, 500] mentions, 5 with (500, 1250] mentions and 3 with (1250, 2500] mentions.

A.2 Performance across number of entities

For consistency, the experiments of the main paper are evaluated across all the selected major entities (chosen using the thresholds defined in Section 5). A natural extension is to assess the model's performance with varying numbers of entities of choice. For instance, if one is interested in only two key characters, can these models maintain consistency when provided with their designative phrases?

In this section, we address this concern and evaluate the MEI models with varying numbers of input entities. We present the per-entity F1-score of all entities across the AFT dataset. The results for MEIRa-H are showcased in Fig. 4, Fig. 5 and Fig. 6. The first column of the heatmap shows the per-entity F1-score when it is the sole target entity in the document. For e.g., the value in the first column in Fig. 4 corresponding to the entity *Baba Mustapha* (0.93) indicates the performance of the model when *Baba Mustapha* is the only target entity.

As we move across the columns of a particular row (ignoring the first column), the column number indicates the number of target entities used at

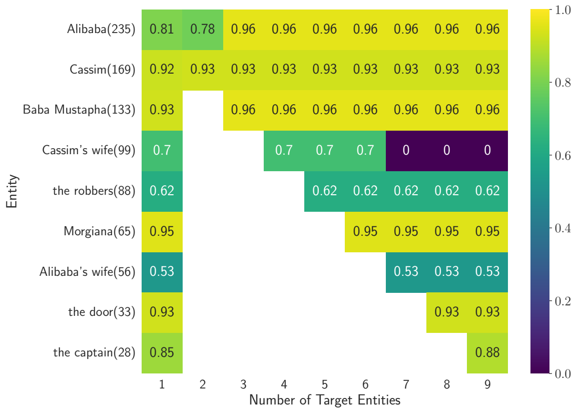

Figure 4: Performance of MEIRa-H across number of target entities for the document Ali Baba and the Forty Thieves.

Figure 5: Performance of MEIRa-H across number of target entities for Aladdin.

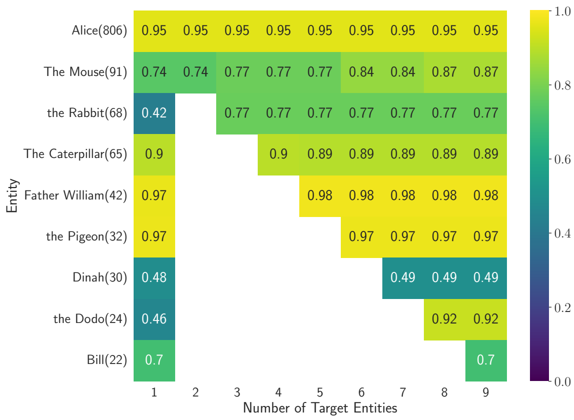

Figure 6: Performance of MEIRa-H across number of target entities for Alice in Wonderland.

inference. For instance, if the column number is k, the target entities are the top-k frequent entities. Again, the 4^{th} column in the row corresponding to *Baba Mustapha* indicates its individual F1-score in the experiment where the four input entities are *Alibaba*, *Cassim*, *Baba Mustapha* and *Cassim's* *wife*.

There are a few individual cases where the performance significantly varies with modifying the number of input entities. For example, *Cassim's wife* is confused with *Alibaba's wife* after the latter's introduction. However, overall, the per-entity F1-score remains consistent across varying numbers of input entities across all three documents. These results demonstrate the effectiveness of MEIRa-H for applications requiring variable numbers of target entities. This consistency is mainly due to the variable entity training, where a randomly chosen subset of major entities is selected in each iteration. Excluding this procedure leads to significant fluctuation in performance while modifying the number of target entities.

A.3 Prompts

We provide exact prompts for all the few-shot prompting experiments. Please note that not all the major entities listed in the few shot examples are necessary to be present in the text.

A.4 Budget and Hardware details

The supervised models were trained on a 24GB NVIDIA RTX 4090Ti GPU. For experiments with the open source language models, we used two 48GB NVIDIA RTX A6000 GPU's. For GPT-4 and GPT-3.5 experiments, we spent approximately 175$ in total, covering both initial explorations and the computation of final results.

Instruction

```
You will receive a Text along with a list of Key Entities and their corresponding Cluster IDs
as input. Your task is to perform Coreference Resolution on the provided text to categorize
"each word belonging to a cluster" with its respective cluster id. Also briefly describe the key
entities in 1-2 sentences before starting the coreference task.
Follow the format below to label a word with its cluster ID:
word#cluster_id
Please keep in mind:
- Ensure the output adheres to the specified format for easy parsing.
- Classify the words in the given text without altering any of the other content.
```

Example Input:

```
Key Entities:
1. Katharine Hilbery - #1
2. Mr. Denham - #2
3. Mrs. Hilbery - #3
4. Mr. Hilbery - #4
5. Mr. Fortescue - #5

Text:
CHAPTER I It was a Sunday evening in October , and in common with many other young ladies of
her class , Katharine Hilbery was pouring out tea . Perhaps a fifth part of her mind was thus
occupied , and the remaining parts leapt over the little barrier of day which interposed between
Monday morning and this rather subdued moment , and played with the things one does voluntarily
and normally in the daylight . But although she was silent , she was evidently mistress of a
situation which was familiar enough to her , and inclined to let it take its way for the six
hundredth time , perhaps , without bringing into play any of her unoccupied faculties . A single
glance was enough to show that Mrs. Hilbery was so rich in the gifts which make tea-parties of
elderly distinguished people successful , that she scarcely needed any help from her daughter
, provided that the tiresome business of teacups and bread and butter was discharged for her
. Considering that the little party had been seated round the tea-table for less than twenty
minutes , the animation observable on their faces , and the amount of sound they were producing
collectively , were very creditable to the hostess . It suddenly came into Katharine 's mind
that if someone opened the door at this moment he would think that they were enjoying themselves
; he would think , " What an extremely nice house to come into ! "
```

Example Output:

```
Description of Key Entities present in the text:
#1 - Katharine Hilbery: A young and apparently rich lady and the daughter of Mrs. Hilbery. She
and Mrs. Hilbery were organising a party for some distinguished elders.
#3 - Mrs. Hilbery: She is the mother of Katharine Hilbery and is a well-to-do member of the
society and a very efficient and able hostess

Coreference:
CHAPTER I It was a Sunday evening in October , and in common with many other young ladies of
her#1 class , Katharine#1 Hilbery#1 was pouring out tea . Perhaps a fifth part of her#1 mind was
thus occupied , and the remaining parts leapt over the little barrier of day which interposed
between Monday morning and this rather subdued moment , and played with the things one does
voluntarily and normally in the daylight . But although she#1 was silent , she#1 was evidently
mistress of a situation which was familiar enough to her#1 , and inclined to let it take its
way for the six hundredth time , perhaps , without bringing into play any of her#1 unoccupied
faculties . A single glance was enough to show that Mrs.#3 Hilbery#3 was so rich in the gifts
which make tea-parties of elderly distinguished people successful , that she#3 scarcely needed
any help from her#3 daughter#1 , provided that the tiresome business of teacups and bread and
butter was discharged for her#1 . Considering that the little party had been seated round the
tea-table for less than twenty minutes , the animation observable on their faces , and the amount
of sound they were producing collectively , were very creditable to the hostess#3 . It suddenly
came into Katharine#1 's#1 mind that if some one opened the door at this moment he would think
that they were enjoying themselves ; he would think , " What an extremely nice house to come
into ! "
```

Table 9: Prompt for WL Coreference

Instruction
Any word marked with # is supposed to be the head of a noun phrase. Expand this head to contain determiner and adjective phrases. Do not remove or add new words while expanding. Stick to the format.
Example Input:
Montraville# was a **Lieutenant#** in the **army#** : **Belcour#** was his brother **officer#** : they had been to take leave of their **friends#** previous to their departure for **America#** , and were now returning to **Portsmouth#** , where the **troops#** waited orders for embarkation
Example Output:
Montraville **(Montraville)** was a Lieutenant **(a Lieutenant in the army)** in the army **(the army)** : Belcour **(Belcour)** was his brother officer **(his brother officer)** : they had been to take leave of their friends **(their friends)** previous to their departure for America **(America)** , and were now returning to Portsmouth **(Portsmouth)** , where the troops **(the troops)** waited orders for embarkation
Example Input:
Arriving at the verge of the **town#** , he dismounted , and sending the **servant#** forward with the horses , proceeded toward the **place#** , where , in the midst of an extensive pleasure **ground#** , stood the **mansion#** which contained the lovely Charlotte **Temple#** .
Example Output:
Arriving at the verge of the town **(the town)** , he dismounted , and sending the servant **(the servant)** forward with the horses , proceeded toward the place **(the place)** , where , in the midst of an extensive pleasure ground **(an extensive pleasure ground)** , stood the mansion **(the mansion which contained the lovely Charlotte Temple)** which contained the lovely Charlotte Temple **(the lovely Charlotte Temple)** .
Example Input:
"You are a benevolent **fellow#** ," said a young **officer#** to him one day and I have a great mind to give you a fine subject to exercise the goodness of your heart upon.
Example Output:
"You are a benevolent fellow **(a benevolent fellow)** ," said a young officer **(a young officer)** to him one day and I have a great mind to give you a fine subject to exercise the goodness of your heart upon.

Table 10: Prompt for H2S Retrieval

Instruction

Annotate all the entity mentions in the following text with coreference clusters. Use Markdown tags to indicate clusters in the output, with the following format [mention] (#cluster_name). Do not modify any text outside (), only add text inside parenthesis. The cluster names of the key entities are already provided, mark the mentions of the entity with the corresponding cluster name. Mark the mentions of the other entities with (#others). Also briefly describe the key entities in 1-2 sentences before starting the coreference task.

Example Input:

Key Entities:
1. Katharine Hilbery (#katharine_hilbery)
2. Mr. Denham (#mr._denham)
3. Mrs. Hilbery (#mrs._hilbery)
4. Mr. Hilbery (#mr._hilbery)
5. Mr. Fortescue (#mr._fortescue)

Text:
CHAPTER I It was a Sunday evening in October, and in common with [many other young ladies of [her] (#) class] (#) , [Katharine Hilbery] (#) was pouring out tea . Perhaps a fifth part of [her] (#) mind was thus occupied , and the remaining parts leapt over the little barrier of day which interposed between Monday morning and this rather subdued moment , and played with the things one does voluntarily and normally in the daylight . But although [she] (#) was silent , [she] (#) was evidently [mistress] (#) of a situation which was familiar enough to [her] (#) , and inclined to let it take its way for the six hundredth time , perhaps , without bringing into play any of [her] (#) unoccupied faculties . A single glance was enough to show that [Mrs. Hilbery] (#) was so rich in the gifts which make tea-parties of [elderly distinguished people] (#) successful , that [she] (#) scarcely needed any help from [[her] (#) daughter] (#) , provided that the tiresome business of teacups and bread and butter was discharged for [her] (#) . Considering that [the little party] (#) had been seated round the tea-table for less than twenty minutes , the animation observable on [their] (#) faces , and the amount of sound [they] (#) were producing collectively , were very creditable to [the hostess] (#) . It suddenly came into [Katharine 's] (#) mind that if [some one] (#) opened the door at this moment [he] (#) would think that [they] (#) were enjoying [themselves] (#) ; [he] (#) would think , " What [an extremely nice house] (#) to come into ! "

Example Output:

Description of Key Entities present in the text:

#katharine_hilbery - Katharine Hilbery: A young and apparently rich lady and the daughter of Mrs. Hilbery. She and Mrs. Hilbery were organising a party for some distinguished elders.

#mrs._hilbery - Mrs. Hilbery: She is the mother of Katharine Hilbery and is a well-to-do member of the society and a very efficient and able hostess

Coreference:
CHAPTER I It was a Sunday evening in October , and in common with [many other young ladies of [her] (#katharine_hilbery) class] (#others) , [Katharine Hilbery] (#katharine_hilbery) was pouring out tea . Perhaps a fifth part of [her] (#katharine_hilbery) mind was thus occupied , and the remaining parts leapt over the little barrier of day which interposed between Monday morning and this rather subdued moment , and played with the things one does voluntarily and normally in the daylight . But although [she] (#katharine_hilbery) was silent , [she] (#katharine_hilbery) was evidently [mistress] (#others) of a situation which was familiar enough to [her] (#katharine_hilbery) , and inclined to let it take its way for the six hundredth time , perhaps , without bringing into play any of [her] (#katharine_hilbery) unoccupied faculties . A single glance was enough to show that [Mrs. Hilbery] (#mrs._hilbery) was so rich in the gifts which make tea-parties of [elderly distinguished people] (#others) successful , that [she] (#mrs._hilbery) scarcely needed any help from [[her] (#mrs._hilbery) daughter] (#katharine_hilbery) , provided that the tiresome business of teacups and bread and butter was discharged for [her] (#katharine_hilbery) . Considering that [the little party] (#others) had been seated round the tea-table for less than twenty minutes , the animation observable on [their] (#others) faces , and the amount of sound [they] (#others) were producing collectively , were very creditable to [the hostess] (#mrs._hilbery) . It suddenly came into [Katharine 's] (#katharine_hilbery) mind that if [some one] (#others) opened the door at this moment [he] (#others) would think that [they] (#others) were enjoying [themselves] (#others) ; [he] (#others) would think , " What [an extremely nice house] (#others) to come into ! "

Table 11: Prompt for evaluating linking performance

Instruction

Annotate all the entity mentions that refer to the key entities provided. The mention needs to include determiners and adjectives, if present. Use Markdown tags to indicate clusters in the output, with the following format [mention] (#cluster_name). The cluster names of the key entitites are already provided. Mark the mentions of the entity with the corresponding cluster name. Also briefly describe the key entities in 1-2 sentences before starting the coreference task.

Example Input:

Key Entities:
1. Katharine Hilbery (#katharine_hilbery)
2. Mr. Denham (#mr._denham)
3. Mrs. Hilbery (#mrs._hilbery)
4. Mr. Hilbery (#mr._hilbery)
5. Mr. Fortescue (#mr._fortescue)

Text:
CHAPTER I It was a Sunday evening in October , and in common with many other young ladies of her class , Katharine Hilbery was pouring out tea . Perhaps a fifth part of her mind was thus occupied , and the remaining parts leapt over the little barrier of day which interposed between Monday morning and this rather subdued moment , and played with the things one does voluntarily and normally in the daylight . But although she was silent , she was evidently mistress of a situation which was familiar enough to her , and inclined to let it take its way for the six hundredth time , perhaps , without bringing into play any of her unoccupied faculties . A single glance was enough to show that Mrs. Hilbery was so rich in the gifts which make tea-parties of elderly distinguished people successful , that she scarcely needed any help from her daughter , provided that the tiresome business of teacups and bread and butter was discharged for her . Considering that the little party had been seated round the tea-table for less than twenty minutes , the animation observable on their faces , and the amount of sound they were producing collectively , were very creditable to the hostess . It suddenly came into Katharine 's mind that if someone opened the door at this moment he would think that they were enjoying themselves ; he would think , " What an extremely nice house to come into ! "

Example Output:

Description of Key Entities present in the text:

#katharine_hilbery - Katharine Hilbery: A young and apparently rich lady and the daughter of Mrs. Hilbery. She and Mrs. Hilbery were organising a party for some distinguished elders.

#mrs._hilbery - Mrs. Hilbery: She is the mother of Katharine Hilbery and is a well-to-do member of the society and a very efficient and able hostess

Coreference:
CHAPTER I It was a Sunday evening in October , and in common with many other young ladies of [her] (#katharine_hilbery) class , [Katharine Hilbery] (#katharine_hilbery) was pouring out tea . Perhaps a fifth part of [her] (#katharine_hilbery) mind was thus occupied , and the remaining parts leapt over the little barrier of day which interposed between Monday morning and this rather subdued moment , and played with the things one does voluntarily and normally in the daylight . But although [she] (#katharine_hilbery) was silent , [she] (#katharine_hilbery) was evidently mistress of a situation which was familiar enough to [her] (#katharine_hilbery) , and inclined to let it take its way for the six hundredth time , perhaps , without bringing into play any of [her] (#katharine_hilbery) unoccupied faculties . A single glance was enough to show that [Mrs. Hilbery] (#mrs._hilbery) was so rich in the gifts which make tea-parties of elderly distinguished people successful , that [she] (#mrs._hilbery) scarcely needed any help from [[her] (#mrs._hilbery) daughter] (#katharine_hilbery) , provided that the tiresome business of teacups and bread and butter was discharged for [her] (#katharine_hilbery) . Considering that the little party had been seated round the tea-table for less than twenty minutes , the animation observable on their faces , and the amount of sound they were producing collectively , were very creditable to [the hostess] (#mrs._hilbery) . It suddenly came into [Katharine 's] (#katharine_hilbery) mind that if some one opened the door at this moment he would think that they were enjoying themselves ; he would think , " What an extremely nice house to come into ! "

Table 12: Prompt for Direct version of E2E MEI