Better Generalizing to Unseen Concepts: An Evaluation Framework and An LLM-Based Auto-Labeled Pipeline for Biomedical Concept Recognition

Shanshan Liu[1,2], Noriki Nishida[1], Fei Cheng[3], Narumi Tokunaga[1],
Rumana Ferdous Munne[1], Yuki Yamagata[4, 5], Kouji Kozaki[6],
Takehito Utsuro[2], Yuji Matsumoto[1],
[1]RIKEN AIP [2]University of Tsukuba [3]Kyoto University [4]RIKEN R-IH
[5]RIKEN BRC [6]Osaka Electro-Communication University
{shanshan.liu, noriki.nishida, narumi.tokunaga, rumanaferdous.munne,
yuki.yamagata, yuji.matsumoto}@riken.jp
feicheng@i.kyoto-u.ac.jp; kozaki@osakac.ac.jp; utsuro@iit.tsukuba.ac.jp

Abstract

Generalization to unseen concepts is a central challenge due to the scarcity of human annotations in Mention-agnostic Biomedical Concept Recognition (MA-BCR). This work makes two key contributions to systematically address this issue. First, we propose an evaluation framework built on hierarchical concept indices and novel metrics to measure generalization. Second, we explore LLM-based Auto-Labeled Data (ALD) as a scalable resource, creating a task-specific pipeline for its generation. Our research unequivocally shows that while LLM-generated ALD cannot fully substitute for manual annotations, it is a valuable resource for improving generalization, successfully providing models with the broader coverage and structural knowledge needed to approach recognizing unseen concepts. Code and datasets are available at https://github.com/bio-ie-tool/hi-ald.

1 Introduction

Biomedical Concept Recognition (BCR) —identifying ontology concepts expressed in free-text passages—is foundational for knowledge-intensive applications. Accurate and efficient CR systems may facilitate the construction and maintenance of structured biomedical knowledge bases, accelerate knowledge discovery, and ultimately support downstream applications such as therapeutic innovation.

Most prior works formulate the BCR task as mention-based recognition: detect a text span (a "mention") in the text and then link it to an ontology concept (Li et al., 2016; Luo et al., 2021; Wang et al., 2023; Caufield et al., 2024; Groza et al., 2024). While being effective when concepts are explicitly expressed, this setting is less aligned with biomedical discourse, where many concepts are expressed implicitly. This misalignment persists even when mentions are annotated.

For instance, in an abstract included by the HPO GSC+ corpus (Lobo et al., 2017), the span *"melanin pigment synthesis in the hair, skin, and eyes"* is annotated as *Generalized hypopigmentation (HP:0007513)*. The concept is implicit with respect to that span—it is neither the ontology label nor a semantically equivalent paraphrase—and the annotation is supported by intra-abstract cues (e.g., hypopigmentation, reduced retinal pigment) rather than surface matching.

In addition, mention-based supervision requires costly two-level annotation (mention spans and mention-concept mappings), limiting scalability in domains where expert labeling is expensive. We therefore study **MA-BCR**, which directly identifies ontology concepts from a passage without requiring intermediate mention spans. This formulation better matches biomedical discourse and alleviates annotation burden (no need for mention spans for model training and inference).

Independent of the task formulation, a key requirement for real-world deployment of recognizers is the ability to **generalize to unseen concepts**, given that manually labeled datasets (MLDs) cover only a small fraction of concepts in biomedical ontologies due to the high requirements on expertise and annotation time. For example, the HPO GSC+ corpus comprises 228 abstracts and covers approximately 2.4% of Human Phenotype Abnormality (HPA) concepts in the Human Phenotype Ontology (HPO) (Gargano, 2023). Despite its importance, this capability has been under-evaluated. Beyond results showing that recognizing unseen concepts was nearly infeasible for a recognizer MA-COIR (Liu et al., 2025), **clear experimental evidence on unseen-concept recognition remains scarce**.

To systematically probe model generalization in recognizing unseen concepts—where models cannot predict the target directly—we propose an innovative evaluation framework. This framework is built upon two core components. First, we develop three types of hierarchical indices (ontology-aware, semantic-based, and hybrid), while each of them

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 3718–3734
March 24-29, 2026 ©2026 Association for Computational Linguistics

can serve as a structured concept space to make generalization measurable. Second, we introduce two new metrics that operate within this space: Unseen Recall-oriented Closeness (U-RC), which quantifies how close a model gets to the unseen concept, and Unseen Candidate set Size (U-CS), which measures how much the model shrinks the search space compared to the full label sets.

Our evaluation framework provides the tools to measure generalization, yet the question of how to improve it remains. A practical pathway is to expand concept coverage and contextual diversity through Auto-Labeled Data (ALD). Given their success as automatic annotation providers in other NLP tasks (Tan et al., 2024), Large Language Models (LLMs) present a scalable avenue for higher concept coverage. This potential is especially important for MA-BCR, a task that requires good language understanding capabilities for assigning passage-level labels without relying on surface mentions. We therefore design a novel auto-labeling pipeline based on LLMs. Its components are tailored for MA-BCR, and the final design is determined through extensive empirical validation. We employ our evaluation framework to assess this data-centric approach, specifically asking:

- *RQ1: Is LLM-generated ALD sufficient for training a recognizer that performs effectively on both seen and unseen concepts?*
- *RQ2: Despite the noise, does ALD improve generalization to unseen concepts?*

Our investigation proceeds as follows. We first quantify the quality of our ALD by benchmarking it against Manual-Labeled Data (MLD). We then employ the MA-COIR recognizer, which outputs pre-defined hierarchical concept indices as predictions, allowing for a grounded assessment of generalization. Finally, by training MA-COIR on different scales of MLD and ALD, we compare their effectiveness in enhancing model performance.

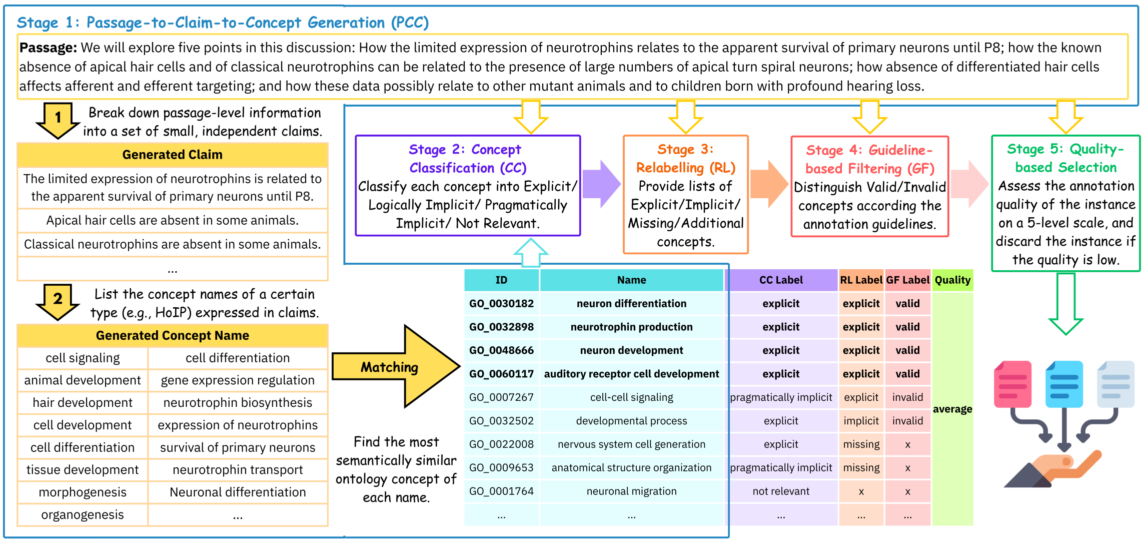
On two pairs of concepts and ontologies, HPA concepts in HPO and Homeostasis Imbalance Process (HoIP) concepts in HoIP Ontology (Yamagata et al., 2024), models trained on ALD consistently achieve substantial gains in unseen-oriented metrics as the data volume increases, despite exhibiting lower exact-matching accuracy than their MLD-trained counterparts. This indicates that exposure to broad and noisy supervision enhances hierarchy-aware generalization even when exact-match accuracy lags. We position LLM-based auto-labeling not as a replacement for manually labeling, but as a complementary source of coverage and structural signal for unseen concept recognition. Our contributions are as follows:

- We introduce an LLM-based auto-labeling pipeline for MA-BCR, and construct two large auto-labeled datasets. To our knowledge, LLM-based automatic labeling has not been explored in MA-BCR before our work.
- We design three types of hierarchical indices and two metrics for directly assessing hierarchy-aware learning and search-space reduction on unseen concepts.
- Results show that models trained on large-scale ALD underperform MLD-trained models on F1 but generalize better to unseen concepts, clarifying ALD's role in MA-BCR.

2 Related Works

Biomedical Concept Recognition. Biomedical concept recognition has traditionally been approached as a two-stage task: mention detection followed by entity linking. Early systems leverage neural architectures for named entity recognition and linking (Li et al., 2016; Luo et al., 2021), while more recent methods introduce LLM-based pipeline (Wang et al., 2023; Caufield et al., 2024; Groza et al., 2024; El Khettari et al., 2024) or reformat that task as a one-step sequence-to-sequence generation framework (Liu et al., 2025). However, these approaches often struggle with concept ambiguity, limited coverage, and implicit expressions—issues exacerbated by the sparsity of high-quality annotated data.

Hierarchical Indexing. Hierarchical indexing has proven effective in tasks involving large output spaces, such as extreme multi-label classification (Zhang et al., 2021; Kharbanda et al., 2022), document retrieval (Tay et al., 2022), and biomedical concept recognition (BCR) (Liu et al., 2025). By organizing labels or documents into trees based on semantic information through K-Means clustering, these methods improve efficiency and precision. Despite its promise, hierarchical indexing remains under-explored in BCR, where ontologies naturally provide structured concept taxonomies.



Stage 1: Passage-to-Claim-to-Concept Generation (PCC)

Passage: We will explore five points in this discussion: How the limited expression of neurotrophins relates to the apparent survival of primary neurons until P8; how the known absence of apical hair cells and of classical neurotrophins can be related to the presence of large numbers of apical turn spiral neurons; how absence of differentiated hair cells affects afferent and efferent targeting; and how these data possibly relate to other mutant animals and to children born with profound hearing loss.

1 Break down passage-level information into a set of small, independent claims.

Generated Claim
The limited expression of neurotrophins is related to the apparent survival of primary neurons until P8.
Apical hair cells are absent in some animals.
Classical neurotrophins are absent in some animals.
...

2 List the concept names of a certain type (e.g., HoIP) expressed in claims.

Generated Concept Name	
cell signaling	cell differentiation
animal development	gene expression regulation
hair development	neurotrophin biosynthesis
cell development	expression of neurotrophins
cell differentiation	survival of primary neurons
tissue development	neurotrophin transport
morphogenesis	Neuronal differentiation
organogenesis	...

Matching

Find the most semantically similar ontology concept of each name.

Stage 2: Concept Classification (CC) Classify each concept into Explicit/ Logically Implicit/ Pragmatically Implicit/ Not Relevant.

Stage 3: Relabelling (RL) Provide lists of Explicit/Implicit/ Missing/Additional concepts.

Stage 4: Guideline-based Filtering (GF) Distinguish Valid/Invalid concepts according the annotation guidelines.

Stage 5: Quality-based Selection Assess the annotation quality of the instance on a 5-level scale, and discard the instance if the quality is low.

ID	Name	CC Label	RL Label	GF Label	Quality
GO_0030182	**neuron differentiation**	**explicit**	**explicit**	**valid**	average
GO_0032898	**neurotrophin production**	**explicit**	**explicit**	**valid**	
GO_0048666	**neuron development**	**explicit**	**explicit**	**valid**	
GO_0060117	**auditory receptor cell development**	**explicit**	**explicit**	**valid**	
GO_0007267	cell-cell signaling	pragmatically implicit	explicit	invalid	
GO_0032502	developmental process	explicit	implicit	invalid	
GO_0022008	nervous system cell generation	explicit	missing	x	
GO_0009653	anatomical structure organization	pragmatically implicit	missing	x	
GO_0001764	neuronal migration	not relevant	x	x	
...	...	...	...	...	

Figure 1: Overview of the LLM-based Auto-labeling Pipeline. Given an input passage, the pipeline begins by generating intermediate claims (Arrow 1), followed by generating candidate concept names from the claims (Arrow 2). The resulting names are matched to ontology terms by comparing the representations (encoded by SapBERT) of generated and ontological names, forming a preliminary list of concept candidates, shown in the blue-shaded table. Then concept classification, relabeling, and guideline-based filtering steps are applied to get the final annotations (highlighted in bold). The instance will be taken as a training instance if its quality meets the requirement.

3 Methodology

3.1 Task formulation

Mention-agnostic Biomedical Concept Recognition (MA-BCR) aims to directly identify a subset of ontology-defined concepts $\{C'_1, ..., C'_p\} \subseteq O$ that are referenced in a given text Q, while the ontology $O = \{C_1, ..., C_n\}$ is constructed by domain experts. We frame this as an end-to-end generative task: the model directly produces a sequence of unique concept identifiers $\{I_{C'_1}, ..., I_{C'_p}\}$ for Q, with each I_{C_i} corresponding to a concept C_i in O.

MA-BCR diverges fundamentally from conventional NER+EL pipelines for BCR. It is specifically designed to capture not only **explicit** concepts—those directly realized in the text via surface forms or synonyms—but also **implicit** concepts. The latter are referenced through logical implications, domain-specific inferences, or unspoken premises, making them unlikely to appear verbatim or as simple paraphrases.

3.2 LLM-based Auto-Labeling Pipeline

LLMs offer a powerful foundation for automatic biomedical concept annotation. Trained on vast textual corpora, LLMs encode extensive biomedical knowledge and can infer relevant concepts even when they are not explicitly mentioned. Furthermore, LLMs are highly adaptable: with prompt-based control, a single model can function as a generator, classifier, filter, or evaluator, enabling the construction of a flexible, modular pipeline.

To figure out **what label quality can LLM-based auto-labeling achieve for MA-BCR**, we carefully design an LLM-based pipeline with five stages (Figure 1). Each stage is described in the following paragraphs. Full implementation details and inference costs are provided in Appendix A.1.

Stage 1: Passage-to-Claim-to-Concept Generation. Direct generation of concept names from passages often yields low recall and limited relevance, particularly for complex biomedical concepts (e.g., ∼30% recall on HoIP-MLD). To address this, we introduce an intermediate *claim generation* step, in which key biomedical assertions are summarized into concise claims (Figure 1-Arrow 1). These claims serve as more focused inputs for concept derivation, substantially improving both coverage and grounding. On HoIP-MLD, this step raises recall from ∼30% to ∼62%.

Concept names generated from claims are subsequently mapped to ontology labels via semantic matching (Figure 1-Arrow 2). We encode both the generated names and all ontology concept names (including synonyms) into 768-dimensional vectors using mean-pooled SapBERT embeddings.[1] Candidate matches are retrieved with Faiss (Douze

[1] Model card: cambridgeltl/SapBERT-from-PubMedBERT-fulltext

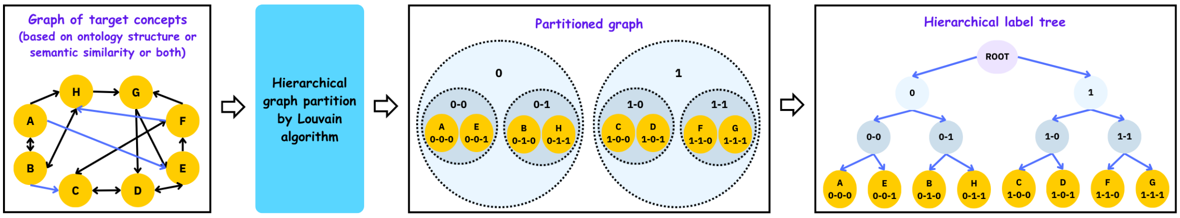


Figure 2: Overview of hierarchical search index construction. Ontology concepts are represented as nodes in a graph (yellow), where edges reflect either ontological relations (blue), semantic similarities (black), or both. A graph partitioning algorithm (e.g., Louvain) is then applied to recursively divide the graph into nested subgraphs. In the illustrated example, the initial graph is first partitioned into two coarse-level clusters (labeled 0 and 1), which are further subdivided into finer clusters (e.g., 0-0, 0-1, 1-0, 1-1). Based on the partitioned structure, a hierarchical label tree is constructed: internal nodes (gray) represent clusters, while leaf nodes (yellow) correspond to individual ontology concepts. Each concept is assigned an index reflecting its position in the tree (e.g., concept A is labeled 0-0-0, indicating its membership in cluster 0, then 0-0, and finally its position).

et al., 2024), using L2 distance as implemented by default. We select the top-1 match for each generated name if the similarity score is ≥ 0.6, forming a preliminary candidate list (illustrated in the blue-shaded table in Figure 1).

Stage 2: Concept Classification. The preliminary candidate list often contains both relevant and irrelevant concepts. This arises for two reasons: (i) LLM may generate irrelevant concepts due to its limited capability, and (ii) similarity-based matching may link a name to semantically close but irrelevant terms. To resolve this, we formulate a filtering process as a four-way classification problem. Each candidate is categorized as: (1) *Explicit*, (2) *Logically Implicit*: concepts required by the passage's logical structure, e.g., necessary premises, (3) *Pragmatically Implicit*: concepts plausibly inferred from domain knowledge or text content, though not logically required, or (4) *Not Relevant*.

This finer-grained categorization serves two purposes. First, it improves filtering accuracy: preliminary experiments show that this four-class classification outperforms both binary (relevant/not relevant) and three-class (explicit/implicit/not relevant) settings in distinguishing relevant concepts. Second, it enables flexible downstream usage: applications prioritizing precision may retain only explicit and logically implicit concepts, while discovery-oriented tasks can also benefit from pragmatically implicit ones. In our experiments, concepts classified as "Not Relevant" are discarded.

Stage 3: Relabeling. Even after Concept Classification, candidate concepts can still include false positives and false negatives. Residual false positives mainly stem from misclassifications in the Concept Classification stage, whereas residual false negatives arise from omissions during the Passage-to-Claim-to-Concept generation or from overly aggressive filtering in Concept Classification.

To address these issues, we employ an LLM to evaluate each passage–concept pair. For each sample, the LLM generates a justification highlighting strengths (concepts present) and weaknesses (concepts missing), and categorizes the candidate as explicit, implicit, or missing. It also identifies additional relevant concepts mentioned in the passage but absent from the candidate list. This feedback is then used to add missing concepts and remove incorrect ones, producing a more accurate and comprehensive set of annotations.

Stage 4: Guideline-based Filtering. Automatically annotated concepts may not fully align with the standards established by manual annotation. To ensure consistency and compliance with official criteria, we leverage existing annotation guidelines.

For HOIP concepts, the CRAFT corpus provides a 47-page guideline with over 100 examples and 50 figures. We refined the guideline by removing figures and reorganizing textual content, then used an LLM to generate a concise summary (2,128 characters). This summary is incorporated into the prompt that directs the LLM to review each passage and its candidate concept list, classifying each concept as valid or invalid according to the summarized guideline.

Stage 5: Quality-based Selection. Even after relabeling, annotation quality can vary across passages, and low-quality samples (passage-concept pairs) risk introducing noise into downstream applications. To mitigate this, we implement a quality-based selection step that filters at the passage level.

In this stage, each sample is evaluated by an

Case	Ontology		MLD			ALD		
	Uni.Con	Name	Passage	Concept	Uni.Con (Coverage)	Passage	Concept	Uni.Con (Coverage)
HPA	18,354	40,341	228	1,423	431 (2.35%)	54,301	197,824	12,725 (69.33%)
HoIP	29,367	117,072	3,621	7,855	690 (2.35%)	34,097	370,672	15,976 (54.40%)

Table 1: Dataset Statistics. Manually labeled data (MLD) are sourced from HPO-GSC+ and CRAFT; automatically labeled data (ALD) are generated via our auto-labeling pipeline. "Uni.Con" denotes the number of unique concepts. "Coverage" refers to the percentage of ontology concepts presented in the dataset.

LLM according to a five-tier rating scheme: (1) *Very Poor*: majority of concepts missing or irrelevant, (2) *Poor*: partial concept inclusion but lacking relevance, (3) *Average*: Most concepts present with basic contextualization, (4) *Good*: Comprehensive concept coverage with coherent context, (5) *Excellent*: full concept integration with clear explanations. In our experiments, only samples rated "Average" or above are retained.

Unlike earlier stages, which operate at the concept level, this stage evaluates the overall reliability of each sample. Passages dominated by irrelevant or incomplete annotations are discarded, ensuring dataset-level consistency, reducing the risk of noisy samples propagating into downstream applications.

3.3 Hierarchical Index Design

Traditional indexing in large label spaces relies on semantic similarity alone. In BCR, however, structured ontologies provide a second signal—explicit parent-child relations encoding curated expert knowledge. We explore both signals to construct more meaningful and scalable indices.

Following existing practices (Liu et al., 2025), we enforce a hierarchical structure of label tree where each concept is a leaf node (as shown in Figure 2). Internal nodes are limited to at most 10 children, striking a balance between decomposition granularity and model tractability. To handle both tree-like (HPO) and graph-like (HoIP) ontologies, we construct indices using graph-based methods.

We experiment with three graph construction strategies, resulting in three types of indices:

(1) **OSI (Ontology-aware Search Index)**: Edges follow ontology parent-child links with fixed weights of 1.0.

(2) **SSI (Semantic Search Index)**: Edges connect top-10 nearest embedding neighbors; weights reflect semantic similarity range from 0.0-1.0.[2]

(3) **OSSI (Ontology-Semantic Search Index)**: A hybrid combining OSI and SSI, prioritizing ontology structure while integrating semantic proximity.

We then do hierarchical graph partitioning on the graph of concepts, which yields trees that encode ontology structure or preserve semantic coherence.[3] **Corresponding indices help us to understand what type of hierarchical information has been learned by the recognizer.** Comparing SSI with ssID provided by Liu et al. (2025), the biggest difference is that we use graph partitioning for clustering, while they applied K-means method.

4 Experiment Design

4.1 Datasets

Target Concepts. Human Phenotypic Abnormality (HPA) concepts are defined as descendants of `HP:0000118-Phenotypic Abnormality` in HPO. Homeostasis Imbalance Process (HoIP) concepts correspond to Homeostasis Imbalance Process terms included in the HoIP Ontology.

Manual-Labeled Datasets (MLDs). We evaluate on two manually annotated corpora: (1) HPA-MLD, from the HPO GSC+ corpus, and (2) HoIP-MLD, adapted from the CRAFT corpus. Statistics are in Table 1. Examples are in Appendix A.2.

Auto-Labeled datasets (ALDs). To evaluate the scalability and effectiveness of our auto-labeling pipeline, we construct two large-scale ALDs for HPA and HoIP concepts, respectively. For each target concept, we retrieve up to 10 recent PubMed abstracts using the concept's preferred label as a query, via the NIH E-utilities API.[4] Duplicates are removed to ensure passage uniqueness. These passages are then annotated using our auto-labeling pipeline to produce two ALDs for downstream evaluation. Dataset statistics are reported in Table 1.

Data Splits. For auto-labeling evaluation (Table 2), we apply the ALD pipeline to 228 HPA-MLD and a randomly sampled 500-instance subset

[2] Weights are calculated as the same way for generated name-ontology concept matching in Section 3.2, Stage 1.

[3] Details are in Appendix A.4.

[4] `https://www.ncbi.nlm.nih.gov/books/NBK25500/`

of HoIP-MLD. Manual labels are used as gold standard to evaluate how each module impacts quality.

For recognizer training and evaluation (Table 3), we split the data as follows. For **HPA**, we use 45 MLD instances (20%) for development and 183 for testing. Due to the small size of HPA-MLD, we train models only on HPA-ALD (max to 47,152 instances). For **HoIP**, we adopt the CRAFT dev split (375 instances), and evaluate in two training setups: (1) training on HoIP-MLD (max to 2,458 instances) and testing on 375 sampled MLD instances (MLD→MLD); (2) training on HoIP-ALD (max to 16,415 instances) and testing on 3,246 held-out HoIP-MLD instances (ALD→MLD).

To assess generalization, we ensure that at least 30% of test concepts are unseen during training. This is achieved by filtering out training passages containing those concepts. We vary the training size $M = 200 \cdot 2^k,\ k = 0, \ldots, 7$. to test performance scalability. To maintain stable model selection, we always include 100 core training passages with concept distributions closest to the dev set. Additional $(M - 100)$ training instances are sampled randomly. Statistics are provided in Table 7.

4.2 Models

Models for ALD construction. To keep the auto-labeling pipeline scalable and accessible, we constrain LLMs to either (i) models runnable on a single consumer-grade GPU or (ii) cost-effective commercial APIs. After extensive testing across GPT and LLaMA variants, we adopt LLaMA-3-8B[5] for claim and concept generation due to its superior fluency, specificity, and domain grounding in both HPA and HoIP terms. For downstream tasks such as classification, relabeling, and filtering, we use GPT-4o-mini, balancing high reasoning ability with cost efficiency. This setup ensures useful annotations without sacrificing accessibility. More details are in Appendix A.1.

Concept recognizer. We adopt MA-COIR (Liu et al., 2025)—a BART-based seq2seq recognizer that outputs ontology concept indices from free text—as our canonical mention-agnostic model. This choice is methodological: an index-producing recognizer lets us (i) make hierarchical structure explicit to learn and (ii) quantify unseen behavior with hierarchy-aware metrics easily.

We keep the architecture and training recipe[6] fixed and swap MA-COIR's original semantic index for three hierarchical variants: an **ontology-structured** index (OSI), a **semantic-structured** index (SSI), and a **hybrid ontology–semantic** index (OSSI). This design enables a clean assessment of whether structural signals affect generalization.

Our goal is not to propose a new model architecture nor to compete for state-of-the-art. Cross-architecture baselines (often mention-based) are incapable of learning with a designed hierarchy and are orthogonal to our research questions. Prior work has already positioned MA-COIR as a strong MA-BCR approach; our SSI follows a similar design and serves as a strong baseline. Accordingly, we **do not report** comparisons to other models, nor do we re-run the original MA-COIR indexing.

4.3 Evaluation Metrics

We evaluate from three complementary views: (i) exact-match accuracy, (ii) hierarchy-aware closeness to unseen concepts, and (iii) search space of unseen concepts inferable by predictions.

For a passage p, let $Y(p)$ be its gold concept set and $\hat{Y}(p)$ be the set produced by the recognizer. Each concept x is represented by an index sequence $I_x = [i_1, \ldots, i_{|I_x|}]$. Let $\operatorname{lcp}(I_a, I_b)$ be the length of their longest common prefix.

Exact Match. We report precision, recall, and micro-F1. A prediction is correct if it exactly matches any gold concept in $Y(p)$.

Unseen Recall-oriented Closeness (U-RC). We compute closeness for each gold concept unseen during training and then average across all of them as the U-RC. Let $\mathcal{P}_{\text{test}}$ be the test passages. For a passage $p \in \mathcal{P}_{\text{test}}$, let $G_{\text{us}}(p) = \{\, g \in Y(p) \mid g \text{ is unseen} \,\}$ and $\hat{Y}(p)$ be the model's predictions for p. The U-RC is calculated as:

$$\text{U-RC} = \frac{\sum_p \sum_{g \in G_{\text{us}}(p)} \max_{\hat{y} \in \hat{Y}(p)} \frac{\operatorname{lcp}(I_g, I_{\hat{y}})}{|I_g|}}{\sum_p |G_{\text{us}}(p)|}.$$

If $\hat{Y}(p) = \varnothing$, the max is defined as 0. Higher is better; it reflects how closely in the hierarchy, predictions approach the correct *unseen* concepts.

Unseen Candidate-set Size (U-CS). To evaluate the model's utility in reducing the search space for unseen concepts, we propose the U-CS.

For each unseen gold $g \in G_{\text{us}}(p)$, let $\hat{y}^{\star}(p, g) = \arg\max_{\hat{y} \in \hat{Y}(p)} \operatorname{lcp}(I_g, I_{\hat{y}})$ (if $\hat{Y}(p) = \varnothing$, define

[5] Model card: meta-llama/Meta-Llama-3-8B-Instruct

[6] Hyperparameters are listed in Appendix A.3.

HPA-MLD (228 instances)			
Pipeline	**Pre**	**Rec**	**F1**
PCC	53.0	**53.0**	53.0
→ Concept Classification (CC)	68.6	46.9	55.7†
→ Relabeling (RL)	66.3	49.3	**56.6**†
→ QS	54.1	50.8	52.4$\star$
→ CC → RL	70.7	44.9	54.9†
→ CC → RL → QS	**71.3**	45.4	55.4†
HoIP-MLD (500 instances)			
PCC	6.3	62.4	11.4
→ Concept Classification (CC)	12.0	51.7	19.5†
→ Relabeling (RL)	8.0	50.6	13.8†
→ GF	11.1	44.0	17.7†
→ QS	7.2	**65.0**	12.9$\star$
→ CC → RL	14.3	43.3	21.5†
→ CC → RL → GF	**20.3**	40.1	27.0†
→ CC → RL → GF → QS	20.2	43.6	**27.6**†

Table 2: Auto-labeling performance on manually labeled datasets (MLDs). "PCC," "GF," and "QS" denote the stages of Passage-to-Claim-to-Concept generation, Guideline-based Filtering, and Quality-based Selection, respectively. A † indicates a significant difference between the annotations produced by "PCC" and those of a given pipeline, as determined by a one-sided paired t-test ($p < 0.05$). A $\star$ indicates one-sided paired t-test is not applicable, because the remaining instances after "PCC-QS" share same annotations with "PCC."

$\mathrm{lcp} = 0$ and $\hat{y}^\star$ undefined). Let $\pi(p, g)$ be the shared prefix of I_g and $I_{\hat{y}^\star}$ (use the empty prefix ϵ when $\mathrm{lcp} = 0$). Let $\mathcal{H}$ denote the *hierarchical label tree* used to define indices, and $\mathcal{C}_\mathcal{H}(\pi) = \{\, x \mid \pi \preceq I_x \,\}$ be the set of concepts whose index starts with prefix π in $\mathcal{H}$. Define the *number of ontology concepts* as $T \triangleq |\mathcal{C}_\mathcal{H}(\epsilon)|$. For each g, set $S(p, g) = |\mathcal{C}_\mathcal{H}(\pi(p, g))|$, with the convention $S(p, g) = T$ when $\mathrm{lcp} = 0$. We aggregate sizes via a harmonic mean:

$$\text{U-CS} = \left(\frac{1}{\sum_p |G_{\text{us}}(p)|} \sum_p \sum_{g \in G_{\text{us}}(p)} \frac{1}{S(p, g)} \right)^{-1}.$$

Lower is better; using the harmonic mean reduces sensitivity to rare extremely large candidate sets.

Together, micro-F1, U-RC, and U-CS capture whether models *hit* the target, *approach* it in the hierarchy, and *shrink* the search space of it.

5 Results

5.1 RQ1: Is LLM-generated ALD sufficient for training effective recognizers?

To answer the RQ1, we first evaluate the performance of our annotation pipeline and then assess the performance of models trained on ALD.

Annotation Pipeline Effectiveness. As shown in Table 2, each module in our pipeline contributes to the final annotation quality. Concept classification, relabeling, and guideline-based filtering consistently improve precision by refining candidates, while quality-based selection enhances recall. When all modules are combined, the pipeline achieves strong gains in both Precision (53.0→71.3 for HPA; 6.3→20.2 for HoIP) and F1 scores (11.4→27.6 for HoIP), validating its design. However, labels generated by our LLM-based pipeline are significantly misaligned with manual annotations (F1 scores are quite low). This gap is particularly pronounced for the novel and fine-grained HoIP ontology, whose concepts are poorly represented in current LLMs, leading to low data annotation quality. Overall, errors introduced by LLM generation and similarity-based matching can propagate across stages, and later filtering and relabeling only partially mitigate them. As a result, residual noise and passage-level quality variation remain, motivating systematic evaluation of auto-labeled data quality and downstream usefulness.

Recognizers Trained Solely on ALD. We proceed to train the MA-COIR using only the generated ALD. The results, presented in Table 3, indicate that models trained on HoIP-ALD underperform their MLD-trained counterparts in terms of overall F1 score. For instance, even the best ALD-trained model achieves an F1 of only 18.5, which is substantially lower than the worst supervised baseline trained on just 200 manually labeled abstracts (Using OSI, 200 instances in HoIP-MLD for training, F1 is 38.5). The significant differences between ALD- and MLD-trained models highlights that **LLM-generated ALD is not a replacement for high-quality manual annotations at present**.

5.2 RQ2: Does ALD Improve Generalization to Unseen Concepts Despite Noise?

Given that ALD cannot match the performance of MLD for recognizer training, we investigate its complementary value with our evaluation framework. Our results reveal a key insight: while annotation quality constrains exact-match performance, **models can learn structural information even from imperfect annotations.** As shown in Table 3, when the training data volume of HoIP-ALD increases, models show consistent improvement on metrics specifically designed to measure general-

HoIP-MLD									
$\|D\|$		OSI			SSI			OSSI	
	F1	U-RC	U-CS↓	F1	U-RC	U-CS↓	F1	U-RC	U-CS↓
200	38.5	27.4	303.8	40.1	22.0	215.8	44.0	**27.8**	**139.8**
400	50.9	26.7	**84.1**	52.8	24.3	110.6	**63.9**	**27.9**	228.6
800	59.4	28.0	**51.4**	74.5	26.1	70.2	**67.3**	**30.0**	136.5
1,600	70.9	30.2	82.8	71.1	29.0	**71.5**	**73.7**	31.5	271.8
2,458	73.1	29.7	67.4	77.3	28.4	42.1	78.6	30.5	150.8
HoIP-ALD									
200	18.5	**30.1**	211.7	13.7	28.6	**129.1**	16.8	28.5	313.0
400	13.9	**32.6**	243.0	**14.1**	28.6	**95.0**	13.4	32.4	120.2
800	**14.9**	**35.4**	172.9	13.9	31.9	**75.9**	11.9	34.7	103.7
1,600	15.9	**38.0**	109.7	14.8	36.0	**55.0**	**18.4**	35.1	94.2
3,200	**18.3**	37.5	106.5	15.0	36.1	**44.8**	15.5	**39.3**	62.5
6,400	**17.2**	**40.8**	69.6	15.7	38.4	**35.0**	15.9	40.6	52.1
12,800	**16.9**	**41.1**	67.5	16.0	38.7	**34.6**	16.7	40.7	51.5
16,415	**17.7**	40.9	73.3	17.6	38.0	33.2	17.3	41.4	44.4
HPA-ALD									
200	19.1	26.5	100.6	**20.4**	**27.0**	**76.0**	16.7	26.8	103.0
400	**20.3**	28.2	68.1	17.4	29.5	64.0	19.2	**30.2**	**61.7**
800	**20.6**	**30.6**	**55.0**	19.2	30.2	60.2	17.7	30.5	55.1
1,600	**20.4**	33.7	70.3	20.3	33.4	59.4	19.1	**34.5**	**41.4**
3,200	**25.7**	**39.3**	39.7	22.9	35.7	45.6	20.7	36.9	**37.2**
6,400	**26.0**	**41.7**	**32.3**	25.0	38.2	45.3	23.6	39.3	35.0
12,800	**28.6**	39.4	41.0	26.8	38.6	36.4	28.2	**40.1**	**29.3**
25,600	**34.9**	43.4	35.5	33.4	41.2	**24.7**	32.4	**46.0**	25.7
47,152	35.3	46.1	20.1	36.0	41.2	24.7	36.7	45.6	20.5

Table 3: Performance on HoIP concept recognition using models trained on HoIP-MLD, and performance on HoIP and HPA concept recognition using models trained on HoIP-ALD and HPA-ALD. $|D|$ denotes training data size. F1 denotes Micro-F1. U-RC refer to Unseen Recall-oriented Closeness. U-CS indicates Unseen Candidate set Size. The best score across indices is highlighted in bold, while the best score achieved by a type of training data is in red.

ization **regardless of index types**.[7] Additionally, OSSI achieves the best U-RC, while SSI achieves the best U-CS on HoIP-ALD. This demonstrates that the two metrics are not redundant and each capturing a unique and important facet of generalization. The evaluation framework is model-agnostic and can be applied to any recognizer; for models without hierarchical indices, concepts can be mapped to OSI/SSI/OSSI before evaluation.

As ALD volume increases, the improvements in U-RC confirm that predictions get closer to the unseen gold concepts within the hierarchy. U-CS shrinks substantially—from 129.1 to 33.2 on HoIP using SSI, and down to 20.1 on HPA using OSI. This demonstrates that the model, trained with more ALD, learns to effectively ruling out implausible concepts from the entire label set. This trend is even more pronounced on HPA, where higher-quality ALD enables all metrics, including F1, to improve with data volume. These results stand in clear contrast to MLD-trained models, where we observed no clear correlation between training data size and unseen concept recognition, likely due to limited ontology coverage (0.53–1.75%).

We attribute ALD's generalization gains to two factors: (1) ALD substantially expands concept coverage and diversity, providing scale and structural context for learning. (2) Hierarchical indices impose a structure that makes noisy ALD supervision learnable. Consistent with prior findings (Liu et al., 2025), and with the OSI/SSI/OSSI gaps observed on HoIP-MLD (combining ontological and semantic signals brings the best F1), indexing plays a critical role. Overall, **hierarchical indexing and ALDs are complementary**: indices structure noise, while ALDs broaden coverage and reinforce hierarchical signals.

6 Additional Analyses

This section provides complementary analyses to better interpret our results. We first validate the downstream utility of U-RC and U-CS in a recognition-to-reranking setting, and then analyze false positive patterns in the LLM-based auto-

[7] As comparisons between indices are less relevant to our research questions, we provide the analysis in Append A.5.

$\vert D\vert$	Recognizer			Reranker		
	F1	U-RC	U-CS	P	R	F1
200	14.7	25.4	118.9	41.6	29.4	33.7
400	14.3	27.8	127.6	35.5	29.4	32.2
800	14.7	28.9	77.1	41.8	29.0	34.2
1,600	13.8	33.9	88.3	**42.7**	30.5	35.6
3,200	16.2	32.6	95.1	40.2	31.1	35.1
6,400	15.0	34.9	59.3	40.1	33.5	36.5
12,800	15.7	**38.0**	**41.4**	37.3	34.0	35.6
16,415	**17.2**	36.7	45.3	40.1	**39.6**	**39.8**

Table 4: Recognition-to-reranking results under a fixed retrieval budget.

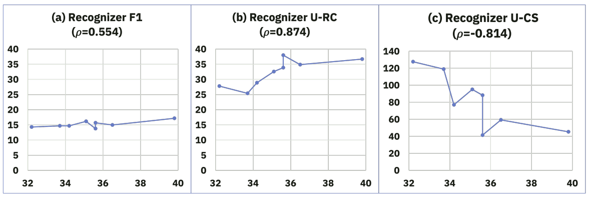


Figure 3: Upstream metrics vs. downstream reranker F1. Each panel plots one recognizer metric against the reranker F1 across eight recognizers trained with different data volumes. ρ denotes Spearman's rank correlation between the upstream metric and reranker F1.

labeling pipeline to clarify its limitations.

6.1 Downstream Utility

To validate the downstream utility of U-RC and U-CS, we conduct a controlled recognition-to-reranking experiment on MLD-HoIP using predictions from eight ALD-trained recognizers (using SSI) over all test passages. For each predicted concept, we retrieve 41 candidate concepts (as the smallest U-CS among recognizers) from the ontology using BM25 and merge the retrieved candidates within each passage to form a passage-level candidate pool. This pool is then reranked by an MLD-trained cross-encoder ($|D| = 2,458$).[8]

Table 4 reports downstream performance under this fixed retrieval budget. Reranker F1 scores vary from 32.2 to 39.8, even when the recognizer micro-F1 remains relatively flat across several training sizes (e.g., 13.8–16.2 from 200 to 3,200), indicating that exact-match accuracy alone does not fully reflect downstream usefulness. Figure 3 further shows that downstream reranker F1 aligns most strongly with U-RC (ρ=0.874) and negatively with U-CS (ρ=−0.814), whereas recognizer micro-F1 exhibits weaker rank correlation (ρ=0.554). Together, these results suggest that U-RC and U-CS capture downstream-relevant quality signals beyond exact-match F1 performance.

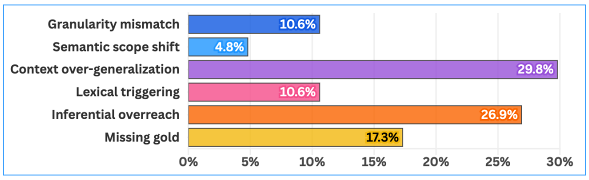


Figure 4: Distribution of false positive outcomes in the LLM-based auto-labeling pipeline. "Missing gold" denotes predictions supported by the passage but absent from dataset-provided gold annotations; the remaining bars correspond to five error types.

6.2 False Positive Error Analysis

We manually analyzed false positive predictions (FPs) from the LLM-based auto-labeling pipeline using a small random sample from HoIP-MLD. Specifically, we randomly selected 20 instances from the 500-instance subset for auto-labeling evaluation, covering 37 gold concepts and 104 initially labeled false positives. We filtered out FPs that are passage-supported but missing from the dataset-provided annotations, as "Missing gold"; 17.3% of apparent FPs fall into this category (see Figure 4). The remaining FPs are grouped into five error types: *granularity mismatch, semantic scope shift, context over-generalization, lexical triggering, and inferential overreach* (definitions in Appendix A.7).

Context over-generalization (29.8%) and *inferential overreach* (26.9%) dominate among true errors, indicating that LLMs often expand beyond passage evidence by turning partial cues or outcomes into broader process claims. The remaining errors include *lexical triggering* (10.6%), *granularity mismatch* (10.6%), and *semantic scope shift* (4.8%). Overall, this analysis clarifies structural limitations of LLM-based auto-labeling—particularly its tendency to over-interpret passage evidence.

7 Conclusion

This work advances Mention-agnostic Biomedical Concept Recognition (MA-BCR) in two key directions: (1) We propose an evaluation framework with hierarchical indices and unseen-aware metrics to quantify generalization. (2) To the best of our knowledge, we introduce the first LLM-based auto-labeling pipeline for MA-BCR and demonstrate that corresponding ALD, while low-quality, enhances generalization to unseen concepts by supplementing coverage and structural information.

[8] Details are provided in Appendix A.6.

Limitations

Despite promising results, several limitations warrant further attention:

Auto-labeling limitations on complex ontologies. While our LLM-based auto-labeling pipeline substantially outperforms direct concept generation, its performance on HoIP remains limited. This reflects the difficulty LLMs face when annotating novel, fine-grained biomedical concepts they probably have never seen during pretraining. Further improvements are needed to enhance precision and recall in underrepresented subdomains. Additionally, using a single LLM across multiple stages (stage 2-5) of auto-labeling may not fully realize the potential of the LLM; different LLMs may perform better at different stages and mitigate error propagation between stages.

Incomplete concept coverage. Our auto-labeled datasets currently cover only 70% of HPA and 54% of HoIP concepts. Many ontology terms fail to retrieve any relevant PubMed abstracts, limiting data diversity. Expanding retrieval strategies and a new data synthesis strategy may help improve both coverage and representativeness.

Indexing strategy remains improvable. OSSI already improves generalization by integrating structural and semantic signals, but more sophisticated methods—e.g., adaptive edge weighting, concept importance modeling, or learned fusion mechanisms—may further enhance performance, especially under noisy supervision.

Scalability of training. As labeled concept coverage grows, training on larger datasets using encoder-decoder models like BART becomes time-intensive. While ontologies are relatively stable, improving training efficiency (e.g., via distillation, continual learning, or lightweight architectures) will be key to scaling this approach in real-world applications.

Automatic evaluation reliability. Finally, although our use of GPT-4o-mini enables scalable evaluation (as the Quality-based selection stage), we found LLM-as-judge still does not perform ideal alignment with manually crafted standards. Human validation remains necessary to calibrate automatic metrics and ensure robust assessment.

Acknowledgments

We thank all reviewers for their careful reading and constructive feedback. We are particularly grateful to the meta-reviewer for their dedicated effort in coordinating the review discussion and providing an exceptionally detailed and clear meta-review. Their thorough reading and thoughtful guidance significantly strengthened this work.

References

J Harry Caufield, Harshad Hegde, Vincent Emonet, Nomi L Harris, Marcin P Joachimiak, Nicolas Matentzoglu, HyeongSik Kim, Sierra Moxon, Justin T Reese, Melissa A Haendel, Peter N Robinson, and Christopher J Mungall. 2024. Structured Prompt Interrogation and Recursive Extraction of Semantics (SPIRES): a method for populating knowledge bases using zero-shot learning. *Bioinformatics*, 40(3):btae104.

Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. 2011. Generalized louvain method for community detection in large networks. In *2011 11th international conference on intelligent systems design and applications*, pages 88–93. IEEE.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

Oumaima El Khettari, Noriki Nishida, Shanshan Liu, Rumana Ferdous Munne, Yuki Yamagata, Solen Quiniou, Samuel Chaffron, and Yuji Matsumoto. 2024. Mention-agnostic information extraction for ontological annotation of biomedical articles. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 457–473, Bangkok, Thailand. Association for Computational Linguistics.

Michael A et al. Gargano. 2023. The human phenotype ontology in 2024: phenotypes around the world. *Nucleic Acids Research*, 52(D1):D1333–D1346.

Tudor Groza, Harry Caufield, Dylan Gration, Gareth Baynam, Melissa A Haendel, Peter N Robinson, Christopher J Mungall, and Justin T Reese. 2024. An evaluation of gpt models for phenotype concept recognition. *BMC Medical Informatics and Decision Making*, 24(1):30.

George Karypis and Vipin Kumar. 1997. Metis: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices.

Siddhant Kharbanda, Atmadeep Banerjee, Erik Schultheis, and Rohit Babbar. 2022. Cascadexml: Rethinking transformers for end-to-end multi-resolution training in extreme multi-label classification. *Advances in neural information processing systems*, 35:2074–2087.

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. 2016. Biocreative V CDR task corpus:

a resource for chemical disease relation extraction. *Database J. Biol. Databases Curation*, 2016.

Shanshan Liu, Noriki Nishida, Rumana Ferdous Munne, Narumi Tokunaga, Yuki Yamagata, Kouji Kozaki, and Yuji Matsumoto. 2025. Ma-coir: Leveraging semantic search index and generative models for ontology-driven biomedical concept recognition. *Preprint*, arXiv:2505.12964.

Manuel Lobo, Andre Lamurias, and Francisco M. Couto. 2017. Identifying human phenotype terms by combining machine learning and validation rules. *BioMed Research International*, 2017(1):8565739.

Ling Luo, Shankai Yan, Po-Ting Lai, Daniel Veltri, Andrew Oler, Sandhya Xirasagar, Rajarshi Ghosh, Morgan Similuk, Peter N Robinson, and Zhiyong Lu. 2021. Phenotagger: a hybrid method for phenotype concept recognition using human phenotype ontology. *Bioinformatics*, 37(13):1884–1890.

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

Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, Tal Schuster, William W. Cohen, and Donald Metzler. 2022. Transformer memory as a differentiable search index. *Preprint*, arXiv:2202.06991.

Qinyong Wang, Zhenxiang Gao, and Rong Xu. 2023. Exploring the in-context learning ability of large language model for biomedical concept linking. *Preprint*, arXiv:2307.01137.

Yuki Yamagata, Tatsuya Kushida, Shuichi Onami, and Hiroshi Masuya. 2024. Homeostasis imbalance process ontology: a study on covid-19 infectious processes.

Jiong Zhang, Wei-Cheng Chang, Hsiang-Fu Yu, and Inderjit Dhillon. 2021. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. *Advances in Neural Information Processing Systems*, 34:7267–7280.

A Appendix

A.1 Auto-Labeled Data Construction

A.1.1 Collect passages for annotation

We retrieved PubMed article abstracts using ontology term names as search keywords. For each term, a single representative name was used to query PubMed via the NIH E-utilities API[9], requesting up to the 10 most recent articles. The abstracts of the retrieved articles were collected as "passages". Although we aimed to obtain 10 abstracts per concept, some terms yielded fewer or no results. Multiple terms could be associated with the same articles, we removed duplicate abstracts to ensure all passages in the dataset are unique.

A.1.2 Prompts

Prompts of Passage-to-Claim-to-Concept generation, Concept Classification, Relabeling, and Guideline-based Filtering are listed in Figure 5, 6, 7, 8, respectively.

In Quality-based Selection stage, we use the prompt similar as the relabeling step (Figure 7), but remove the output request for “explicit/implicit/missing/additional concept”, and only take the “Quality” returned by the LLMs for processing.

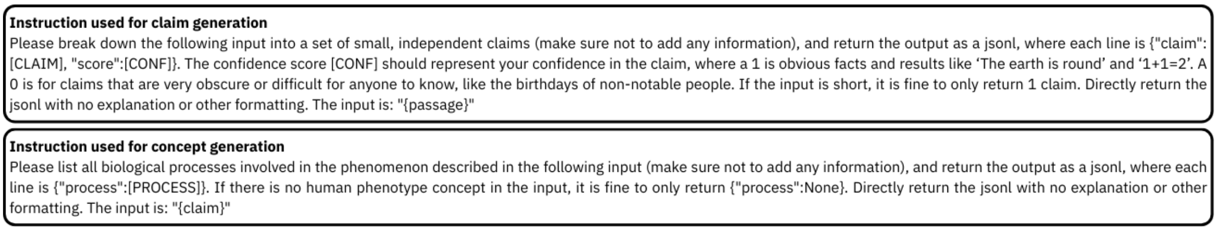
A.1.3 Inference costs

While the pipeline may appear complex, it was designed to replace expensive manual annotation for biomedical ontologies like HPO and HoIP, which lack high-coverage existing training corpora. For instance, annotating 300 HoIP abstracts manually took nearly 6 months by a domain expert. In contrast, our pipeline processes an abstract in under 60 seconds, with a total cost under $150 to generate 90,000+ annotated abstracts—orders of magnitude cheaper and faster. We deliberately selected affordable LLMs (e.g., Llama-3-8B, GPT-4o-mini) over more costly alternatives. We have found Llama-3-8B achieves better scores than GPT-3.5-turbo, GPT-4o-mini and GPT-4o in Passage-to-Claim-to-Concept generation stage. We have found GPT-4o improves F1 by only 0.5 but increases cost 16× in the Concept Classification stage. We believe our design strikes a practical balance between scalability and performance.

We implement Llama-3-8B on one NVIDIA A100 Tensor Core GPU, so there is no API cost for Passage-to-Claim-to-Concept generation, and the processing time is average 22.8 seconds per abstract.

The cost of each stage, given the candidate list obtained by Passage-to-Claim-to-Concept generation using GPT-4o-mini for annotating HPA concepts for one abstract in HPO GSC+ corpus, is listed in the Table 5 for reference. The whole pipeline costs less than simply taking the cost in the table as a total, because each stage will remove

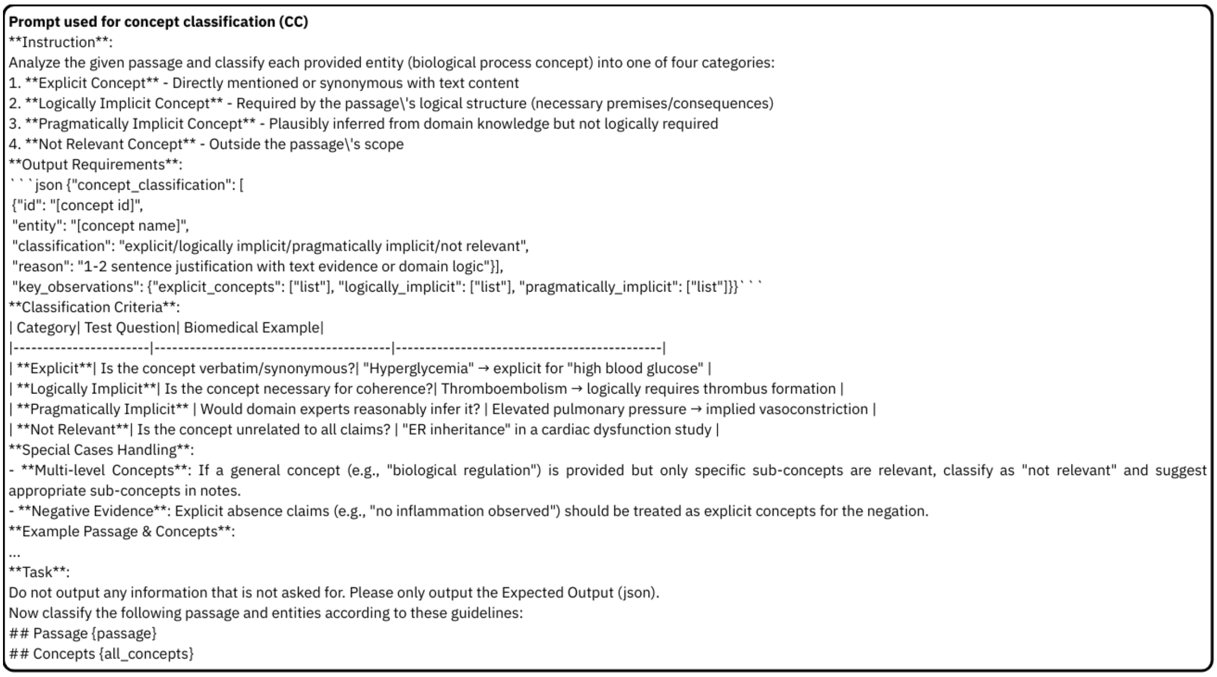
[9] https://www.ncbi.nlm.nih.gov/books/NBK25500/



~~~~
Instruction used for claim generation
Please break down the following input into a set of small, independent claims (make sure not to add any information), and return the output as a jsonl, where each line is {"claim":[CLAIM], "score":[CONF]}. The confidence score [CONF] should represent your confidence in the claim, where a 1 is obvious facts and results like 'The earth is round' and '1+1=2'. A 0 is for claims that are very obscure or difficult for anyone to know, like the birthdays of non-notable people. If the input is short, it is fine to only return 1 claim. Directly return the jsonl with no explanation or other formatting. The input is: "{passage}"
~~~~

~~~~
Instruction used for concept generation
Please list all biological processes involved in the phenomenon described in the following input (make sure not to add any information), and return the output as a jsonl, where each line is {"process":[PROCESS]}. If there is no human phenotype concept in the input, it is fine to only return {"process":None}. Directly return the jsonl with no explanation or other formatting. The input is: "{claim}"
~~~~

Figure 5: The prompts we used for PCC stage for HoIP concept generation.



~~~~
Prompt used for concept classification (CC)
**Instruction**:
Analyze the given passage and classify each provided entity (biological process concept) into one of four categories:
1. **Explicit Concept** - Directly mentioned or synonymous with text content
2. **Logically Implicit Concept** - Required by the passage\'s logical structure (necessary premises/consequences)
3. **Pragmatically Implicit Concept** - Plausibly inferred from domain knowledge but not logically required
4. **Not Relevant Concept** - Outside the passage\'s scope
**Output Requirements**:
```json {"concept_classification": [
 {"id": "[concept id]",
 "entity": "[concept name]",
 "classification": "explicit/logically implicit/pragmatically implicit/not relevant",
 "reason": "1-2 sentence justification with text evidence or domain logic"}],
 "key_observations": {"explicit_concepts": ["list"], "logically_implicit": ["list"], "pragmatically_implicit": ["list"]}}```
**Classification Criteria**:
| Category| Test Question| Biomedical Example|
|----------------------|---------------------------------------|-------------------------------------------|
| **Explicit**| Is the concept verbatim/synonymous?| "Hyperglycemia" → explicit for "high blood glucose" |
| **Logically Implicit**| Is the concept necessary for coherence?| Thromboembolism → logically requires thrombus formation |
| **Pragmatically Implicit** | Would domain experts reasonably infer it? | Elevated pulmonary pressure → implied vasoconstriction |
| **Not Relevant**| Is the concept unrelated to all claims? | "ER inheritance" in a cardiac dysfunction study |
**Special Cases Handling**:
- **Multi-level Concepts**: If a general concept (e.g., "biological regulation") is provided but only specific sub-concepts are relevant, classify as "not relevant" and suggest appropriate sub-concepts in notes.
- **Negative Evidence**: Explicit absence claims (e.g., "no inflammation observed") should be treated as explicit concepts for the negation.
**Example Passage & Concepts**:
...
**Task**:
Do not output any information that is not asked for. Please only output the Expected Output (json).
Now classify the following passage and entities according to these guidelines:
## Passage {passage}
## Concepts {all_concepts}
~~~~

Figure 6: The prompt used for concept classification.

Stage	Time (s)	API cost ($)
Concept Classification	6.81	0.0003
Relabeling	5.62	0.0003
Quality-based Selection	4.04	0.0003

Table 5: Costs for annotating one abstract in the CRAFT corpus using GPT-4o-mini. The candidate lists obtained by Passage-to-Claim-to-Concept generation are used as inputs. The number is an average among 100 abstracts.

part of the concepts, so that the input and output tokens will be reduced.

As Guideline-based Filtering stage only is applied for HoIP concept annotation, we report the cost of annotating 100 passages from CRAFT corpus using GPT-4o-mini, given the candidate lists obtained by Passage-to-Claim-to-Concept generation for reference: 701 seconds and 0.04 dollar.

A.2 Current limitations of MLDs.

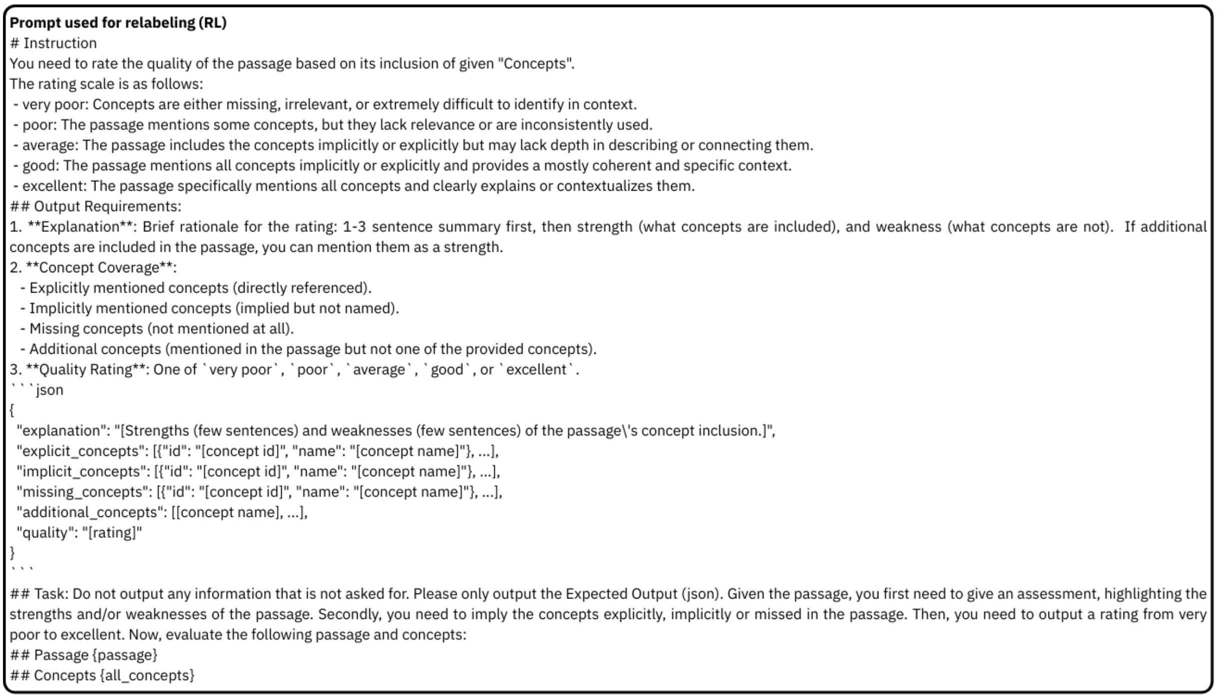
Accurate evaluation of BCR systems relies on high-quality, manually annotated text-concept pairs. In this work, we focus on two critical categories: Human Phenotype Abnormality (HPA), documented in the Human Phenotype Ontology (HPO)[10], and Homeostasis Imbalance Process (HoIP), documented in the Homeostasis Imbalance Process Ontology (HOIP)[11]. While the HPO GSC+ dataset provides reliable annotations for HPA, manually curated resources for HoIP remain scarce and inconsistent, posing significant challenges for systematic evaluation. To partially address this gap, we repurposed the CRAFT corpus—a dataset annotated with Gene Ontology (GO)[12] terms—leveraging the conceptual overlap between HoIP and GO. For HPA, we used HPO GSC+ annotations restricted to descendants of "HP:0000118-Phenotypic abnormality". For HoIP, we used CRAFT passages annotated with GO terms subsumed by the HoIP ontology.

We show some passage-concept pairs as exam-

[10] https://hpo.jax.org/

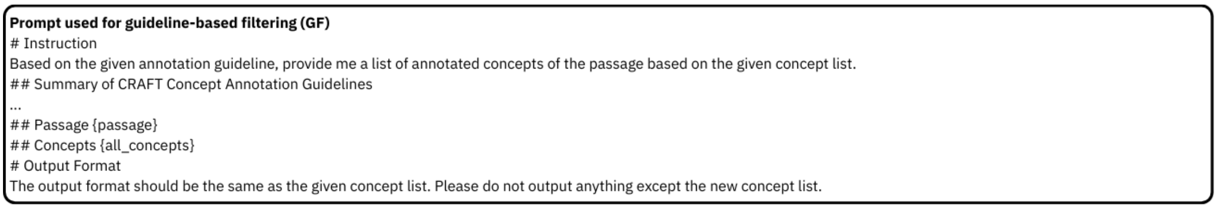
[11] https://bioportal.bioontology.org/ontologies/HOIP

[12] http://purl.obolibrary.org/obo/go.owl



~~~
Prompt used for relabeling (RL)
# Instruction
You need to rate the quality of the passage based on its inclusion of given "Concepts".
The rating scale is as follows:
 - very poor: Concepts are either missing, irrelevant, or extremely difficult to identify in context.
 - poor: The passage mentions some concepts, but they lack relevance or are inconsistently used.
 - average: The passage includes the concepts implicitly or explicitly but may lack depth in describing or connecting them.
 - good: The passage mentions all concepts implicitly or explicitly and provides a mostly coherent and specific context.
 - excellent: The passage specifically mentions all concepts and clearly explains or contextualizes them.
## Output Requirements:
1. **Explanation**: Brief rationale for the rating: 1-3 sentence summary first, then strength (what concepts are included), and weakness (what concepts are not).  If additional
concepts are included in the passage, you can mention them as a strength.
2. **Concept Coverage**:
  - Explicitly mentioned concepts (directly referenced).
  - Implicitly mentioned concepts (implied but not named).
  - Missing concepts (not mentioned at all).
  - Additional concepts (mentioned in the passage but not one of the provided concepts).
3. **Quality Rating**: One of `very poor`, `poor`, `average`, `good`, or `excellent`.
```json
{
  "explanation": "[Strengths (few sentences) and weaknesses (few sentences) of the passage\'s concept inclusion.]",
  "explicit_concepts": [{"id": "[concept id]", "name": "[concept name]"}, ...],
  "implicit_concepts": [{"id": "[concept id]", "name": "[concept name]"}, ...],
  "missing_concepts": [{"id": "[concept id]", "name": "[concept name]"}, ...],
  "additional_concepts": [[concept name], ...],
  "quality": "[rating]"
}
```
## Task: Do not output any information that is not asked for. Please only output the Expected Output (json). Given the passage, you first need to give an assessment, highlighting the
strengths and/or weaknesses of the passage. Secondly, you need to imply the concepts explicitly, implicitly or missed in the passage. Then, you need to output a rating from very
poor to excellent. Now, evaluate the following passage and concepts:
## Passage {passage}
## Concepts {all_concepts}
~~~

Figure 7: The prompt used for relabeling.



```
Prompt used for guideline-based filtering (GF)
# Instruction
Based on the given annotation guideline, provide me a list of annotated concepts of the passage based on the given concept list.
## Summary of CRAFT Concept Annotation Guidelines
...
## Passage {passage}
## Concepts {all_concepts}
# Output Format
The output format should be the same as the given concept list. Please do not output anything except the new concept list.
```

Figure 8: The prompt used for guideline-based filtering.

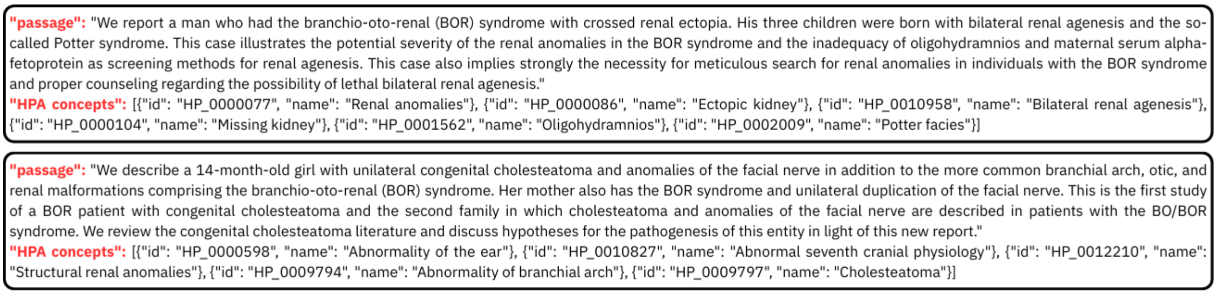
ples in Figure 9 and Figure 10 for better understanding what kind of concepts we are targeted.

Inherent challenges in manual annotation Our analysis uncovered inconsistencies in annotation practices. In the HPO GSC+ corpus, fine-grained terms (e.g., bilateral vestibular schwannoma) are often annotated alongside their hypernyms (vestibular schwannoma, schwannoma), but this hierarchy is not applied systematically. For example, broader parent terms are occasionally omitted even when their descendants are labeled, reflecting inconsistencies in annotation protocols.

The CRAFT corpus presents even more pronounced limitations. Annotated in 2010, it lacks contemporary GO refinements: some terms are now obsolete (e.g., former Homeostasis Imbalance Processes reclassified as molecular functions), while some are annotated at overly broad levels by current standards. Although we excluded obsolete terms, correcting overly general annotations to match modern ontologies would require infeasible manual effort. Consequently, our evaluation necessarily inherits these historical biases.

The scarcity problem. As Table 1-MLD illustrates, the available manual annotations cover only 2.35% of target ontology concepts by both HPO GSC+ and CRAFT. This extreme sparsity—coupled with the inconsistencies described above—underscores a key bottleneck: manual annotations are insufficient to robustly evaluate CR systems, let alone train them.

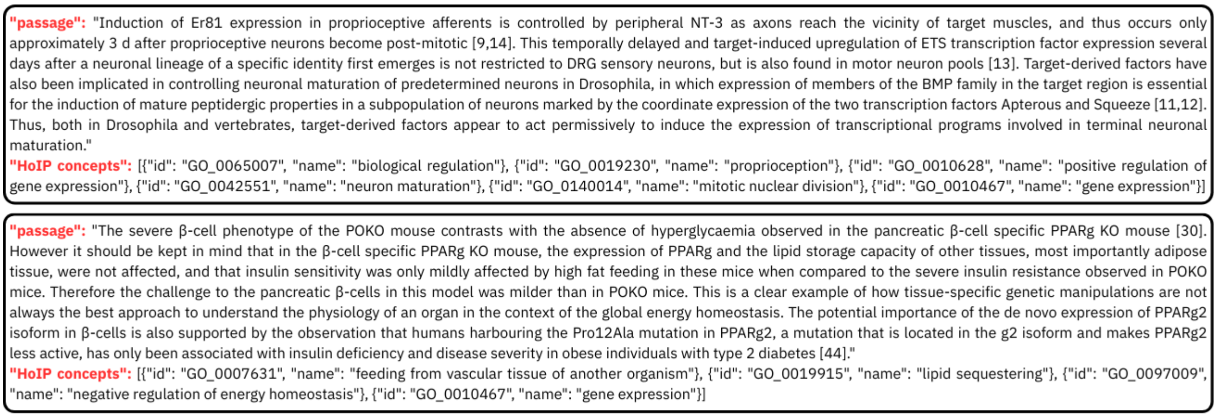
The ratio of implicit concepts. No existing dataset explicitly distinguishes implicit vs. explicit concepts in annotations, so a ratio cannot be reliably calculated. According to our own observation, there is around 20% of annotated concepts implicit with respect to their corresponding annotated mentions.



"passage": "We report a man who had the branchio-oto-renal (BOR) syndrome with crossed renal ectopia. His three children were born with bilateral renal agenesis and the so-called Potter syndrome. This case illustrates the potential severity of the renal anomalies in the BOR syndrome and the inadequacy of oligohydramnios and maternal serum alpha-fetoprotein as screening methods for renal agenesis. This case also implies strongly the necessity for meticulous search for renal anomalies in individuals with the BOR syndrome and proper counseling regarding the possibility of lethal bilateral renal agenesis."
"HPA concepts": [{"id": "HP_0000077", "name": "Renal anomalies"}, {"id": "HP_0000086", "name": "Ectopic kidney"}, {"id": "HP_0010958", "name": "Bilateral renal agenesis"}, {"id": "HP_0000104", "name": "Missing kidney"}, {"id": "HP_0001562", "name": "Oligohydramnios"}, {"id": "HP_0002009", "name": "Potter facies"}]

"passage": "We describe a 14-month-old girl with unilateral congenital cholesteatoma and anomalies of the facial nerve in addition to the more common branchial arch, otic, and renal malformations comprising the branchio-oto-renal (BOR) syndrome. Her mother also has the BOR syndrome and unilateral duplication of the facial nerve. This is the first study of a BOR patient with congenital cholesteatoma and the second family in which cholesteatoma and anomalies of the facial nerve are described in patients with the BO/BOR syndrome. We review the congenital cholesteatoma literature and discuss hypotheses for the pathogenesis of this entity in light of this new report."
"HPA concepts": [{"id": "HP_0000598", "name": "Abnormality of the ear"}, {"id": "HP_0010827", "name": "Abnormal seventh cranial physiology"}, {"id": "HP_0012210", "name": "Structural renal anomalies"}, {"id": "HP_0009794", "name": "Abnormality of branchial arch"}, {"id": "HP_0009797", "name": "Cholesteatoma"}]

Figure 9: Two instances in HPA-MLD (from HPO GSC+ corpus).



"passage": "Induction of Er81 expression in proprioceptive afferents is controlled by peripheral NT-3 as axons reach the vicinity of target muscles, and thus occurs only approximately 3 d after proprioceptive neurons become post-mitotic [9,14]. This temporally delayed and target-induced upregulation of ETS transcription factor expression several days after a neuronal lineage of a specific identity first emerges is not restricted to DRG sensory neurons, but is also found in motor neuron pools [13]. Target-derived factors have also been implicated in controlling neuronal maturation of predetermined neurons in Drosophila, in which expression of members of the BMP family in the target region is essential for the induction of mature peptidergic properties in a subpopulation of neurons marked by the coordinate expression of the two transcription factors Apterous and Squeeze [11,12]. Thus, both in Drosophila and vertebrates, target-derived factors appear to act permissively to induce the expression of transcriptional programs involved in terminal neuronal maturation."
"HoIP concepts": [{"id": "GO_0065007", "name": "biological regulation"}, {"id": "GO_0019230", "name": "proprioception"}, {"id": "GO_0010628", "name": "positive regulation of gene expression"}, {"id": "GO_0042551", "name": "neuron maturation"}, {"id": "GO_0140014", "name": "mitotic nuclear division"}, {"id": "GO_0010467", "name": "gene expression"}]

"passage": "The severe β-cell phenotype of the POKO mouse contrasts with the absence of hyperglycaemia observed in the pancreatic β-cell specific PPARg KO mouse [30]. However it should be kept in mind that in the β-cell specific PPARg KO mouse, the expression of PPARg and the lipid storage capacity of other tissues, most importantly adipose tissue, were not affected, and that insulin sensitivity was only mildly affected by high fat feeding in these mice when compared to the severe insulin resistance observed in POKO mice. Therefore the challenge to the pancreatic β-cells in this model was milder than in POKO mice. This is a clear example of how tissue-specific genetic manipulations are not always the best approach to understand the physiology of an organ in the context of the global energy homeostasis. The potential importance of the de novo expression of PPARg2 isoform in β-cells is also supported by the observation that humans harbouring the Pro12Ala mutation in PPARg2, a mutation that is located in the g2 isoform and makes PPARg2 less active, has only been associated with insulin deficiency and disease severity in obese individuals with type 2 diabetes [44]."
"HoIP concepts": [{"id": "GO_0007631", "name": "feeding from vascular tissue of another organism"}, {"id": "GO_0019915", "name": "lipid sequestering"}, {"id": "GO_0097009", "name": "negative regulation of energy homeostasis"}, {"id": "GO_0010467", "name": "gene expression"}]

Figure 10: Two instances in HoIP-MLD (from CRAFT corpus).

Item	Value
model_card	facebook/bart-large
learning_rate	1e-5
num_epoch	50
batch_size	4
max_length_of_tokens	1024

Table 6: Hyperparameters of the recognizer.

A.3 Details of Concept Recognizer

Hyperparameters. The BART-based language model (facebook/bart-large) used in MA-COIR for recognition is trained with hyperparameters listed in the Table 6. We trained all models on one NVIDIA A100 Tensor Core GPU. For one training instance, the experiment ran for an average of 1.82s/it.

Data splits. The statistics of each data split used for concept recognizer training are shown in the Table 7.

A.4 Details of Hierarchical Graph Partitioning

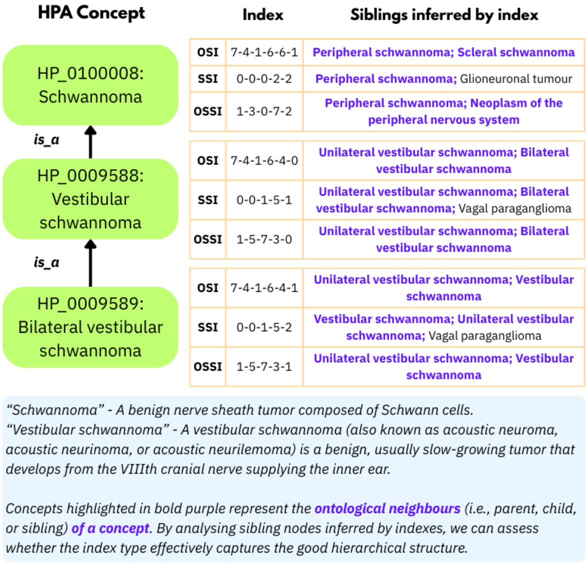
We apply Louvain clustering by default (De Meo et al., 2011), and switch to METIS (Karypis and Kumar, 1997) when subgraphs are too sparse or chain-like to meet the constraint (≤ 10 children of each internal node).

An example of search indices is shown in Figure 11.

A.5 Comparison of Hierarchical indices

As shown in Table 3, On HoIP, OSSI achieves the highest U-RC, while HPA consistently favors OSI. SSI most effectively reduces the candidate set size of unseen concepts on HoIP (lowest U-CS), whereas OSI and OSSI are comparable on HPA. These differences derive from the structural properties of the ontologies: HPO is a shallow, relatively balanced tree, allowing OSI to be both discriminative and learnable. In contrast, HoIP Ontology is deep and entangled, with multi-parent nodes and root-to-leaf paths that may exceed 10 layers. In this setting, OSI produces longer index sequences (avg. depth: 6.15 vs. 5.42 for SSI) and suffers from branching imbalance. While the lack of ontological information degrades F1, SSI compensates by narrowing candidate spaces more aggressively. OSSI balances both structural and semantic signals, yielding superior overall performance.

Additionally, OSSI achieves the best U-RC, while SSI achieves the best U-CS on HoIP-ALD. This demonstrates that the two metrics are not redundant and each capturing a unique and impor-



HPA Concept		Index	Siblings inferred by index
HP_0100008: Schwannoma	OSI	7-4-1-6-6-1	**Peripheral schwannoma; Scleral schwannoma**
	SSI	0-0-0-2-2	**Peripheral schwannoma;** Glioneuronal tumour
	OSSI	1-3-0-7-2	**Peripheral schwannoma; Neoplasm of the peripheral nervous system**
HP_0009588: Vestibular schwannoma	OSI	7-4-1-6-4-0	**Unilateral vestibular schwannoma; Bilateral vestibular schwannoma**
	SSI	0-0-1-5-1	**Unilateral vestibular schwannoma; Bilateral vestibular schwannoma;** Vagal paraganglioma
	OSSI	1-5-7-3-0	**Unilateral vestibular schwannoma; Bilateral vestibular schwannoma**
HP_0009589: Bilateral vestibular schwannoma	OSI	7-4-1-6-4-1	**Unilateral vestibular schwannoma; Vestibular schwannoma**
	SSI	0-0-1-5-2	**Vestibular schwannoma; Unilateral vestibular schwannoma;** Vagal paraganglioma
	OSSI	1-5-7-3-1	**Unilateral vestibular schwannoma; Vestibular schwannoma**

"Schwannoma" - A benign nerve sheath tumor composed of Schwann cells.
"Vestibular schwannoma" - A vestibular schwannoma (also known as acoustic neuroma, acoustic neurinoma, or acoustic neurilemoma) is a benign, usually slow-growing tumor that develops from the VIIIth cranial nerve supplying the inner ear.

*Concepts highlighted in bold purple represent the **ontological neighbours** (i.e., parent, child, or sibling) **of a concept**. By analysing sibling nodes inferred by indexes, we can assess whether the index type effectively captures the good hierarchical structure.*

Figure 11: An example of constructed search indices. When the information used for graph construction is modified, a concept may share the same index prefix with different concepts.

tant facet of generalization. U-RC measures hierarchical proximity but not the reduction in search space. A prediction and the gold concept may reside in the same penultimate cluster that contains 2 or more candidates—a critical difference U-RC misses. Therefore, we propose the U-CS. We highlight the importance of **selecting evaluation metrics aligned with downstream application goals**. When predictions are used directly, closeness metrics like F1 and U-RC are paramount. However, when downstream modules perform precision refinement, candidate space reduction (e.g., U-CS) becomes the more relevant indicator.

A.6 Details of Concept Reranker

Model. Our downstream reranker is a cross-encoder implemented with the HuggingFace Transformers library using `AutoModelForSequenceClassification` (`num_labels=1`), initialized from a SapBERT checkpoint. Given a passage and a candidate concept text, the model outputs a scalar relevance score for reranking.

Training objective and negatives. We train the reranker with a listwise softmax loss over groups consisting of one positive concept and k hard negatives (k=4 by default). For each training passage–gold concept pair, we use the gold concept text as a BM25 query to retrieve a candidate list from the ontology and treat the retrieved non-gold concepts as hard negatives (20 per passage–gold concept pair). Hard negatives are randomly resampled per batch to improve robustness.

Model selection on gold-based candidates. For dev-time model selection, we evaluate using the same supervision format as training: for each passage–gold concept pair, we construct a BM25 candidate set using the gold concept text and measure reranking quality (e.g., nDCG@K/MRR@K). Early stopping is applied based on this gold-based dev evaluation, ensuring that model selection does not depend on recognizer-generated candidates.

Final evaluation on recognizer-based candidates. For the downstream utility experiment, candidate sets are generated from recognizer predictions: each predicted concept retrieves a fixed number of ontology candidates via BM25 (41 per prediction), which are merged (deduplicated union) into a passage-level pool and reranked. We select a score threshold on the dev split by maximizing micro-F1, and then apply the dev-selected threshold to compute micro Precision/Recall/F1 on the test split.

A.7 Error taxonomy

To enable systematic characterization of false positive predictions in biological process annotation, we define an error taxonomy consisting of five categories.

Predictions are evaluated against a passage-supported reference set constructed as follows: we begin with the dataset-provided gold annotations, and additionally include biological process concepts that are explicitly stated or unambiguously implied in the passage but missing from the original gold. These missing concepts are identified through manual review of predictions initially labeled as false positives. Accordingly, a prediction is considered erroneous only if it cannot be justified by passage evidence under this reference set. This refinement step corrects for incomplete coverage in the original dataset annotations and does not depend on model predictions beyond using them to surface candidate missing concepts for review.

Importantly, error categories describe **how** an unsupported prediction arises, rather than simply whether it is incorrect.

A.7.1 Definition of error types

E-1: Granularity Mismatch A granularity mismatch occurs when the passage supports a specific

HoIP-MLD					
Training			Test		
$\|D\|$	$\|Concept\|$	Coverage (%)	$\|Concept\|$	$\|Unseen\_Concept\|$	Seen (%)
200	155	0.53	229	227	38.86
400	236	0.80	229	160	53.71
800	313	1.07	229	144	58.52
1,600	423	1.44	229	119	66.81
2,458	515	1.75	229	111	69.00
HoIP-ALD					
200	744	2.53	635	477	24.88
400	1,400	4.77	635	424	33.23
800	2,399	8.17	635	379	40.31
1,600	3,881	13.22	635	328	48.35
3,200	5,801	19.75	635	286	54.96
6,400	8,181	27.86	635	246	61.26
12,800	10,844	36.93	635	226	64.41
16,415	11,848	40.34	635	219	65.51
HPA-ALD					
200	416	2.27	386	304	21.24
400	848	4.62	386	288	25.39
800	1,523	8.30	386	264	31.61
1,600	2,515	13.70	386	234	39.38
3,200	3,942	21.48	386	207	46.37
6,400	5,724	31.19	386	186	51.81
12,800	7,918	43.14	386	163	57.77
25,600	10,188	55.51	386	137	64.51
47,152	11,989	65.32	386	125	67.62

Table 7: Dataset statistics of splits for recognizer training and evaluation. $|D|$ denotes training data size. $|Concept|$ denotes the number of unique concepts. “Coverage(%)” refers to the percentage of ontology concepts presented in the dataset. “Seen(%)” refers to the percentage of test concepts presented in the training set.

concept, but the prediction selects a term that is hierarchically related in the ontology (an ancestor or descendant node) while failing to match the precise level of abstraction evidenced in the text. These errors are strictly confined to the same ontology lineage; the model identifies the correct conceptual path but fails to calibrate the node's depth relative to the textual evidence.

E-2: Semantic Scope Shift This category applies when a prediction deviates from the semantic domain of the target concept to an unrelated class. Unlike the hierarchical depth errors in E-1, a scope shift represents a thematic misalignment where the prediction belongs to an entirely different conceptual branch. For instance, misidentifying a *learning* event as a *memory* event changes the underlying biological phenomenon being documented, thereby misrepresenting the specific functional activity characterized in the passage.

E-3: Context Over-generalization This error arises when the presence of an isolated observation or a specific biological outcome is misinterpreted as sufficient evidence for a broader, encompassing concept that is not itself stated. In these cases, the textual evidence may describe a necessary component or a resultant state of a process, but the existence of the process as a whole is not grounded in the passage. Unlike the hierarchical positioning in E-1, E-3 represents an inductive leap where partial information is erroneously treated as the occurrence of a complex, multi-stage event.

E-4: Lexical Triggering Lexical triggering refers to predictions driven by the presence of salient keywords or surface patterns rather than by the actual conceptual information. These errors reflect a reliance on stereotypical associations, where the mention of a domain-specific term—such as the name of a gene or protein (*e.g., "Shh"*) or a specific body part (*e.g., "testis"*)—biases the model toward predicting an associated concept (*e.g., "gene expression"* or *e.g., "male gonad development"*) that is not explicitly supported by the text. Here, the model's internal statistical bias regarding the keyword overrides the contextual evidence.

E-5: Inferential Overreach Inferential overreach captures predictions that necessitate unstated logical steps or external assumptions beyond the explicit description. While E-3 over-generalizes from an observed outcome to a broader conceptual class, E-5 adds entirely new explanatory logic that is absent from the text. This occurs when the model presupposes missing intermediate connections, moving from a simple descriptive statement to an unsupported causal chain.

A.7.2 Diagnostic Criteria

To ensure consistent classification, we distinguish the five categories via a three-step decision logic:

1. *Lineage Calibration:* If the prediction and a supported concept share the same ontological path (i.e., one is an ancestor of the other), is the error solely due to the node's depth? (If yes → *E-1*)

2. *Domain Alignment:* Does the prediction shift to a fundamentally different semantic class or an unrelated branch within the ontology? (If yes → *E-2*)

3. *Evidence Gap Analysis:* Is the lack of support due to over-extending a partial observation (*E-3*), keyword-driven bias (*E-4*), or the addition of an unsupported causal chain (*E-5*)?