ON THE LIMITS OF CANDIDATE GENERATION IN BIOMEDICAL ENTITY LINKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Biomedical entity linking systems rely on candidate generation to retrieve potential entities prior to disambiguation. This stage imposes a hard upper bound on end-to-end performance: if the correct entity is not retrieved, no downstream model can recover it. Despite this, candidate generation is typically treated as a standalone retrieval problem, with limited attention to its structural limitations. In this work, we argue that candidate generation is inherently incomplete under any single retrieval paradigm and that improving coverage requires exploiting complementary retrieval behaviors. We introduce a heterogeneous ensemble framework that combines diverse candidate generators and analyze how their differences contribute to increased coverage. Empirically, our approach substantially improves the inclusion of gold entities in candidate sets across multiple biomedical benchmarks. When paired with standard disambiguation models, including cross-encoders and LLM-based rerankers, these gains translate into consistent end-to-end improvements, with our best disambiguator achieving an average increase of 8.1 absolute points over the best-performing evaluated model on each dataset. We further show that diversity—not just individual model strength—is an important contributor to these gains, and we quantify the diminishing returns of adding additional generators. Finally, we demonstrate that a simple heuristic for aggregating candidates achieves near-optimal performance relative to an oracle, suggesting that most gains arise from coverage expansion rather than complex fusion strategies. Our results highlight candidate generation as a fundamental bottleneck in entity linking and provide a principled approach to mitigating it.

1 INTRODUCTION

Biomedical entity linking (EL) maps textual mentions to standardized concepts in curated knowledge bases and is a core component of biomedical NLP systems (Lee et al., 2016; Roberts et al., 2015). Most EL systems follow a two-stage pipeline: candidate generation retrieves potential entities, and a disambiguation model selects the correct one (Logeswaran et al., 2019; Humeau et al., 2020). While advances in cross-encoders and large language models (LLMs) have improved ranking accuracy (Wu et al., 2020; Ye & Mitchell, 2025), overall performance remains constrained by candidate generation: if the correct entity is absent, no downstream model can recover it; a limitation already highlighted by Partalidou et al. (2022).

Despite its importance, candidate generation is typically treated as a standalone retrieval problem. Prior work has explored dictionary matching (Aronson & Lang, 2010; Hoffart et al., 2011), sparse retrieval (Neumann et al., 2019), dense retrieval (Liu et al., 2021; Zhang et al., 2022), and autoregressive generation (Cao et al., 2021). These approaches implicitly assume a single retrieval paradigm can capture diverse mention–entity relationships. However, biomedical terminology exhibits substantial variability—synonymy, abbreviations, and compositional structure—making high coverage unlikely for any individual method (Blair et al., 2014; Henriksson et al., 2014).

We argue that candidate generation is inherently incomplete under any single paradigm and that improving coverage requires leveraging complementary retrieval behaviors. Different generators capture distinct regions of the mention–entity space: lexical methods handle exact matches, heuristic expansions recover structured variants, neural retrievers capture semantic similarity, and autoregressive models generate entities from context. Candidate generation can thus be viewed as the union of partially overlapping retrieval functions, where coverage depends on both strength and diversity.

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

Motivated by this view, we introduce a heterogeneous ensemble framework that combines diverse candidate generators (Figure 1) and aggregates their outputs. Rather than improving individual retrievers, our approach emphasizes complementarity. We analyze marginal coverage gains and overlap, showing that diversity—alongside individual model strength—is an important factor in performance. Comparing to an oracle upper bound, we find that a simple aggregation heuristic achieves near-optimal results.

Across multiple biomedical EL benchmarks, our approach substantially increases gold entity inclusion in candidate sets. When combined with downstream disambiguation models, including cross-encoders and LLM-based rerankers, this yields consistent gains in end-to-end performance, with an average improvement of 8.1 points over the strongest single model evaluated on each dataset, all of which are competitive baselines. Notably, even a lightweight configuration of our approach surpasses the strongest evaluated single-model baseline while reducing inference cost, highlighting the practical value of improved candidate coverage.

Our contributions are as follows:

- We present a conceptual perspective on candidate generation in entity linking, arguing that it is inherently incomplete under any single retrieval paradigm and frame it as the union of complementary retrieval functions.
- We introduce a heterogeneous ensemble framework that leverages diverse candidate generators and show that complementarity is an important contributor to performance gains alongside individual model strength.
- We provide an empirical analysis of coverage, including marginal gains, unique retrieval and overlap between generators, revealing diminishing returns as additional generators are added.
- We show that a simple aggregation heuristic approaches the oracle upper bound, indicating that most gains come from expanded coverage and that complex fusion strategies are unnecessary.
- We achieve consistent improvements in downstream biomedical entity linking performance across multiple benchmarks while maintaining computational efficiency.

2 Related Work

Entity Linking Methods Early entity linking approaches relied on heuristic rules, string matching, and structured knowledge bases (Aronson & Lang, 2010; Hoffart et al., 2011). More recent methods leverage transformer-based representations (Vaswani et al., 2023; Devlin et al., 2019), including domain-adapted biomedical models (Lee et al., 2019; Gu et al., 2021; Yasunaga et al., 2022), to encode mentions and entities in a shared semantic space for similarity-based retrieval (Liu et al., 2021; Wu et al., 2020). Other work has improved individual linking systems by addressing specific retrieval and ambiguity challenges, including synonym variation (Sung et al., 2020) and homonymous entity names (Garda & Leser, 2024). Most systems adopt a two-stage pipeline in which candidate generation is followed by disambiguation (Humeau et al., 2020; Wu et al., 2020), making overall performance dependent on candidate coverage. Recent benchmark of biomedical entity linking systems have further highlighted the contribution of candidate generation and disambiguation errors to end-to-end performance (Kartchner et al., 2023).

Ensemble Methods Prior work has explored ensembles to improve robustness by combining multiple systems (Ganaie et al., 2022; Rajani & Mooney, 2016; João et al., 2020; Zhang et al., 2019). These approaches typically operate at the final prediction stage using stacking or aggregation strategies. Closer to our setting, some work in entity normalization has also employed ensembling at earlier stages of the pipeline (Hebert et al., 2022; Borchert et al., 2025; Lai et al., 2022); however, these methods treat ensembling as a pragmatic enhancement rather than a principled design choice. For example, BioSyn (Sung et al., 2020) combines sparse TF-IDF and dense representations within a single model while xMEN (Borchert et al., 2025) combines TF-IDF and SapBERT during candidate generation before reranking. Motivated by these approaches, we move beyond pairwise or ad hoc combinations and study candidate generation across a broader range of heterogeneous retrieval paradigms. We analyze overlap, unique retrieval, marginal gains, and diminishing returns to quantify how retriever strength and complementary coverage jointly shape ensemble performance.

LLM-based Disambiguation Recent work has increasingly incorporated LLMs into biomedical entity linking. BioELQA formulates entity linking as multiple-choice question answering, using

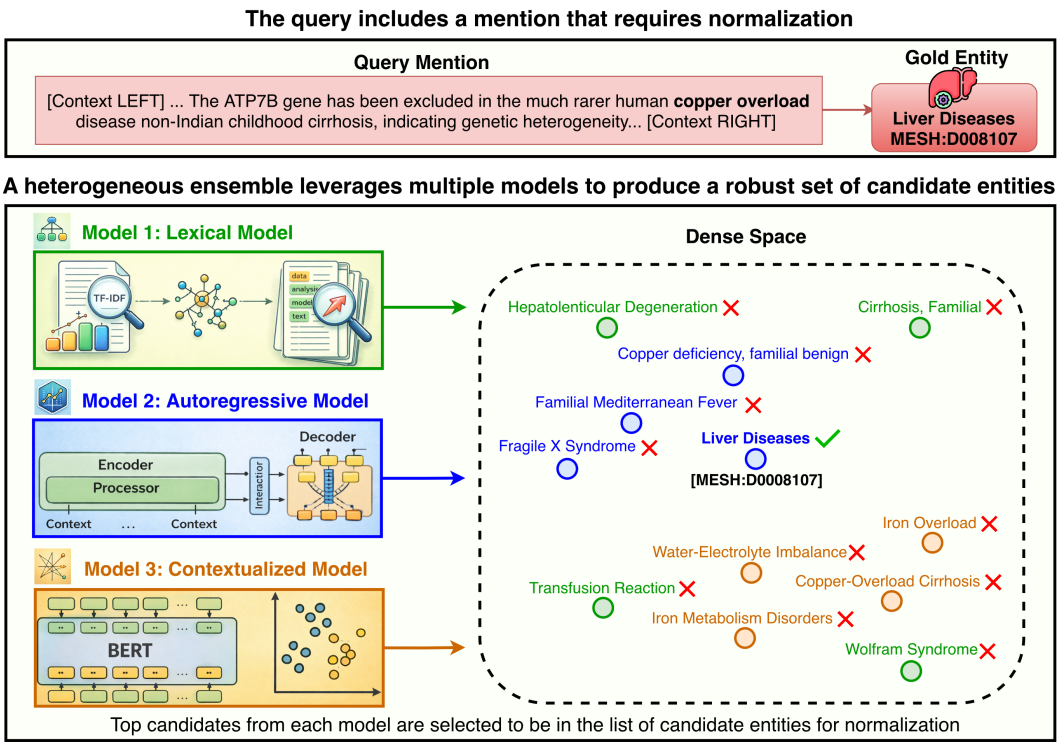


Figure 1: Heterogeneous ensemble improves candidate retrieval for entity normalization. Given an ambiguous mention in a query, multiple models: lexical (Model 1), autoregressive (Model 2), and contextualized (Model 3) are used to generate candidate entities. The top k candidates from each model are selected and combined into a unified candidate set. By leveraging complementary strengths across models, the correct entity ("Liver Diseases") is successfully retrieved, whereas individual models alone may fail, highlighting the benefit of ensemble strategy.

retrieved candidates as options for a generative model (Lin et al., 2024). Other approaches use LLMs to simplify complex biomedical mentions before retrieval (Borchert et al., 2024), constrain generation toward valid biomedical entities through restrictive decoding (Lin et al., 2025), or improve retrieval through generative relevance feedback (Shlyk & Hunter, 2026). LLMs have also been applied directly to reranking and final entity selection (Haffoudhi et al., 2026; Ye & Mitchell, 2025). Across these approaches, candidate generation remains an important component: methods that rerank or select among retrieved entities are limited by the candidates made available to them. This motivates the importance of improving candidate retrieval through more diverse generation strategies.

3 Method

3.1 Candidate Generation as Complementary Retrieval

We consider the candidate generation stage of entity linking as a retrieval problem that maps a mention m to a set of candidate entities $\mathcal{C}(m) \subseteq \mathcal{E}$, where $\mathcal{E}$ is the entity vocabulary. In standard approaches, a single retrieval function f is used to construct this set:

$$\mathcal{C}(m) = f(m).$$

However, due to the diversity of mention–entity relationships in biomedical text—including synonymy, abbreviation, and compositional variation—any individual retrieval function is inherently incomplete. Different retrieval paradigms capture distinct aspects of this mapping. For example, lexical methods favor exact or near-exact matches, while neural retrievers capture semantic similarity.

We therefore view candidate generation as the union of multiple retrieval functions:

$$\mathcal{C}(m) = \bigcup_{i=1}^{N} f_i(m),$$

where each f_i represents a candidate generator with distinct inductive biases. Under this formulation, overall coverage depends not only on the quality of individual generators but also on their complementarity, i.e., the extent to which they retrieve different subsets of relevant entities.

3.2 Heterogeneous Candidate Generators

We construct a set of heterogeneous candidate generators spanning multiple retrieval paradigms:

- **Exact or sparse lexical retrieval**, which retrieves candidates via exact or approximate string matching and token-level overlap between mentions and entity names or synonyms. We include SciSpacy (Neumann et al., 2019), which captures n-gram overlap to support both exact and partial matches, and SapBERT (Liu et al., 2021), which learns synonym-aware representations for more flexible alias matching.
- **Contextualized dense retrieval**, which embeds mentions and entities into a shared vector space and retrieves candidates based on semantic similarity. We include KRISSBERT (Zhang et al., 2022), trained with a self-supervised objective, and ArboEL (Agarwal et al., 2022), a bi-encoder–based retrieval model.
- **Autoregressive retrieval**, which generate additional candidates by directly producing normalized entity names from context. We include BioBART (Yuan et al., 2022a), an encoder–decoder model for autoregressive EL.

Each generator f_i produces a ranked list of k candidates per mention. These generators differ in retrieval behavior and are expected to retrieve partially overlapping but complementary sets.

3.3 Candidate Aggregation

Given candidate lists from multiple generators, we construct a unified candidate set. Let $\mathcal{C}_i(m)$ denote the ranked list produced by generator f_i, and we build a final set $\mathcal{C}_{ens}(m)$ of size k that maximizes coverage. We use a simple interleaving strategy that merges candidates while preserving diversity: generators are ranked by standalone performance, and we iterate in round-robin order, selecting the next highest-ranked candidate from each generator and skipping duplicates until k candidates are collected. This ensures representation across generators while minimizing redundancy and maintaining a fixed set size. Importantly, this aggregation does not rely on learned weighting or complex fusion; as shown in Section 4.5, it approaches an oracle upper bound, indicating that improvements stem primarily from expanded coverage rather than optimized combination weights.

3.4 Coverage and Complementarity

Under this framework, the effectiveness of candidate generation can be analyzed in terms of:

- **Coverage**: the proportion of mentions for which at least one generator retrieves the correct entity,
- **Overlap**: the degree to which different generators retrieve the same candidates,
- **Marginal contribution**: The additional coverage gained by introducing a new generator.

This perspective allows us to interpret the ensemble not simply as a combination of models, but as a mechanism for exploring complementary regions of the mention–entity space. In the experimental section, we empirically analyze these factors and show that diversity across generators is a key driver of improved performance.

3.5 Entity Disambiguation with Cross-Encoders and LLMs

Given the ensemble candidate set, we perform entity disambiguation using BERT-based cross-encoders and LLMs to assess the impact of improved candidate generation. Cross-encoder details are provided in Appendix A.2.1. For LLMs, we evaluate Qwen3.5-9B and Qwen3.5-27B models (Qwen, 2026). The overall framework of our heterogeneous ensemble approach is illustrated in Figure 2.

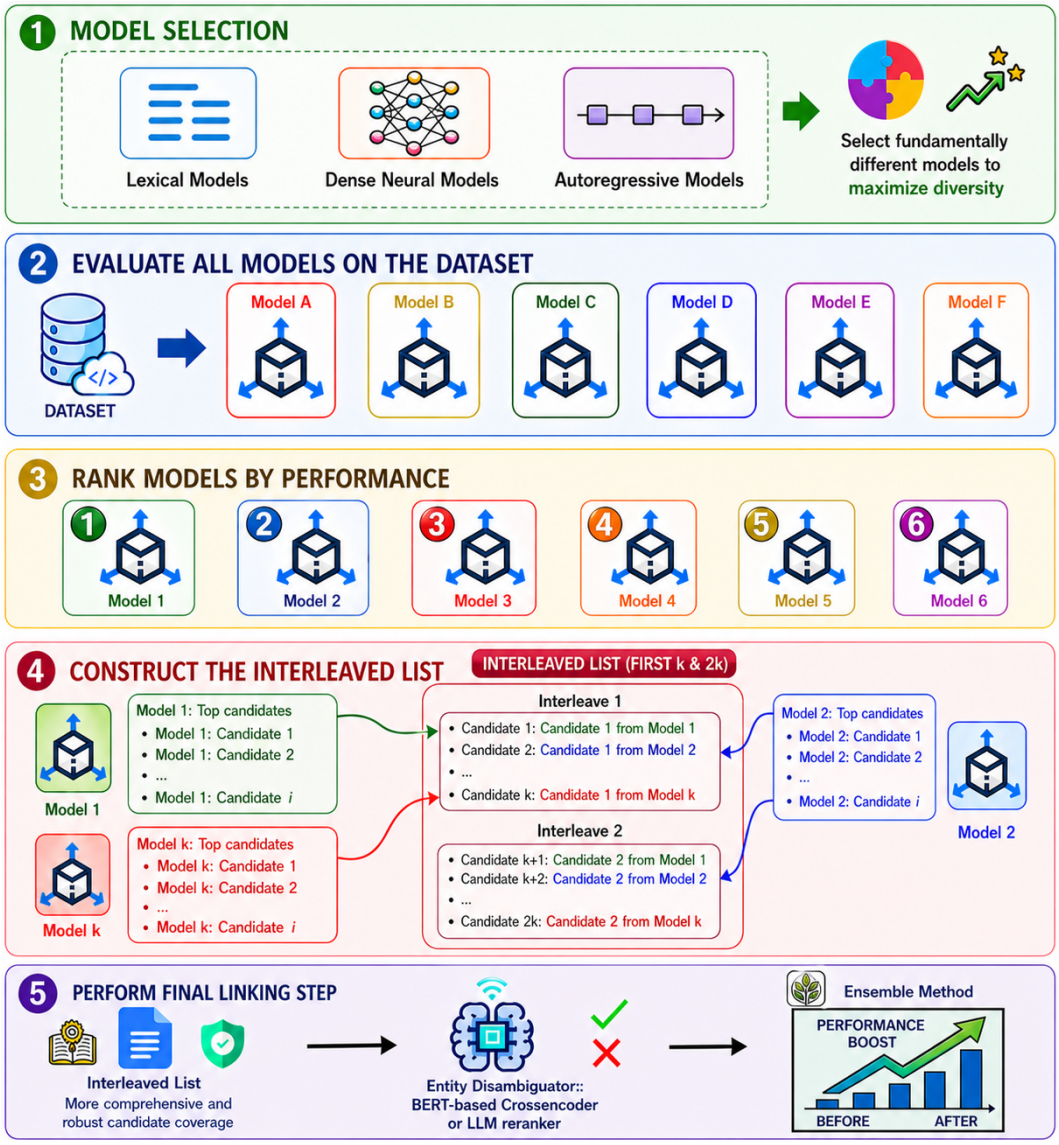


Figure 2: Overview of the heterogeneous ensemble framework for entity linking. In the first step, diverse and complementary models are selected to maximize candidate diversity. Each model is then evaluated on the target dataset and ranked according to its performance. Top candidates from each model are interleaved to construct a unified candidate list that balances contributions across models. This aggregated candidate set is then passed to an entity disambiguation module for final linking.

3.6 Experimental Setup

Datasets. We evaluate on a diverse set of biomedical entity linking benchmarks spanning multiple domains, ontologies, and text sources (Table 1), including PubMed abstracts (PubMed, 2026) and PubMed Central (PMC) full-text articles (PMC, 2026), accessed via BigBio (Fries et al., 2022).

The datasets vary in size, ontology complexity, and mention distribution, providing a challenging testbed for candidate generation. In particular, differences in train–test overlap and entity diversity allow us to assess coverage under varying levels of lexical and semantic variability. Additional dataset details are provided in Appendix A.1.

Models. Our ensemble uses the five candidate generators described in Section 3.2: SapBERT, KRISSBERT, SciSpacy, ArboEL, and BioBART. For benchmarking, we additionally evaluate xMEN

Dataset	#Train	#Val	#Test	Overlap	Source	# Unique Ents	Ontology
NCBI Disease	5,134	787	960	0.816	PubMed Abstracts	789	MEDIC
BC5CDR	9,428	9,638	9,750	0.773	PubMed Abstracts	2,348	MeSH
NLM Chem	20,866	5,269	11,624	0.623	PMC Full-Text	1,787	MeSH
GNormPlus	3,489	724	3222	0.084	PubMed Abstracts	1,353	Entrez
NLM Gene	10,660	2,011	2729	0.541	PMC Full-Text	3,348	Entrez
MedMentions ST21PV	122,174	40,864	40,143	0.774	PubMed Abstracts	25,419	UMLS

Table 1: Statistics of the datasets used. #Train, #Val, and #Test denote the number of mentions in each split. Overlap measures the proportion of shared mentions across training and test splits.

(Borchert et al., 2025) as a separate baseline, but do not include it in the ensemble. xMEN is particularly relevant to our setting as it combines TF-IDF and SapBERT for candidate generation.

Implementation Details. We implement all candidate generators within the BioEL framework (Bathala et al., 2025). Each generator retrieves $k = 64$ candidates per mention, and the final ensemble list is constrained to the same size via interleaving. Performance plateaus beyond $k = 64$ (Table 9), further supporting this choice of configuration. Full implementation details are provided in Appendix A.2. For downstream disambiguation, we evaluate a cross-encoder (Wu et al., 2020) and LLM-based rerankers following Ye & Mitchell (2025), with details provided in Appendix A.3.

Metrics. We evaluate candidate generation using Recall@k, the proportion of mentions for which the correct entity appears in the top-k retrieved candidates. This metric directly captures candidate coverage and defines an upper bound on downstream performance. For end-to-end entity linking, we report accuracy, the proportion of correctly linked mentions. Improvements in accuracy directly reflect how improved coverage translates into better disambiguation performance.

4 Results

4.1 Improved Candidate Coverage

We first evaluate the effectiveness of the proposed approach in improving candidate generation. Table 2 reports Recall@k across all datasets. All results are based on a single run.

The heterogeneous ensemble consistently improves recall@k metric over individual generators and prior methods, indicating that combining retrieval paradigms increases the likelihood that the correct entity is included. These gains are achieved without increasing candidate set size, confirming that improvements stem from coverage rather than list expansion. Additional results for $k = 1$ to 64 are provided in the Appendix Figure 3.

4.2 Impact on End-to-End Entity Linking

We next evaluate downstream performance using cross-encoders and LLM-based rerankers (Table 3), observing consistent improvements across datasets with an average gain of 8.1% for the best disambiguator, confirming that increased candidate coverage improves disambiguation.

We observe a slight degradation in cross-encoder performance on BC5CDR, likely caused by near-saturation. On this dataset, the candidate generation stage already exceeds 98% accuracy on the training set (Figure 4 in the Appendix), limiting the cross-encoder's ability to further discriminate between candidates.

For LLM-based disambiguation, Qwen3.5-27B improves over SOTA on all evaluated datasets, demonstrating strong generalization capabilities. Qwen3.5-9B exhibits a performance drop on the NLM-Gene dataset, suggesting that smaller LLMs may struggle with the complexity of gene normalization tasks.

4.3 Effect of Model Diversity

To better understand the source of these improvements, we analyze the effect of combining different types of candidate generators. We compare ensembles composed of similar models (e.g., multiple dense retrievers) with those combining heterogeneous models. Concretely, *Ensemble 1* consists of

Dataset		SapBERT	KRISSBERT	SciSpacy	ArboEL	BioBART	xMEN	Ensemble
Model	**k**				**Recall@k**			
NCBI-Disease	**1**	0.800	0.746	0.748	0.802	0.627	0.833	—
	10	0.902	0.820	0.874	0.902	0.798	0.911	**0.951** (+0.040)
	64	0.938	0.833	0.924	0.922	0.890	0.934	**0.991** (+0.053)
BC5CDR	**1**	0.815	0.720	0.843	0.902	0.822	0.880	—
	10	0.908	0.765	0.919	0.945	0.914	0.925	**0.988** (+0.043)
	64	0.927	0.775	0.945	0.951	0.958	0.932	**0.995** (+0.037)
NLM-Chem	**1**	0.662	0.551	0.616	0.790	0.507	0.759	—
	10	0.826	0.596	0.729	0.860	0.674	0.837	**0.937** (+0.077)
	64	0.884	0.610	0.775	0.865	0.828	0.884	**0.957** (+0.073)
GNormPlus	**1**	0.191	0.108	0.110	0.585	0.084	0.444	—
	10	0.806	0.122	0.547	0.655	0.418	0.837	**0.890** (+0.053)
	64	0.908	0.122	0.674	0.661	0.573	0.918	**0.938** (+0.020)
NLM-Gene	**1**	0.072	0.278	0.055	0.560	0.058	0.405	—
	10	0.586	0.512	0.406	0.770	0.450	0.778	**0.785** (+0.007)
	64	0.897	0.533	0.628	0.783	0.742	0.893	**0.938** (+0.041)
MM-ST21PV	**1**	0.558	0.548	0.613	0.687	0.614	0.697	—
	10	0.781	0.712	0.806	0.808	0.867	0.814	**0.919** (+0.052)
	64	0.838	0.747	0.859	0.815	0.932	0.846	**0.968** (+0.036)

Table 2: Comparison of Recall@k across all individual biomedical entity linking models and the proposed ensemble method. Underlined scores denotes the best performance among individual models, while **bold scores** indicates the overall best performance.

	Best Evaluated Model	Disambiguator		
Dataset		**Cross-encoder**	**Qwen3.5-9B**	**Qwen3.5-27B**
NCBI-Disease	0.833	0.851 (+0.018)	0.866 (+0.033)	**0.875** (+0.042)
BC5CDR	0.902	0.896 (–0.006)	**0.918** (+0.016)	**0.918** (+0.016)
NLM-Chem	0.790	0.853 (+0.063)	0.889 (+0.099)	**0.900** (+0.110)
GNormPlus	0.585	0.769 (+0.184)	0.643 (+0.058)	**0.832** (+0.247)
NLM-Gene	0.560	**0.628** (+0.068)	0.511 (–0.049)	0.591 (+0.031)
MM-ST21PV	0.697	0.740 (+0.043)	**0.748** (+0.051)	0.739 (+0.042)
Avg.	0.728	0.790 (+0.062)	0.763 (+0.035)	**0.809** (+0.081)

Table 3: Accuracy Comparison of Disambiguation Models on Heterogeneous Ensemble Candidate.

three architecturally distinct models: ArboEL, SciSpacy, and BioGenEL. In contrast, *Ensemble 2* reduced diversity by combining ArboEL, BioBART, and BioGenEL, where BioGenEL (Yuan et al., 2022b) is an enhanced variant of BioBART specifically designed for EL.

To ensure a fair comparison focused on diversity, we include SciSpacy in the first ensemble as it is the most comparable to BioBART in terms of performance among the remaining models. This design choice allows us to isolate the effect of model diversity without confounding it with differences in individual model strength.

Results in Table 4 and Figure 5 in the Appendix show that heterogeneous ensembles consistently outperform homogeneous ones, despite comparable performance among their individual models. This indicates that retrieval diversity is an important contributor to improved candidate coverage.

4.4 Coverage and Complementarity Analysis

To quantify complementarity between candidate generators, we analyze the overlap and marginal contribution of each model. Specifically, we measure: (i) the proportion of mentions for which each generator uniquely retrieves the correct entity in Table 5, (ii) the overlap between candidate sets produced by different generators in Table 6, (iii) the incremental gain in Recall@k as additional generators are added in Figure 6.

Dataset	Ensemble 1 ArboEL / SciSpacy / BioGenEL	Ensemble 2 ArboEL / BioBART / BioGenEL
NCBI-Disease	**0.874**	0.858
BC5CDR	**0.912**	0.910
NLM-Chem	**0.888**	0.883
GNormPlus	**0.789**	0.705
NLM-Gene	**0.581**	0.570
MM-ST21PV	**0.725**	0.721
Avg.	**0.795**	0.775

Table 4: Final linking accuracy of Qwen3.5-27B using two candidate generation ensembles with different levels of model diversity. Ensemble 1 (ArboEL, SciSpacy, BioGenEL) combines architecturally heterogeneous models, while Ensemble 2 (ArboEL, BioBART, BioGenEL) includes two closely related generative models, resulting in lower diversity.

The results indicate that each generator contributes to the overall quality of the candidate lists by retrieving distinct subsets of correct entities not captured by other models. As shown in Table 5, even the weakest standalone generator, KRISSBERT, contributes more unique gold entities to the ensemble than BioBART or SciSpacy. Similarly, on GNormPlus, SapBERT contributes more unique gold entities than ArboEL despite substantially lower standalone accuracy (19.1% vs. 80.0%). However, we observe diminishing returns as additional generators are added: each successive model contributes progressively fewer unique correct entities, irrespective of whether generators are added in ascending or descending order of performance, as illustrated in Figure 6 in the Appendix. This indicates that the most significant gains arise from combining a small number of complementary models, as additional generators increasingly tend to retrieve entities already identified by earlier ones. These findings support our hypothesis that improvements stem from both complementarity between retrieval strategies and individual model performance.

Generator	NCBI-Disease	BC5CDR	NLMChem	GNormPlus	NLM-Gene	MM-ST21PV	Avg.
ArboEL	0.21%	0.07%	0.35%	4.72%	7.20%	0.76%	2.22%
SapBERT	0.42%	0.07%	0.46%	4.83%	1.59%	0.54%	1.31%
SciSpacy	1.15%	0.07%	0.43%	1.15%	0.55%	1.28%	0.77%
BioBART	0.00%	0.24%	0.62%	0.71%	0.70%	1.59%	0.64%
KrissBERT	0.83%	0.29%	1.51%	0.62%	2.05%	0.72%	1.01%

Table 5: Proportion of unique retrieval for each generator across datasets. Each value represents the percentage of mentions retrieved only by the corresponding generator and missed by all others with 12 candidates per model

	NCBI-Disease	BC5CDR	NLMChem	GNormPlus	NLM-Gene	MM-ST21PV	Avg.
1 generator	1.87%	0.35%	2.35%	8.04%	4.80%	3.15%	3.43%
2 generators	4.27%	1.88%	5.23%	19.65%	9.09%	6.12%	7.71%
3 generators	5.83%	5.41%	11.80%	36.06%	21.33%	10.92%	15.23%
4 generators	18.75%	24.92%	38.76%	25.33%	34.12%	18.60%	26.75%
5 generators	68.65%	67.12%	38.53%	7.39%	26.97%	59.32%	44.66%

Table 6: Distribution of retrieval overlap across generators for each dataset. Each row "n generators" represents the percentage of mentions retrieved by exactly n distinct generators.

4.5 Comparison to Oracle Upper Bound

To assess the effectiveness of our aggregation strategy, we compare against an oracle that selects the best possible combination of candidates from the five generators at the mention level for each dataset. This oracle maximizes recall@k for all k within their combined candidate pool, providing an upper bound on candidate-generation coverage for this specific generator set. End-to-end accuracy may still depend on candidate ordering and its interaction with the disambiguator.

Results in Table 7 show that our simple interleaving heuristic achieves performance close to this oracle upper bound. This suggests that most gains comes from expanding candidate coverage, and

that more complex fusion strategies are unnecessary as they can offer only limited additional benefit. For completeness, we also evaluate Reciprocal Rank Fusion and Borda count in Appendix A.4.1.

Dataset	Cross-encoder		Qwen3.5-9B	
	Heuristic	Oracle	Heuristic	Oracle
NCBI-Disease	0.851	0.852 (+0.001)	0.866	0.868 (+0.002)
BC5CDR	0.896	0.901 (+0.005)	0.918	0.921 (+0.003)
NLM-Chem	0.853	0.846 (-0.007)	0.889	0.893 (+0.004)
GNormPlus	0.769	0.775 (+0.006)	0.643	0.652 (+0.009)
NLM-Gene	0.628	0.637 (+0.009)	0.511	0.514 (+0.003)
MM-ST21PV	0.740	0.744 (+0.004)	0.748	0.763 (+0.015)
Average	0.790	0.793 (+0.003)	0.763	0.768 (+0.005)

Table 7: Impact of model ordering on downstream disambiguation. **Heuristic** is the simple strategy that consist of prioritizing candidates generated by higher-performing models first and **Oracle** represents the best ordering of candidates from the different model at mention level.

4.6 Efficiency Analysis

Finally, although the full ensemble can be computationally demanding, it can be adapted to resource-constrained settings by using more efficient generators. As shown in Table 8, a lightweight configuration of our ensemble (SciSpacy, SapBERT, and KRISSBERT) outperforms ArboEL, the strongest single model evaluated in our experiments, by 3.2% while reducing average running time (training + inference) by a factor of 3.17. This highlights the practical feasibility of the approach and suggests that improved coverage can be achieved without prohibitive computational overhead.

Dataset	Accuracy		Time (hours)		△ Accuracy	Speedup Factor
	ArboEL	Lightweight Ensemble	ArboEL	Lightweight Ensemble		
NCBI-Disease	0.802	0.842	1.99	1.23	+4.0%	1.6×
BC5CDR	0.902	0.898	7.94	3.92	-0.4%	2.0×
NLMChem	0.790	0.802	18.78	14.82	+1.2%	1.3×
GNormPlus	0.585	0.679	7.58	3.05	+9.4%	2.5×
NLM-Gene	0.560	0.565	16.04	7.42	+0.5%	2.2×
MM-ST21PV	0.687	0.734	423.63	44.78	+2.4%	9.5×
Avg.	0.721	0.753	—	—	**+3.2%**	**3.17x**

Table 8: Accuracy and running time comparison between our Lightweight Ensemble and ArboEL. The Lightweight Ensemble ensemble achieves higher accuracy while requiring less computational time, highlighting the framework efficiency under resource constraints.

5 Conclusion

We introduced a heterogeneous ensemble approach to candidate generation for biomedical entity linking, based on the observation that no single retrieval paradigm achieves sufficient coverage. By combining complementary generators, our method improves candidate coverage and yields consistent gains in downstream linking performance across six benchmarks. Our analysis shows that these improvements are driven by both diversity and individual model strength, with diminishing returns as additional generators are added. Moreover, we demonstrate that a simple aggregation heuristic approaches the candidate generator oracle upper bound, suggesting that most of the achievable improvements stem from combining complementary retrieval signals and further optimization on fusion mechanisms is unnecessary. Importantly, our framework maintains strong performance while remaining computationally efficient, making it suitable for scalable and cost-effective biomedical EL.

Limitations. Our approach relies on complementary candidate generators; as retrieval overlap increases, the benefits of ensembling diminish. Moreover, maximizing performance may require larger ensembles, increasing computational cost and potentially limiting deployment in latency-sensitive settings. Finally, our evaluation focuses on biomedical entity linking, and performance may differ in domains with less lexical and synonym variation.

AI USE STATEMENT

We used generative AI tools to refine the wording and clarity of portions of the manuscript, assist with code generation and debugging, and help identify potentially relevant related work. All AI-assisted text was reviewed and edited by the authors, all generated code was verified and tested for correctness, and suggested related works were checked by the authors. Generative AI tools were not used to generate research ideas or draw scientific conclusions. The authors take full responsibility for the final content of the paper, including all text, results, claims, and code produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce our experiments in Section 3.6 and the Appendix. Section 3.6 describes the datasets, evaluated models, and implementation setup. The Appendix provides additional dataset details (Section A.1), model hyperparameters and training times (Section A.2), and LLM prompts, parameters, and inference times (Section A.3). It also includes additional robustness analyses across random seeds (Section A.5.1).

ACKNOWLEDGMENTS

REFERENCES

Dhruv Agarwal, Rico Angell, Nicholas Monath, and Andrew McCallum. Entity linking via explicit mention-mention coreference modeling. pp. 4644–4658, July 2022. doi: 10.18653/v1/2022.naacl-main.343. URL https://aclanthology.org/2022.naacl-main.343.

Alan Aronson and François-Michel Lang. An overview of metamap: Historical perspective and recent advances. *Journal of the American Medical Informatics Association : JAMIA*, 17:229–36, 05 2010. doi: 10.1136/jamia.2009.002733.

Prasanth Bathala, Christophe Ye, Batuhan Nursal, Shubham Lohiya, David Kartchner, and Cassie S. Mitchell. BioEL: A comprehensive python package for biomedical entity linking. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1709–1721, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.93. URL https://aclanthology.org/2025.findings-naacl.93/.

David R. Blair, Kuansan Wang, Svetlozar Nestorov, James A. Evans, and Andrey Rzhetsky. Quantifying the impact and extent of undocumented biomedical synonymy. *PLoS Computational Biology*, 10(9):e1003799, 2014. doi: 10.1371/journal.pcbi.1003799.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

Florian Borchert, Ignacio Llorca, and Matthieu-P Schapranow. Improving biomedical entity linking for complex entity mentions with llm-based text simplification. *Database*, 2024:baae067, 07 2024. ISSN 1758-0463. doi: 10.1093/database/baae067. URL https://doi.org/10.1093/database/baae067.

Florian Borchert, Ignacio Llorca, Roland Roller, Bert Arnrich, and Matthieu-P Schapranow. xmen: a modular toolkit for cross-lingual medical entity normalization. *JAMIA Open*, 8(1):ooae147, 02 2025. ISSN 2574-2531. doi: 10.1093/jamiaopen/ooae147. URL https://doi.org/10.1093/jamiaopen/ooae147.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=5k8F6UU39V.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. pp. 4171–4186, June 2019. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Rezarta Dogan, Robert Leaman, Sun Kim, Dongseop Kwon, Chih-Hsuan Wei, Donald Comeau, Yifan Peng, David Cissel, Cathleen Coss, Carol Fisher, Rob Guzman, Preeti Kochar, Stella Koppel, Dorothy Trinh, Keiko Sekiya, Janice Ward, Deborah Whitman, Susan Schmidt, and Zhiyong lu. Nlm-chem, a new resource for chemical entity recognition in pubmed full text literature. *Scientific Data*, 8, 03 2021a. doi: 10.1038/s41597-021-00875-1.

Rezarta Dogan, Chih-Hsuan Wei, David Cissel, Nicholas Miliaras, Olga Printseva, Oleg Rodionov, Keiko Sekiya, Janice Ward, and Zhiyong lu. Nlm-gene, a richly annotated gold standard dataset for gene entities that addresses ambiguity and multi-species gene recognition. *Journal of Biomedical Informatics*, 118:103779, 04 2021b. doi: 10.1016/j.jbi.2021.103779.

Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10, 2014. URL https://api.semanticscholar.org/CorpusID:234064.

Jason Alan Fries, Leon Weber, Natasha Seelam, Gabriel Altay, Debajyoti Datta, Ruisi Su, Samuele Garda, Sunny MS Kang, Stella Biderman, Matthias Samwald, Stephen H. Bach, Wojciech Kusa, Samuel Cahyawijaya, Fabio Barth, Simon Ott, Mario Sänger, Bo Wang, Alison Callahan, Daniel León Perinan, Théo Gigant, Patrick Haller, Jenny Chim, Jose Posada, John Giorgi, Karthik Rangasai Sivaraman, Marc Pàmies, Marianna Nezhurina, Robert Martin, Moritz Freidank, Nathan Dahlberg, Shubhanshu Mishra, Shamik Bose, Nicholas Broad, Yanis Labrak, Shlok S Deshmukh, Sid Kiblawi, Ayush Singh, Minh Chien Vu, Trishala Neeraj, Jonas Golde, Albert Villanova del Moral, and Benjamin Beilharz. Bigbio: a framework for data-centric biomedical natural language processing. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

M.A. Ganaie, Minghui Hu, A.K. Malik, M. Tanveer, and P.N. Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2022.105151. URL https://www.sciencedirect.com/science/article/pii/S095219762200269X.

Samuele Garda and Ulf Leser. Belhd: improving biomedical entity linking with homonym disambiguation. *Bioinformatics*, 40(8):btae474, 07 2024. ISSN 1367-4811. doi: 10.1093/bioinformatics/btae474. URL https://doi.org/10.1093/bioinformatics/btae474.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23, October 2021. ISSN 2637-8051. doi: 10.1145/3458754. URL http://dx.doi.org/10.1145/3458754.

Samy Haffoudhi, Fabian M. Suchanek, and Nils Holzenberger. Lela: an llm-based entity linking approach with zero-shot domain adaptation, 2026. URL https://arxiv.org/abs/2601.05192.

Ada Hamosh, Alan F Scott, Joanna S Amberger, Carol A Bocchini, and Victor A McKusick. Online mendelian inheritance in man (omim), a knowledgebase of human genes and genetic disorders. *Nucleic acids research*, 33(suppl_1):D514–D517, 2005.

Liam Hebert, Raheleh Makki, Shubhanshu Mishra, Hamidreza Saghir, Anusha Kamath, and Yuval Merhav. Robust candidate generation for entity linking on short social media texts. In *Proceedings of the Eighth Workshop on Noisy User-generated Text (W-NUT 2022)*, pp. 83–89, Gyeongju, Republic of Korea, October 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.wnut-1.8/.

Aron Henriksson, Hans Moen, Maria Skeppstedt, Vidas Daudaravičius, and Martin Duneld. Synonym extraction and abbreviation expansion with ensembles of semantic spaces. *Journal of Biomedical Semantics*, 5(1):6, 2014. ISSN 2041-1480. doi: 10.1186/2041-1480-5-6. URL https://doi.org/10.1186/2041-1480-5-6.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In Regina Barzilay and Mark Johnson (eds.), *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 782–792, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL https://aclanthology.org/D11-1072/.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring, 2020. URL https://arxiv.org/abs/1905.01969.

Renato Stoffalette João, Pavlos Fafalios, and Stefan Dietze. Better together: an ensemble learner for combining the results of ready-made entity linking systems. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, SAC '20, pp. 851–858. ACM, March 2020. doi: 10.1145/3341105.3373883. URL http://dx.doi.org/10.1145/3341105.3373883.

David Kartchner, Jennifer Deng, Shubham Lohiya, Tejasri Kopparthi, Prasanth Bathala, Daniel Domingo-Fernández, and Cassie Mitchell. A comprehensive evaluation of biomedical entity linking models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14462–14478, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.893. URL https://aclanthology.org/2023.emnlp-main.893.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention, 2023. URL https://arxiv.org/abs/2309.06180.

Tuan Lai, Heng Ji, and ChengXiang Zhai. Improving candidate retrieval with entity profile generation for Wikidata entity linking. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 3696–3711, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.292. URL https://aclanthology.org/2022.findings-acl.292/.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, September 2019. ISSN 1367-4811. doi: 10.1093/bioinformatics/btz682. URL http://dx.doi.org/10.1093/bioinformatics/btz682.

Sunwon Lee, Donghyeon Kim, Kyubum Lee, Jaehoon Choi, Seongsoon Kim, Minji Jeon, Sangrak Lim, Donghee Choi, Sunkyu Kim, Aik Choon Tan, and Jaewoo Kang. Best: Next-generation biomedical entity search tool for knowledge discovery from biomedical literature. *PLOS ONE*, 11, 10 2016. doi: 10.1371/journal.pone.0164680.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database*, 2016:baw068, 05 2016. ISSN 1758-0463. doi: 10.1093/database/baw068. URL https://doi.org/10.1093/database/baw068.

Zhenxi Lin, Ziheng Zhang, Xian Wu, and Yefeng Zheng. Biomedical entity linking as multiple choice question answering. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 2390–2396, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.214/.

Zhenxi Lin, Ziheng Zhang, Jian Wu, Yefeng Zheng, and Xian Wu. Guiding large language models for biomedical entity linking via restrictive and contrastive decoding. In Christos Christodoulopoulos,

Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 23745–23759, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.1292. URL https://aclanthology.org/2025.findings-emnlp.1292/.

Carolyn E Lipscomb. Medical subject headings (mesh). *Bulletin of the Medical Library Association*, 88(3):265, 2000.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. Self-alignment pretraining for biomedical entity representations. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4228–4238, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.334. URL https://aclanthology.org/2021.naacl-main.334.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. Zero-shot entity linking by reading entity descriptions. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3449–3460, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1335. URL https://aclanthology.org/P19-1335/.

Donna Maglott, Jim Ostell, Kim D Pruitt, and Tatiana Tatusova. Entrez gene: gene-centered information at ncbi. *Nucleic acids research*, 33(suppl_1):D54–D58, 2005.

Sunil Mohan and Donghui Li. Medmentions: A large biomedical corpus annotated with umls concepts, 2019. URL https://arxiv.org/abs/1902.09476.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. ScispaCy: Fast and robust models for biomedical natural language processing. In Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii (eds.), *Proceedings of the 18th BioNLP Workshop and Shared Task*, pp. 319–327, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5034. URL https://aclanthology.org/W19-5034.

Eleni Partalidou, Despina Christou, and Grigorios Tsoumakas. Improving zero-shot entity retrieval through effective dense representations. In *Proceedings of the 12th Hellenic Conference on Artificial Intelligence*, SETN '22, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450395977. doi: 10.1145/3549737.3549771. URL https://doi.org/10.1145/3549737.3549771.

PMC. Pmc, 2026. URL https://pmc.ncbi.nlm.nih.gov. Website to access PMC.

PubMed. Pubmed, 2026. URL https://pubmed.ncbi.nlm.nih.gov/. Website to access Pubmed.

Qwen. Qwen, 2026. URL https://qwen.ai/blog?id=qwen3.5. Qwen website to Qwen 3.5 model.

Nazneen Fatema Rajani and Raymond Mooney. Combining supervised and unsupervised enembles for knowledge base population. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1943–1948, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1201. URL https://aclanthology.org/D16-1201/.

Kirk Roberts, Matthew S. Simpson, Ellen M. Voorhees, and William R. Hersh. Overview of the TREC 2015 clinical decision support track. In Ellen M. Voorhees and Angela Ellis (eds.), *Proceedings of The Twenty-Fourth Text REtrieval Conference, TREC 2015, Gaithersburg, Maryland, USA, November 17-20, 2015*, volume 500-319 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2015. URL http://trec.nist.gov/pubs/trec24/papers/Overview-CL.pdf.

Darya Shlyk and Lawrence Hunter. Improving biomedical entity linking with generative relevance feedback. *Bioinformatics*, 42(2):btag011, 02 2026. ISSN 1367-4811. doi: 10.1093/bioinformatics/btag011. URL https://doi.org/10.1093/bioinformatics/btag011.

Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. Biomedical entity representations with synonym marginalization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3641–3650, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.335. URL https://aclanthology.org/2020.acl-main.335/.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong lu. Gnormplus: An integrative approach for tagging genes, gene families, and protein domains. *BioMed research international*, 2015:918710, 08 2015. doi: 10.1155/2015/918710.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. Scalable zero-shot entity linking with dense entity retrieval. 2020.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. Linkbert: Pretraining language models with document links, 2022. URL https://arxiv.org/abs/2203.15827.

Christophe Ye and Cassie S. Mitchell. LLM as entity disambiguator for biomedical entity-linking. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 301–312, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-252-7. doi: 10.18653/v1/2025.acl-short.25. URL https://aclanthology.org/2025.acl-short.25/.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. BioBART: Pretraining and evaluation of a biomedical generative language model. In Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii (eds.), *Proceedings of the 21st Workshop on Biomedical Language Processing*, pp. 97–109, Dublin, Ireland, May 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.bionlp-1.9. URL https://aclanthology.org/2022.bionlp-1.9.

Hongyi Yuan, Zheng Yuan, and Sheng Yu. Generative biomedical entity linking via knowledge base-guided pre-training and synonyms-aware fine-tuning. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4038–4048, Seattle, United States, July 2022b. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.296. URL https://aclanthology.org/2022.naacl-main.296.

Fanjin Zhang, Xiao Liu, Jie Tang, Yuxiao Dong, Peiran Yao, Jie Zhang, Xiaotao Gu, Yan Wang, Bin Shao, Rui Li, and Kuansan Wang. Oag: Toward linking large-scale heterogeneous entity graphs. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pp. 2585–2595, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362016. doi: 10.1145/3292500.3330785. URL https://doi.org/10.1145/3292500.3330785.

Sheng Zhang, Hao Cheng, Shikhar Vashishth, Cliff Wong, Jinfeng Xiao, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Knowledge-rich self-supervision for biomedical entity linking, December 2022. URL https://aclanthology.org/2022.findings-emnlp.61.

A APPENDIX

Dataset	R@32	R@64	R@128	R@200
NCBI-Disease	0.984	0.991	0.992	0.994
BC5CDR	0.993	0.995	0.996	0.997
NLM-Chem	0.948	0.957	0.961	0.964
GNormPlus	0.926	0.938	0.951	0.958
NLM-Gene	0.916	0.938	0.951	0.956
MM-ST21PV	0.955	0.968	0.976	0.980

Table 9: Recall@32, Recall@64, Recall@128, and Recall@200 of the ensemble strategy across datasets, showing that performance largely plateaus after $k = 64$, with only marginal gains from increasing the number of candidates.

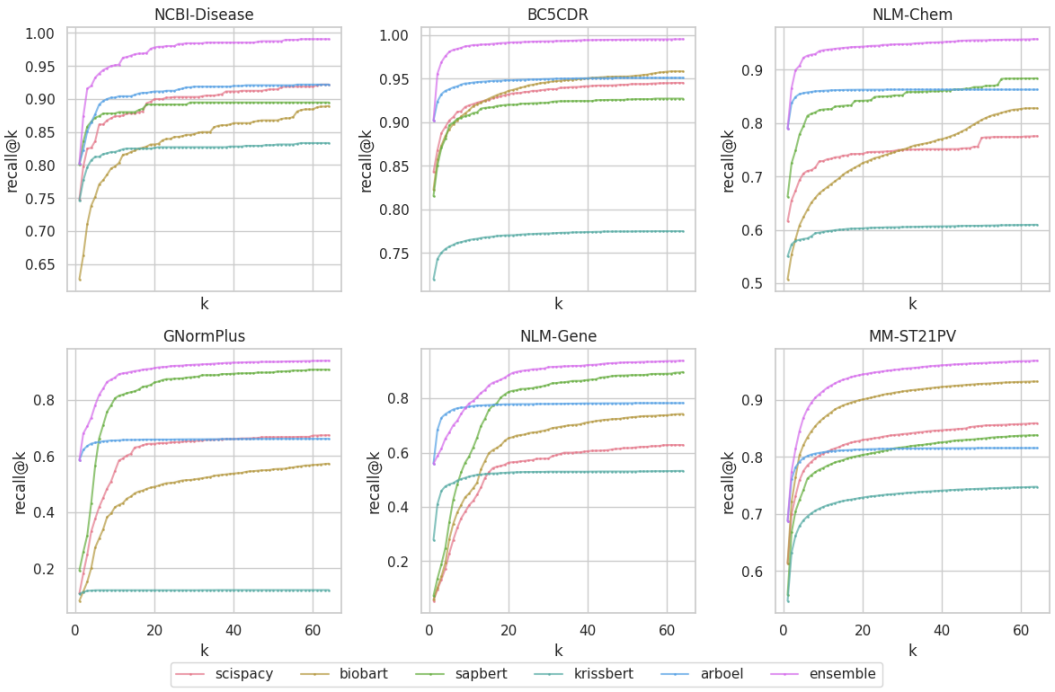


Figure 3: Comparison Recall@k for between all individual entity linking models and our ensemble method. Overall, the ensemble method consistently surpasses the top-performing individual model across datasets, highlighting its effectiveness in improving the candidate generation stage.

This appendix provides additional details and experimental results, and is organized as follows. All experiments reported in the appendix are based on a single run unless stated otherwise.

- Additional details on the datasets used in our study (Section A.1).
- Additional information on the entity linking model experiments, including hyperparameters and training time (Section A.2).
- Additional details on the LLM experiments, including prompts, parameters, and inference time (Section A.3).
- Additional experiments comparing our heuristic aggregation to more complex fusion strategies (Section A.4.1); analyses of the impact of ensemble size (Section A.4.2) and interleaving strategy (Section A.4.3), and the contribution of heterogeneous candidate generation to downstream disambiguation (Section A.4.4).

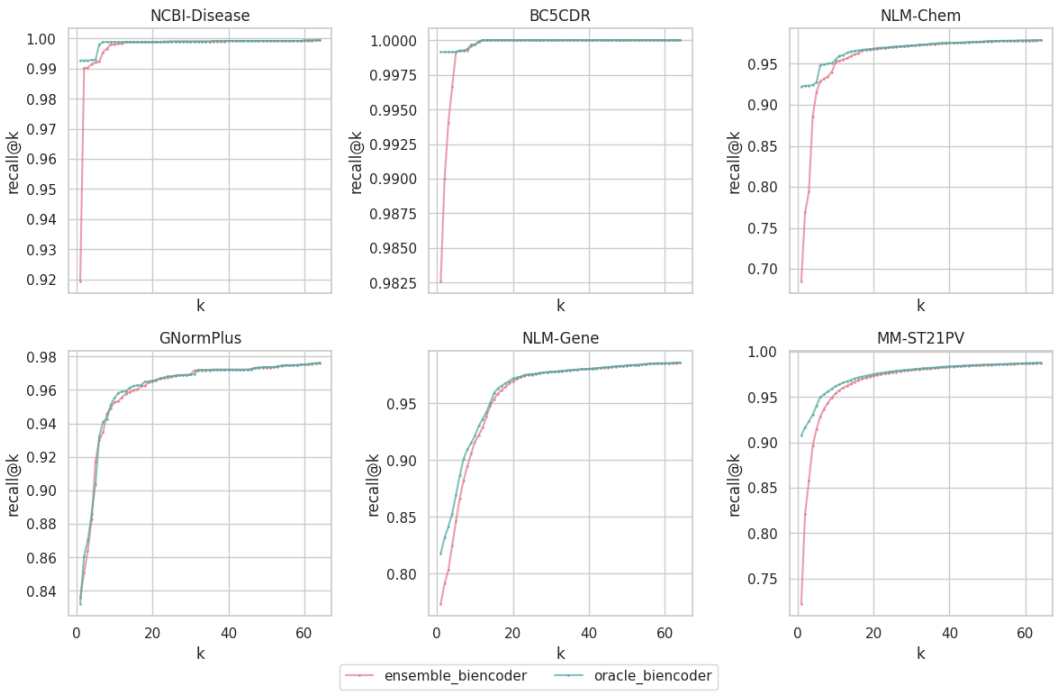


Figure 4: Training-set Recall@k of the ensemble biencoder (our heuristic configuration) and the oracle biencoder across all datasets. Accuracy on BC5CDR is nearly saturated, indicating that the candidate generation stage already retrieves the correct entity at the top position in almost all cases, leaving little room for the cross-encoder to learn meaningful re-ranking signals. This saturation effect is even more pronounced for the oracle biencoder, which also reaches near-saturation on the NCBI-Disease dataset.

- Additional robustness and error analyses, including robustness across random seeds (Section A.5.1), an analysis of unretrieved gold entities on the datasets with the lowest candidate-generation recall (Section A.5.2, an analysis of performance by training-set coverage (Section A.5.3) and the effect of fine-tuning (Section A.5.4).

A.1 Datasets

We provide detailed descriptions of the datasets used in our experiments below.

NCBI Disease (Dogan et al., 2014) is a widely used benchmark for disease entity linking. It contains 793 PubMed abstracts annotated with 6881 disease mentions, which are mapped to 790 unique concepts from MEDIC ontology, which integrates concepts from both MeSH (Lipscomb, 2000) and OMIM (Hamosh et al., 2005).

BC5CDR (Li et al., 2016) consists of 1,500 PubMed abstracts annotated for both chemical and disease entities. It includes 4,409 chemical mentions (BC5CDR-chem) and 5,818 disease mentions (BC5CDR-dis), with all annotations normalized to MeSH concepts based on the Comparative Toxicogenomics Database (CTD).

NLM Chem (Dogan et al., 2021a) is a gold-standard corpus for chemical entity linking that emphasizes diversity and full-text complexity. It contains chemical mentions annotated in 150 full-text articles from PubMed Central (PMC), with all mentions normalized to MeSH concepts.

GNormPlus (Wei et al., 2015) is a benchmark dataset for gene mention normalization comprising 694 PubMed abstracts. It includes the BioCreative II Gene Mention (BC2GM) corpus as a subset,

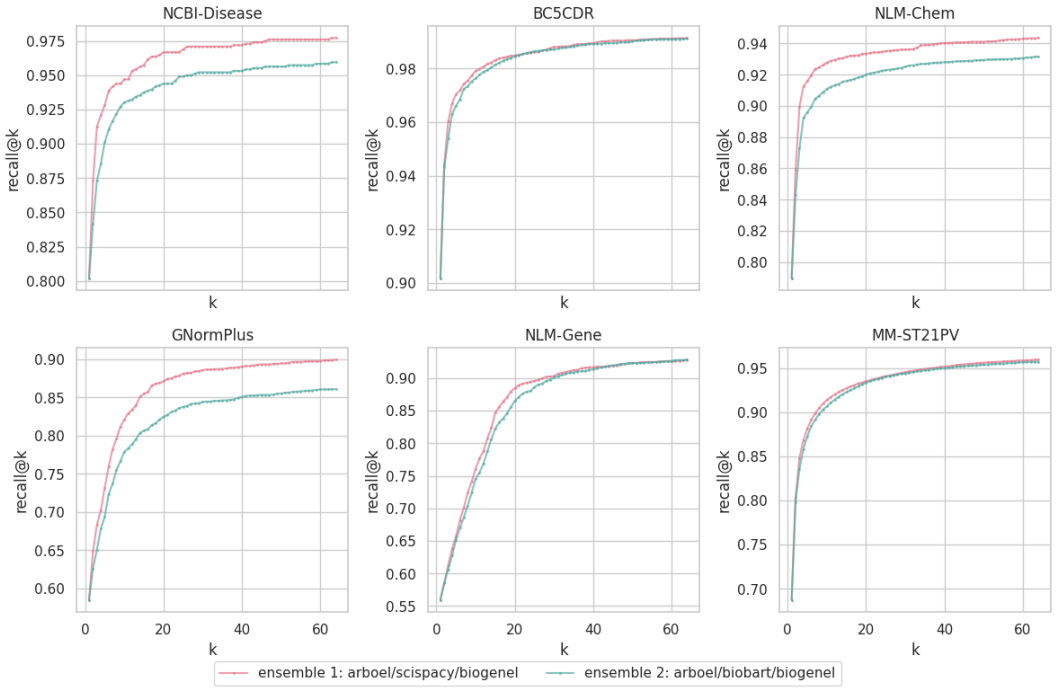


Figure 5: Recall@k comparison of two ensemble configurations with different levels of model diversity on the candidate generation stage. Ensemble 1, composed of architecturally diverse models (ArboEL, SciSpacy, and BioGenEL), consistently outperforms Ensemble 2, which includes the more similar pair BioBART and BioGenEL. These results demonstrate that greater model heterogeneity leads to more robust candidate set construction.

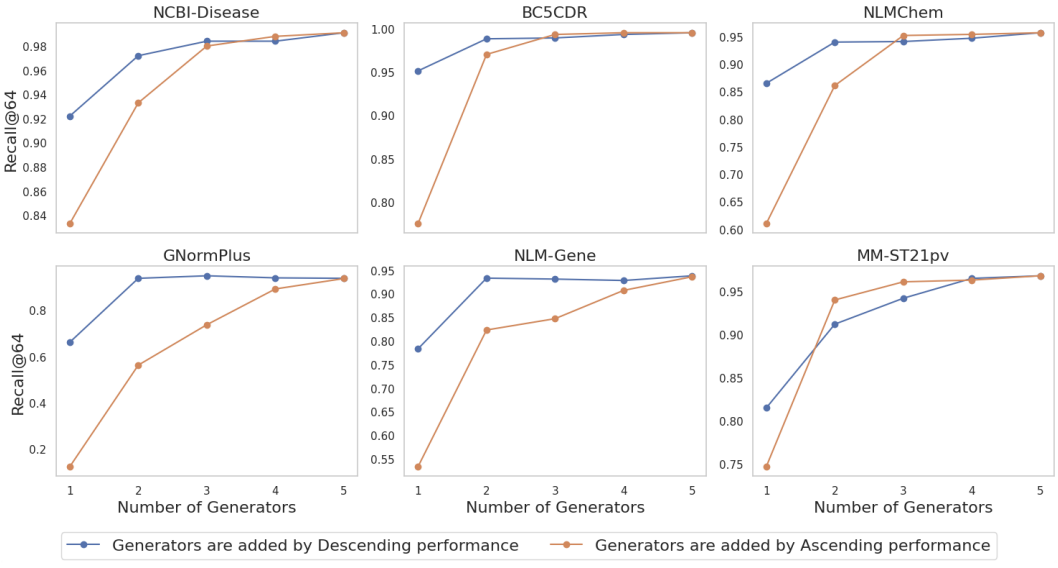


Figure 6: Increasing Generators Leads to Recall Saturation and Diminishing Returns

along with an additional 151 annotated abstracts. Gene mentions are linked to the Entrez Gene database (Maglott et al., 2005)

NLM Gene (Dogan et al., 2021b) is a large-scale corpus of over 500 full-text biomedical articles annotated with gene mentions. Similar to GNormPlus, entity normalization is performed against

Model	Dataset	Epochs	LR	Batch	Additional details
SapBERT	**NCBI-Disease**	1	1e−4	240	We initialize the model with "microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext" from `https://huggingface.co/microsoft/BiomedNLP-BiomedBERT-base-uncased-abstract-fulltext`. SapBERT is first pre-trained on UMLS, and subsequently fine-tuned on the ontology associated with each dataset. We use `loss=ms_loss` and `type_of_triplets=all`.
	BC5CDR	1	1e−4	240	
	NLM-Chem	1	1e−4	240	
	GNormPlus	1	1e−4	240	
	NLM-Gene	1	1e−4	240	
	MM-ST21PV	1	1e−4	240	
BioBART/ BioGenEL	**NCBI-Disease**	8	1e−5	8	For BioBART, we initialize the model with `GanjinZero/biobart-v2-base`. For BioGenEL, we initialize the model with the checkpoint released from bioGenEL paper (Yuan et al., 2022b) (`https://github.com/Yuanhy1997/GenBioEL/tree/main/src`). We set `max_steps=20000`, `lr_scheduler_type=polynomial`, We initialize the model with `max_length=384`.
	BC5CDR	8	1e−5	8	
	NLM-Chem	8	1e−5	8	
	GNormPlus	8	1e−5	8	
	NLM-Gene	8	1e−5	8	
	MM-ST21PV	8	1e−5	8	
ArboEL Biencoder	**NCBI-Disease**	10	3e−5	64	We initialize the model with "dmis-lab/biobert-base-cased-v1.2" from `https://huggingface.co/dmis-lab/biobert-base-cased-v1.2`. We use `embed_batch_size=3500`, `knn=10`, and `gradient_accumulation_steps=10`.
	BC5CDR	10	3e−5	64	
	NLM-Chem	10	3e−5	64	
	GNormPlus	9	3e−5	64	
	NLM-Gene	10	3e−5	64	
	MM-ST21PV	10	3e−5	64	
ArboEL Crossencoder	**NCBI-Disease**	2	2e−5	2	We initialize the model with "dmis-lab/biobert-base-cased-v1.2" from `https://huggingface.co/dmis-lab/biobert-base-cased-v1.2`
	BC5CDR	1	2e−5	2	
	NLM-Chem	5	2e−5	2	
	GNormPlus	1	2e−5	2	
	NLM-Gene	5	2e−5	2	
	MM-ST21PV	2	2e−5	2	

Table 10: Training hyperparameters used for each evaluated model across all datasets. Additional settings are reported when they are shared across all datasets for a given model.

the Entrez Gene database (Maglott et al., 2005). The use of full-text articles makes this dataset particularly challenging due to longer contexts and increased ambiguity.

MedMentions-ST21PV (Mohan & Li, 2019) is a curated subset of the original MedMentions dataset. It restricts entity annotations to 21 selected semantic type groups from the Unified Medical Language System (UMLS) (Bodenreider, 2004), focusing on more relevant biomedical concepts. The subset contains 4,392 PubMed abstracts with entity mentions linked to UMLS concepts, and is commonly used for large-scale biomedical entity linking evaluation.

A.2 More on Entity Linking Models

The training hyperparameters and implementation details for each entity linking model are summarized in Table 10. We first optimize hyperparameters on the smallest dataset NCBI-Disease, and reuse this configuration across other datasets. Adjustments are introduced only when the initial configuration fails to yield satisfactory performance. SciSpacy and KRISSBERT are excluded from the table as they do not require training.

Training and inference times for each model are reported in Figures 7 and 8 respectively. Among the evaluated methods, ArboEL exhibits the longest training time due to the need to construct a graph encompassing all entities from both the dataset and the reference ontology, followed by pruning operations. This process becomes particularly expensive for large-scale datasets. For instance, the MM-ST21PV dataset includes approximately 2.3 million entities in the ST21PV subset of UMLS and around 200k entities in the dataset itself, resulting in a biencoder training time of up to 15 days. In terms of inference, BioBART and BioGenEL are the most time-consuming due to their autoregressive nature. Their computational cost increases significantly with the number of mentions; for example, processing 40,000 mentions in the MM-ST21PV dataset requires approximately 3.5 days for BioGenEL.

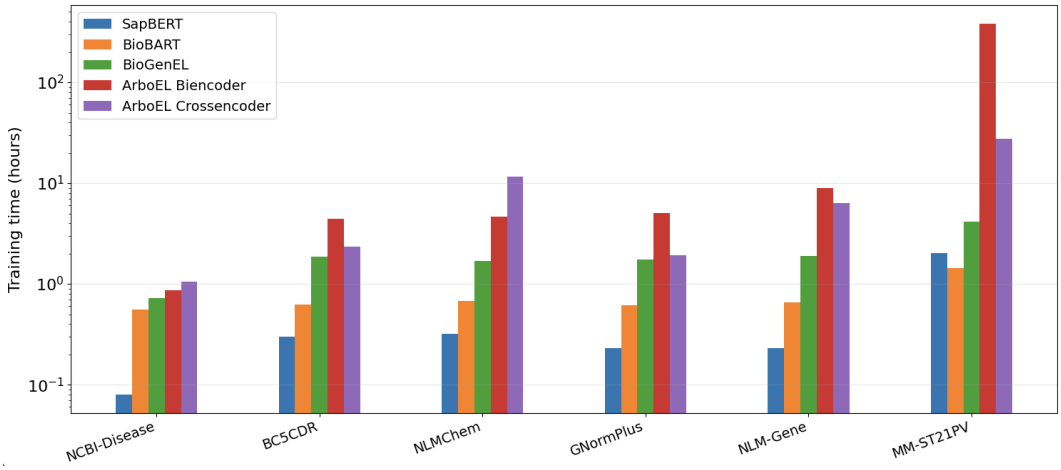


Figure 7: Training Time of Entity Linking Models Across Datasets (log scale). All experiments were run on a single Nvidia A40 GPU.

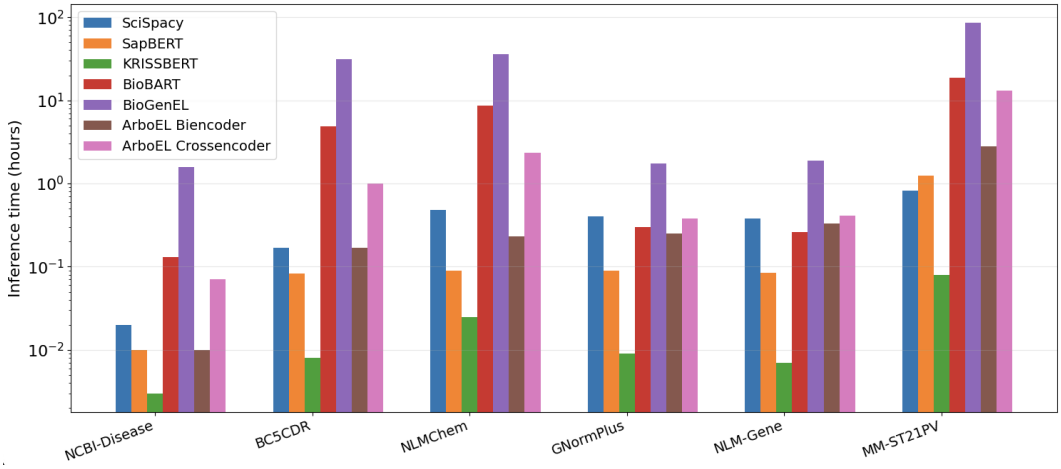


Figure 8: Inference Time of Entity Linking Models Across Datasets (log scale). All experiments were run on a single Nvidia A40 GPU.

A.2.1 Cross-encoder

The cross-encoder architecture of Wu et al. (2020) jointly encodes the mention context and each candidate entity $\boldsymbol{y}_{m,e}$. This enables fine-grained token-level interactions between the mention and candidate. The mention is represented together with its surrounding context as <[CLS] ctxt_l [M_s] mention [M_e] ctxt_r [SEP]>, while the candidate entity is represented using its title and description as <[CLS] title [ENT] description [SEP]>. Special tokens [M_s] and [M_e] are used to explicitly mark the mention span, while [ENT] structure the entity representation. This formulation allows the model to leverage cross-attention to directly compare the mention with each candidate entity, producing a relevance score for ranking. Specifically, candidate entities are scored by applying a linear transformation $\boldsymbol{W}$ to the embedding $\boldsymbol{y}_{m,e}$:

$$s_{\text{cross}}(m, e) = \boldsymbol{y}_{m,e}\boldsymbol{W}.$$

A.3 More on LLMs disambiguation experiment

For the LLM-based entity disambiguation experiments, we adopt the same prompt as proposed by Ye & Mitchell (2025) (see Figure 9). Elements enclosed in *{italics}* in the prompt correspond to input arguments passed to the LLM, defined as follows:

System Instruction
You are a professional data annotator and curator.
Your task is to identify the correct entity for a given mention based on the provided context and the descriptions of {*number_candidates*} candidate entities.

User
Here are a few examples:
{*topk_examples*}

This is the specific mention that needs to be linked to the correct entity:
{*mention*}

This is the context where the mention appears:
{*context*}

These are the candidate entities to choose from:
{*candidates*}

You MUST PROVIDE an ANSWER among the candidates
Use step-by-step reasoning but do not add provide any explanations to me. I only want the final answer.

Return the results in JSON format with just the CUI.
For instance {*example_answers*} are valid answers.

Figure 9: Prompt used by the LLM for the final linking. (Ye & Mitchell, 2025)

- `number_candidates`: Number of candidate entities provided to the model.
- `topk_examples`: In-context examples included in the prompt to guide the model. Similar to Ye & Mitchell (2025) study, we use 3 in-context examples.
- `mention`: The target mention to be linked.
- `context`: The textual context in which the mention appears.
- `candidates`: The information associated with each candidate entity.
- `example_answers`: Examples of valid output formats.

All LLMs were evaluated using greedy decoding with temperature set to 0 and thinking mode disabled. We also tested Qwen with thinking mode enabled using the recommended setting ($T = 1.0$), but observed no improvement on NCBI-Disease (0.861 vs. 0.864) despite substantially higher inference time (32 h vs. 1 h); we therefore disabled thinking mode in all experiments. Due to computational constraints, all our experiments with LLMs are based on a single run. However, to assess variability across runs, we repeated the evaluation of Qwen3.5-9B on the ensemble method in our main results (Section 4.2) five times on a subset of datasets. As shown in Table 11, we observe negligible variation across runs, indicating that the results are highly stable. The inference times for Qwen3.5-9B and Qwen3.5-27B models are reported in Table 12. Experiments were conducted on a single NVIDIA A40 GPU for Qwen3.5-9B and a single NVIDIA H100 GPU for Qwen3.5-27B, using the vLLM package (Kwon et al., 2023).

Dataset	Qwen3.5-9B (mean ± var)
NCBI-Disease	0.865 ± 6.09e-7
GNormPlus	0.645 ± 1.89e-7
NLM-Gene	0.511 ± 2.29e-6

Table 11: Performance stability of Qwen3.5-9B across 5 runs using greedy decoding (temperature = 0). Variance is extremely low indicating highly robust results.

In contrast to the work of Ye & Mitchell (2025), we extend both the context length and the candidate set size. Specifically, we use the full available context (rather than truncating to 64 words) and increase the number of candidates from 10 to 64. As shown in Table 13, these modifications consistently improve performance.

This discrepancy with the findings of Ye & Mitchell (2025) can be explained by two factors. First, our heterogeneous ensemble retrieval method produces a more robust candidate set, such that entities ranked between positions 11 and 64 still have a meaningful probability of being correct. Second,

Dataset	Qwen3.5-9B (A40)	Qwen3.5-27B (H100)
NCBI-Disease	1.53	0.97
BC5CDR	13.40	7.47
NLMChem	18.63	18.17
GNormPlus	1.90	1.65
NLM-Gene	1.84	2.22
MM-ST21PV	38.55	28.11

Table 12: Inference time (in hours) of Qwen3.5-9B on an A40 GPU and Qwen3.5-27B on an H100 GPU across datasets.

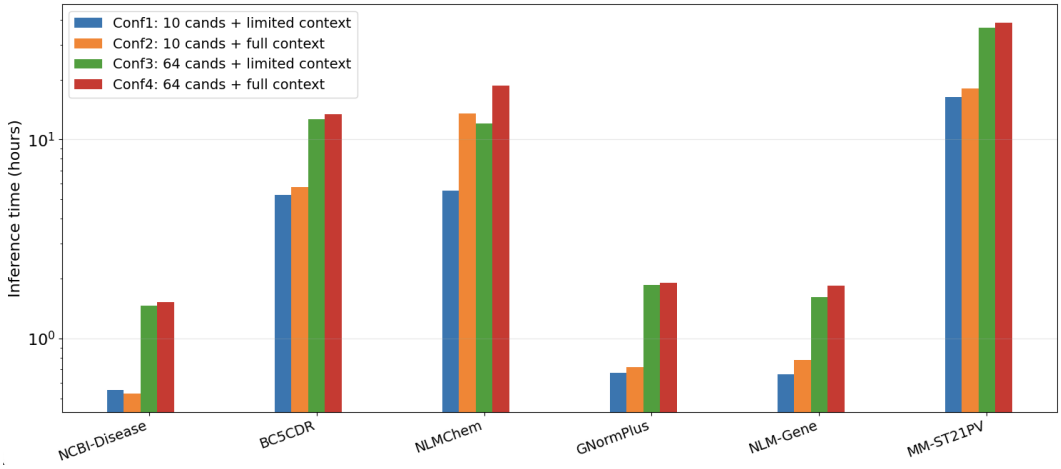


Figure 10: Inference time of Qwen3.5-9B under varying numbers of candidates and context lengths (log scale). Using 64 candidates with the full context can increase inference time by up to a factor of three. All experiments were run on a single Nvidia A40 GPU

we employ Qwen3.5-9B, which is better for handling longer contexts compared to the Qwen2.5-7B model used in their study. However, this advantage comes at the cost of increased inference time, as the context window grows substantially with the inclusion of more candidates and contextual information as shown in Figure 10.

Dataset	Top-10 candidates		Top-64 candidates	
	Limited context	Full context	Limited context	Full context
NCBI-Disease	0.847	0.858	0.859	**0.866**
BC5CDR	0.912	0.913	0.911	**0.918**
NLM-Chem	0.881	0.876	0.888	**0.889**
GNormPlus	0.612	0.618	0.631	**0.643**
NLM-Gene	0.490	0.493	0.510	**0.511**
MM-ST21PV	0.727	0.732	0.743	**0.748**
Avg.	0.745	0.748	0.757	**0.763**

Table 13: Accuracy of the Qwen3.5-9B model under varying numbers of candidate entities (Top-10 vs. Top-64) and context lengths (mention with 64 surrounding words vs. full context). Best results for each dataset are shown in **bold**. Overall, larger candidate sets and full-context inputs tend to yield improved performance.

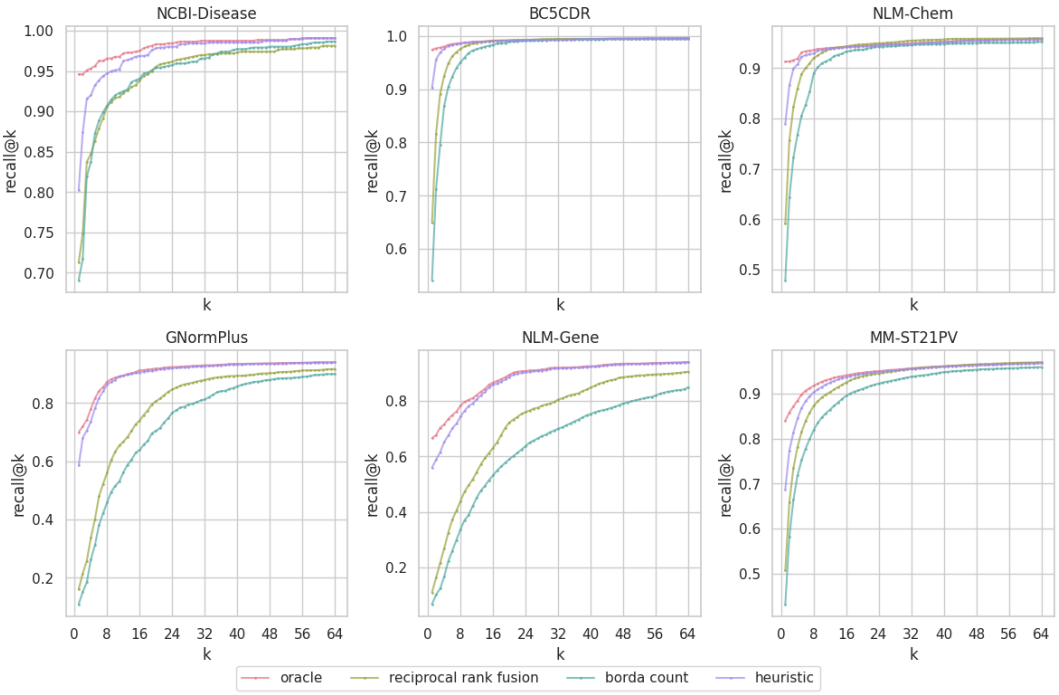


Figure 11: Candidate generator performance under different fusion strategies: Heuristic (round-robin aggregation based on generator performance), Oracle (best possible per-mention combination), Reciprocal Rank Fusion, and Borda count.

A.4 Additional Experiments

A.4.1 Experiments with different fusion strategies

To complement our main results, we evaluated alternative candidate aggregation strategies, namely Reciprocal Rank Fusion (RRF) and Borda count.

As shown in Figure 11, our simple heuristic ensemble consistently outperforms these two fusion methods across all datasets. This behavior can be understood by examining how each approach handles aggregations of candidates.

In practice, individual models can perform very poorly on certain mention with the correct entity being absent from their candidate lists. In such cases, all returned candidates from these models effectively act as noisy competitors. If the correct entity appears in the candidate list of only one model, it can be overwhelmed by the accumulation of incorrect candidates contributed by other models.

With the heuristic aggregation, this issue is mitigated: a candidate ranked highly by any single model is still promoted near the top, since the method effectively preserves strong individual signals without heavily penalizing disagreement across models.

In contrast, fusion methods such as RRF and Borda aggregate rankings across all models. This makes them sensitive to noise: candidates that appear frequently—even if incorrect—are systematically promoted, while the correct entity, supported by only one model, is diluted and pushed lower in the ranking.

As a result, recall@k consistently drops across all values of k and datasets. This effect is particularly pronounced in datasets like GNormPlus and NLM-Gene, where several models (e.g., KRISSBERT, SciSpacy, and BioBART) frequently fail to include the correct entity in their candidate lists, amplifying the noise introduced during fusion.

A.4.2 Impact of the Number of Models

Table 14 summarizes the impact of removing each model from the full ensemble on the disambiguation performance of Qwen3.5-9B. Overall, reducing the number of models consistently degrades performance on average, highlighting the importance of combining more models. Increasing the number of models improves linking performance by enhancing candidates quality. The full ensemble achieves the best average accuracy (76.3%), while removing individual models can lead to noticeable drops (0.4% to 3.9% decrease).

The largest performance decrease is observed when removing ArboEL, with the average accuracy dropping to 0.724. This is consistent with ArboEL being the strongest individual model, whose high-quality predictions play a dominant role in shaping the effectiveness of the aggregated candidate set.

Additionally, the impact of removing a model varies across datasets. For example, removing ArboEL leads to a substantial drop on GNormPlus and NLM-Gene, while its effect is smaller on datasets such as BC5CDR. Similarly, other models contribute differently depending on the dataset, indicating that each model provides complementary, dataset-specific benefits. The impact of removing one model on the candidate sets is shown in Figure 12.

Dataset	Full (Original)	w/o ArboEL	w/o BioBART	w/o SciSpacy	w/o KRISSBERT	w/o SapBERT
NCBI-Disease	0.866	**0.894**	0.865	0.873	0.863	0.864
BC5CDR	**0.918**	0.910	0.917	0.916	0.916	**0.918**
NLM-Chem	0.889	0.884	0.883	**0.892**	0.889	0.884
GNormPlus	**0.643**	0.523	0.641	0.620	0.640	0.619
NLM-Gene	**0.511**	0.411	0.506	0.509	0.503	0.509
MM-ST21PV	**0.748**	0.721	0.743	0.740	0.743	0.747
Avg.	**0.763**	0.724	0.759	0.758	0.759	0.757

Table 14: Sensitivity of Ensemble Performance to Model Composition. We analyze the effect of removing each model from the full ensemble on Qwen3.5-9B linking performance. "Full" denotes the complete ensemble, while "w/o X" indicates the exclusion of model X. Results are shown as accuracy across datasets, illustrating how each model contributes to the overall effectiveness and complementarity of the ensemble.

A.4.3 Effect of the Interleaving Strategy

In Section 4.1, our interleaved candidate construction prioritizes candidates from higher-performing models when building the ensemble list. However, it is unclear whether this design choice is even necessary, as section 4.5 shows that even sub-optimal candidate ordering can approach oracle-level performance. To investigate this, we analyze the impact of model ordering in the interleaved candidate list. Specifically, we investigate how the position of the correct entity influences disambiguation performance—i.e., whether placing the correct entity earlier is necessary, or whether its presence alone in the candidate set is sufficient for optimal performance.

To this end, we compare two configurations: (1) *best-first* ordering, where candidates are selected in a round-robin manner from models sorted by descending performance (i.e., stronger models contribute candidates earlier), and (2) *worst-first* ordering where candidates from weaker models are prioritized (i.e., KRISSBERT, BioBART, SciSpacy, SapBERT, ArboEL).

The results in Table 15 show that prioritizing candidates from stronger models yields a slight improvement in average performance. Although the gains are modest, they suggest that placing the correct entity earlier in the candidate list can marginally increase the likelihood that the disambiguation model—whether an LLM or a cross-encoder—selects it correctly. This observation is consistent with the findings from the oracle study. While this design choice is not critical, it is straightforward to implement and provides a measurable benefit. The impact of model ordering in the interleaved candidate list is shown in Figure 13

For completeness, we also investigate the effect of the interleaving block size used to construct the candidate list. The block size determines how many consecutive candidates are selected from each model before switching to the next. Smaller block sizes (e.g., 1) promote higher diversity by

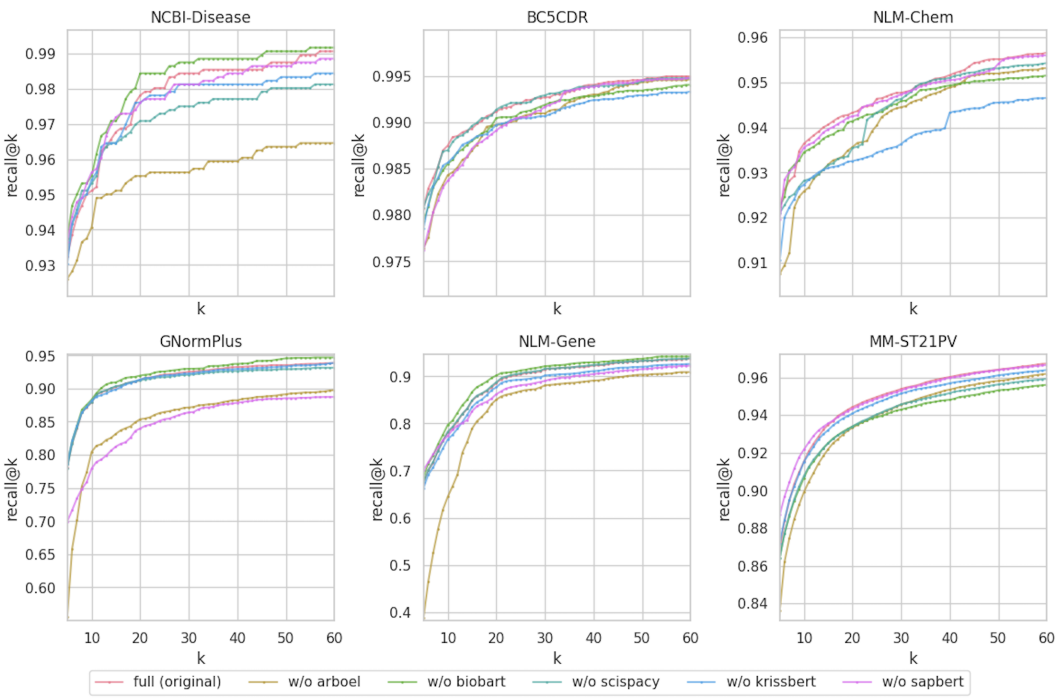


Figure 12: Sensitivity of Ensemble Performance to Model Composition. We analyze the effect of removing each model from the full ensemble on candidate generation performance for different recall@k ($k = 5$ to 60). "Full" denotes the complete ensemble, while "w/o X" indicates the exclusion of model X. Results are shown as Recall@k across datasets, illustrating how each model contributes to the overall effectiveness and complementarity of the ensemble.

Dataset	Cross-encoder		Qwen3.5-9B	
	Best-first (Original)	Worst-first	Best-first (Original)	Worst-first
NCBI-Disease	0.851	0.842 (-0.009)	0.866	0.865 (-0.001)
BC5CDR	0.896	0.892 (-0.004)	0.918	0.919 (+0.001)
NLM-Chem	0.853	0.838 (-0.015)	0.889	0.883 (-0.006)
GNormPlus	0.769	0.777 (+0.008)	0.643	0.656 (+0.013)
NLM-Gene	0.628	0.620 (-0.008)	0.511	0.490 (-0.021)
MM-ST21PV	0.740	0.738 (-0.002)	0.748	0.733 (-0.015)
Average	0.790	0.786 (-0.004)	0.763	0.758 (-0.005)

Table 15: Impact of model ordering on downstream disambiguation. **Best-first** prioritizes candidates generated by higher-performing models first, whereas **Worst-first** prioritizes those from lower-performing models. The (+) and (-) annotations denote the performance difference of the worst-first ordering relative to the original best-first configuration. Overall, the results show a very small but consistent advantage to prioritizing candidates from stronger models.

frequently alternating between models, while larger block sizes prioritize the top candidates from individual models.

As shown in Table 16, smaller block sizes tend to yield slightly better performance, likely due to a more balanced mix of candidates across models. However, the improvement is modest (around +0.2%), indicating that the overall impact is limited. Overall, while fine-grained interleaving is beneficial, the results suggest that block size is not a highly sensitive parameter and does not substantially affect performance. This observation is again consistent with the findings from the oracle study. The effect of model ordering in the interleaved candidate list is illustrated in Figure 14.

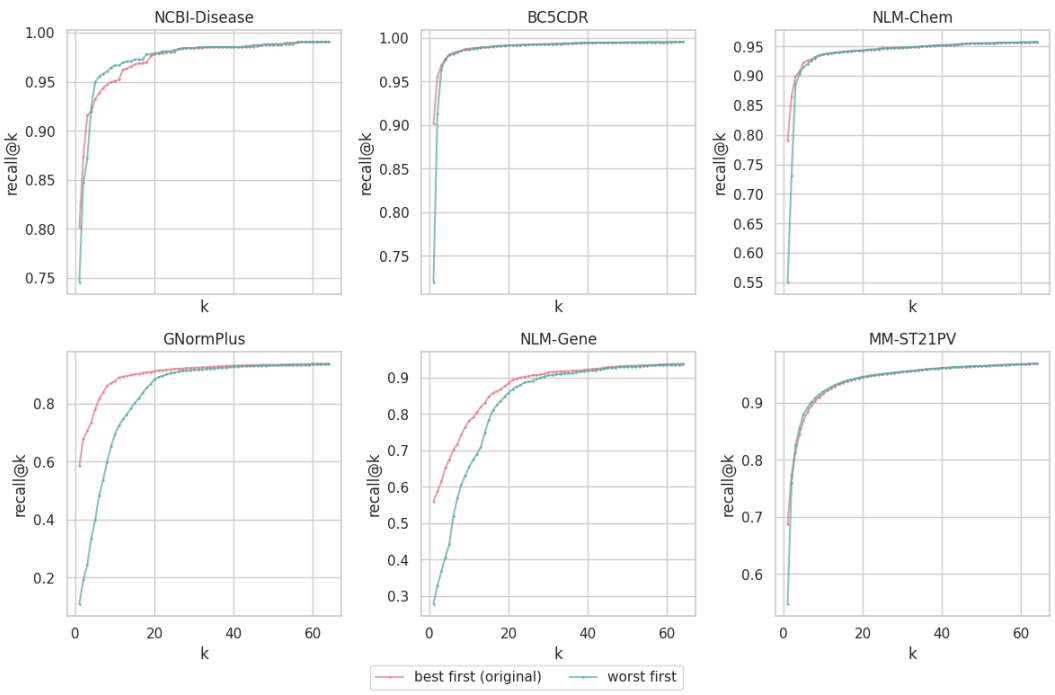


Figure 13: Impact of model ordering in the recall@k of the interleaved candidate list.

Dataset	b=1 (Original)	b=4	b=8	b=12
NCBI-Disease	0.866	0.866	0.867	0.865
BC5CDR	0.918	0.916	0.914	0.915
NLM-Chem	0.889	0.888	0.885	0.886
GNormPlus	0.643	0.643	0.648	0.646
NLM-Gene	0.511	0.509	0.514	0.513
MM-ST21PV	0.748	0.746	0.742	0.743
Avg.	0.763	0.761	0.761	0.761

Table 16: Effect of Interleaving Size on Qwen3.5-9B disambiguation performance. We vary the block interleaving size b, where b denotes the number of consecutive candidates selected from each model before switching to the next model (e.g., selecting the top b candidates from each model in a round-robin fashion).

A.4.4 The Importance of Ensembling for Effective Entity Disambiguation

The results in Table 3 showed substantial improvements over the best-performing evaluated model on each dataset. To examine whether these gains primarily arise from stronger disambiguation or from improved candidate generation, we conduct an ablation study that applies the same disambiguator to different candidate sets.

Table 17 compares Qwen3.5-27B using candidates generated by ArboEL and by our heterogeneous ensemble. With ArboEL candidates, Qwen3.5-27B improves by only +0.6 points on average over the best-performing evaluated model for each dataset and decreases performance on two datasets. Using the same disambiguator with the heterogeneous ensemble instead yields an average gain of +8.1 points, with improvements across all datasets.

These results highlight the critical role of candidate generation: the effectiveness of even a strong disambiguation model is fundamentally constrained by the quality and coverage of its candidate set, while improved candidate diversity through ensembling enables robust and consistent performance gains.

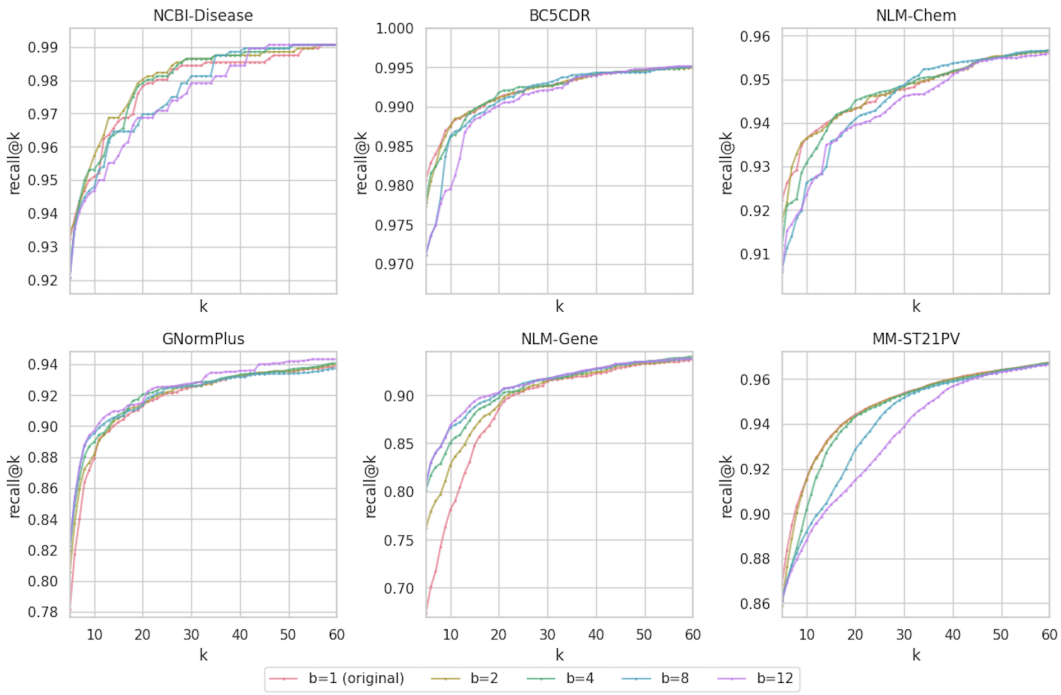


Figure 14: Effect of Interleaving Size on the Ensemble Candidate List.

Dataset	Best Evaluated Model	ArboEL	ArboEL + Qwen3.5-27B	Heterogeneous ensemble + Qwen3.5-27B
NCBI-Disease	0.833	0.802	0.839 (+0.006)	0.875 (+0.042)
BC5CDR	0.902	0.902	0.881 (-0.021)	0.918 (+0.016)
NLM-Chem	0.790	0.790	0.823 (+0.033)	0.900 (+0.110)
GNormPlus	0.585	0.585	0.623 (+0.038)	0.832 (+0.247)
NLM-Gene	0.560	0.560	0.574 (+0.014)	0.591 (+0.031)
MM-ST21PV	0.697	0.687	0.668 (-0.029)	0.739 (+0.042)
Avg.	0.728	0.721	0.734 (+0.006)	0.809 (+0.081)

Table 17: Ablation study comparing the impact of candidate generation on entity disambiguation performance. We apply Qwen3.5-27B to candidate sets produced by ArboEL and by our heterogeneous ensemble, and compare against the best-performing evaluated model on each dataset.

A.5 Robustness and Error Analysis

A.5.1 Robustness Across Random Seeds

Due to the computational cost of the full experimental pipeline, the main results are based on single runs. To assess the sensitivity of our key findings to random initialization and data ordering, we repeat the experiments corresponding to Tables 2, 3, and 7 on NCBI-Disease using two additional random seeds, for a total of three runs.

k	ArboEL	BioBART	Ensemble
1	0.808 ± 0.002	0.631 ± 0.001	—
10	0.906 ± 0.001	0.793 ± 0.001	$0.950 \pm 7 \times 10^{-4}$
64	$0.926 \pm 2 \times 10^{-4}$	0.885 ± 0.001	$0.990 \pm 2 \times 10^{-4}$

Table 18: Mean and standard deviation of Recall@k over three runs on NCBI-Disease. We report individual generators whose outputs varied across seeds; SapBERT produced identical results across all runs.

	Cross-encoder	Qwen3.5-9B	Qwen3.5-27B
Accuracy	$0.853 \pm 2 \times 10^{-4}$	$0.867 \pm 2 \times 10^{-5}$	$0.876 \pm 7 \times 10^{-5}$

Table 19: Mean and standard deviation of end-to-end accuracy over three runs on NCBI-Disease.

	Heuristic	Oracle
Cross-encoder	$0.853 \pm 2 \times 10^{-4}$	$0.853 \pm 7 \times 10^{-5}$
Qwen3.5-9B	$0.867 \pm 2 \times 10^{-5}$	$0.870 \pm 3 \times 10^{-4}$

Table 20: Mean and standard deviation over three runs when comparing heuristic and oracle candidate aggregation on NCBI-Disease.

Across these repeated experiments, variance is small relative to the main performance differences. The heterogeneous ensemble consistently maintains its candidate-recall improvements and end-to-end gains, while the heuristic aggregation remains close to the candidate-coverage oracle defined over the five evaluated generators. These results provide additional evidence that the main conclusions are robust on NCBI-Disease, while the remaining experiments should be interpreted as single-run estimates.

A.5.2 Analysis of Unretrieved Gold Entities

GNormPlus and NLM-Gene exhibit the lowest Recall@64 among the evaluated datasets (Table 2), and leaves a non-negligible set of 6.2% gold entities unrecovered despite combining five heterogeneous retrievers. To understand what limits further coverage gains, we analyze test mentions for which the gold entity is absent from the ensemble's top-64 candidate set. We identify 195 such mentions in GNormPlus and 168 in NLM-Gene.

A major source of error is the lexical mismatch between the mention and the canonical entity name or its aliases. This includes non-standard identifiers (e.g., *KIAA0039* $\rightarrow$ *POLD3*), paraphrastic references (e.g., *IGF system* $\rightarrow$ *IGFBP6*), and group-level mentions mapped to specific genes (e.g., *transcription factors* $\rightarrow$ *FOS*). These cases account for approximately 40% of the errors in GNormPlus and 30% in NLM-Gene.

A second source of difficulty is ambiguity in the Entrez Gene knowledge base. While names in MEDIC, MeSH, and UMLS are overwhelmingly associated with a single concept, Entrez contains substantially more shared names. In GNormPlus and NLM-Gene, 13.8% and 12.4% of KB names are ambiguous, respectively, and an exact mention match corresponds to a median of 4 and 8 gene records. For example, *p32* and *p51* can refer to 8 and 11 distinct genes. Such ambiguity accounts for approximately 25% and 30% of the remaining errors.

Finally, lower train-test entity overlap may further increase retrieval difficulty. Among test gold entities, 87.6% in GNormPlus and 44.5% in NLM-Gene do not appear in the corresponding training split, compared with 17–38% for the other datasets.

A.5.3 Analysis by Training-Set Coverage

To evaluate how our pipeline compares with existing models on entities unseen during training, we stratify test mentions according to whether their gold entity appears in the training split. This allows us to assess whether the observed gains persist in an entity-level zero-shot setting. Table 21 reports end-to-end accuracy for the heterogeneous ensemble with Qwen3.5-27B and for each individual model.

The full pipeline remains substantially more robust on OOV entities than the evaluated individual models. This effect is particularly pronounced on GNormPlus, where OOV entities constitute most of the test set: the ensemble with Qwen3.5-27B reaches 0.810 accuracy on the OOV subset, compared with 0.546 for ArboEL. Similar improvements are observed across the other OOV subsets. These results show that the proposed pipeline generalizes more effectively to entities unseen during training.

Dataset	Stratum	n	Ensemble + Qwen3.5-27B	ArboEL	SapBERT	SciSpacy	BioBART	KRISSBERT
NCBI-Disease	Seen	795	0.883	0.801	0.764	0.727	0.629	0.899
	OOV	165	0.800	0.455	0.570	0.527	0.267	0.000
BC5CDR	Seen	7,598	0.914	0.931	0.805	0.834	0.844	0.924
	OOV	2,152	0.927	0.800	0.851	0.875	0.745	0.000
NLM-Chem	Seen	7,260	0.934	0.851	0.704	0.521	0.576	0.881
	OOV	4,364	0.837	0.692	0.592	0.774	0.393	0.000
MedMentions	Seen	31,072	0.749	0.756	0.572	0.628	0.655	0.708
	OOV	9,071	0.704	0.457	0.509	0.560	0.471	0.000
GNormPlus	Seen	400	0.990	0.860	0.122	0.032	0.040	0.878
	OOV	2,822	0.810	0.546	0.201	0.120	0.090	0.000
NLM-Gene	Seen	1,514	0.655	0.643	0.038	0.036	0.040	0.502
	OOV	1,215	0.501	0.455	0.115	0.077	0.082	0.000

Table 21: Performance stratified by whether the gold entity appears in the training split. "Seen" denotes test mentions whose gold entity occurs during training, while "OOV" denotes gold entities unseen during training.

A.5.4 Fine-tuning LLMs further enhance the results

The primary focus of our study is improving the candidate generation stage through a heterogeneous ensemble approach. Beyond this, it is valuable to examine whether our method can also enhance LLM fine-tuning performance, given that it provides a higher-quality pool of candidates compared to prior work. To this end, we fine-tune Qwen3.5-4B using LoRA (Hu et al., 2021), with rank $r = 128$, scaling factor $\alpha = 32$, dropout $d = 0.05$, a learning rate of 5×10^{-4} and 5 training epochs. We select Qwen3.5-4B due to computational resource constraints. Our experiments are limited to NCBI-Disease, GNormPlus and NLM-Gene datasets as BC5CDR is already saturated (with training accuracy reaching 98%) while NLM-Chem and MM-ST21PV are prohibitively large.

Our results in Table 22 indicate that fine-tuning can further amplify the performance of our method. We observe no improvement on NCBI-Disease, likely because performance of the base model on the training set is already strong (87.9%), leaving limited room for further gains. The total fine-tuning time measured on a single NVIDIA H100 GPU is also reported in the Table.

Dataset	Qwen3.5-4B (Base)	Qwen3.5-4B (Fine-tuned)	Training Time (hrs)
NCBI-Disease	0.855	0.850 (-0.005)	59
GNormPlus	0.609	0.673 (+0.064)	11
NLM-Gene	0.460	0.509 (+0.049)	38

Table 22: Performance comparison between the base and fine-tuned versions of Qwen3.5-4B, along with fine-tuning time per dataset.