

# Teaching Through Analogies: A Modular Pipeline for Educational Analogy Generation

Mariam Barakat<sup>1</sup> and Ekaterina Kochmar<sup>1,2</sup>

<sup>1</sup>Mohamed bin Zayed University of Artificial Intelligence, <sup>2</sup>University of Victoria  
{mariam.barakat, ekaterina.kochmar}@mbzuai.ac.ae

## Abstract

Analogies help learners understand unfamiliar concepts by relating them to known concepts. Despite recent advances, large language models (LLMs) continue to struggle to generate analogies of comparable quality to those produced by humans. We present a modular pipeline for educational analogy generation, decomposing the task into four stages: source finding, sub-concept generation, explanation generation, and evaluation. Grounded in Structure Mapping Theory, the pipeline enables systematic, stage-by-stage analysis of how model choice and input configuration affect analogy quality. We evaluate 12 state-of-the-art LLMs across six model families on two datasets with structured sub-concept annotations (SCAR and ParallelPARC), alongside seven embedding models for closed-setting retrieval. Our results show that sub-concepts substantially improve explanation quality and closed setting retrieval precision but provide limited benefit in open-ended source generation. We further introduce an LLM-as-a-judge evaluation methodology and validate its scoring against human annotations from seven annotators, finding that Claude Sonnet 4.6 aligns more reliably with human rankings than with fine-grained absolute scores. Taken together, our findings reveal cross-stage interactions that isolated studies cannot capture, and highlight sub-concept grounding as a key driver of analogy quality generation.

## 1 Introduction

As educational systems evolve in the age of AI, memorisation is no longer sufficient. Instead, students are required to be able to reason across domains and understand underlying structures – skills that are foundational to critical thinking in a rapidly changing world. One of the effective ways to develop these abilities is through analogies (Eriksson et al., 2024), which facilitate learning by mapping unfamiliar knowledge (*target system*) to prior

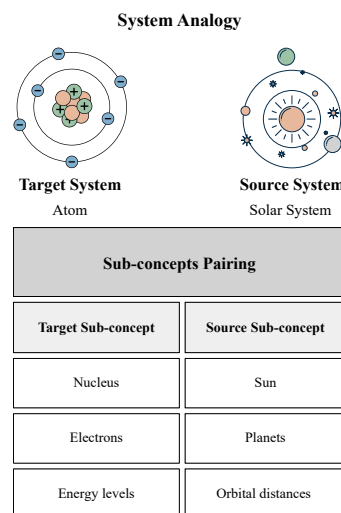


Figure 1: Illustrative example of a system analogy between the *atom* and the *solar system*, showing the source and target systems and their aligned sub-concepts.

knowledge (*source system*) through relational structures (Ye et al., 2024; Czinczoll et al., 2022; Bettin and Ott, 2023). Even when a significant cognitive gap exists, intermediate “bridging” analogies can guide students toward comprehension (Brown and Clement, 1989). In this work, we frame analogy generation as a modular pipeline to enable systematic analysis and improvement of each stage of the process.

Following previous research (Yuan et al., 2023a), we use the term *system* rather than *concept* to emphasise that analogies map multiple components. We refer to these constituent properties as *sub-concepts* throughout this paper, using “sub-” to emphasise that they are components of the larger system rather than standalone concepts.

As shown in Figure 1, explaining the *atom* (target system) through the *solar system* (source system) requires the explained mapping of individual sub-concepts: the *nucleus* maps to the *sun*, *electrons* map to *planets*, and *energy levels* map to

*orbital distances*.

Automatically generating high-quality analogies remains an open challenge, as poorly constructed analogies can introduce or reinforce misconceptions through incorrect mappings or oversimplification of underlying concepts (Zhou et al., 2025; Shao et al., 2025).

Despite recent progress, analogy generation research remains fragmented: prior work addresses retrieval, generation, or evaluation in isolation, without connecting them into a unified pipeline (Yuan et al., 2023a,b; Zhai et al., 2024; Sultan et al., 2024). Source finding relies predominantly on embedding-based retrieval over closed candidate pools, favouring surface similarity over structural alignment (Yuan et al., 2023a; Czinczoll et al., 2022; Jiayang et al., 2023), while sub-concept annotations in SCAR (Yuan et al., 2023a) and ParallelPARC (Sultan et al., 2024) are used for dataset construction rather than to guide generation. Evaluation remains under-addressed, as neither lexical metrics nor LLM-as-a-judge approaches align reliably with human judgments (Zhai et al., 2024; Bhavya et al., 2024), and most studies evaluate GPT-4 variants exclusively (Yuan et al., 2023a; Kim et al., 2023; Sultan et al., 2024; Shao et al., 2025; Li et al., 2024), leaving cross-model generalisability unexplored.

In this paper, we make three contributions: (1) a modular pipeline for educational analogy generation, grounded in Structure Mapping Theory (Gentner, 1981), decomposed into source finding, sub-concept generation, explanation generation, and evaluation; (2) a systematic cross-model evaluation across 12 LLMs from six families, finding that sub-concepts improve retrieval and explanation quality but offer limited benefit in open-ended source generation; and (3) a closed-to-open evaluation methodology, moving from controlled evaluation against gold annotations on SCAR (Yuan et al., 2023a) and ParallelPARC (Sultan et al., 2024) to LLM-generated candidates with baseline, embedding-based, and re-ranking selection strategies.

Each stage in our proposed pipeline corresponds to a research question:

**RQ1: Source Finding.** Can LLMs identify suitable source systems for a given target in both closed and open settings, moving beyond predefined candidate pools?

**RQ2: Sub-concept Generation.** Can LLMs reliably generate sub-concept mappings given a target-

source pair, and at which pipeline stages do sub-concepts improve performance?

**RQ3: Explanation Generation.** How well can LLMs produce full analogical explanations, and what input configurations yield the highest quality?

**RQ4: Evaluation.** How well does LLM-as-a-judge evaluation align with human judgments of analogy quality? Moreover, which generation model is the best at source finding in open settings?

All code and data are publicly available at: <https://github.com/Myriam2002/Toward-Usable-Scientific-Analogies>.

## 2 Related Work

**Source Finding.** Prior approaches to source finding rely primarily on embedding-based retrieval within closed candidate pools. Yuan et al. (2023a) use greedy and Kuhn-Munkres matching but focus on mapping given concepts rather than discovering new sources. Similarly, Yuan et al. (2023b) employ embedding similarity, but focus only on simple one- or two-relation analogies. Jiayang et al. (2023) show that LLMs are easily distracted by surface similarity, achieving only 30% accuracy compared to 85% for humans, while Ye et al. (2024) report substantial performance gaps on longer analogies. Li et al. (2024) further show that free generation outperforms embedding-based retrieval for analogies in the history domain, suggesting fundamental limitations of similarity-based methods. However, open-ended source discovery, where models must identify suitable sources without predefined candidates, remains largely unexplored.

**Sub-concept Generation.** Datasets such as SCAR (Yuan et al., 2023a) and ParallelPARC (Sultan et al., 2024) provide rich sub-concept annotations linking source and target systems. However, these annotations are primarily used for dataset construction rather than as inputs to generation models. Yuan et al. (2023a) show that sub-concept matching is critical for analogy quality, yet models achieve only 62% accuracy compared to 86% for humans, even when sub-concepts on both sides are provided. Moreover, evaluation is restricted to closed settings with fully specified inputs. The ability of LLMs to generate sub-concepts in open settings and their impact across pipeline stages has not been studied.

**Explanation Generation.** A well-matched source and accurate sub-concept mappings are necessary but not sufficient, as the explanation

is what ultimately determines whether a learner grasps the analogy. Producing explanations that capture *why* an analogy holds remains particularly challenging. Zhai et al. (2024) separate concept generation and explanation generation, finding that while LLMs approach human performance on concept generation, they struggle to produce explanations with sufficient relational depth. Bhavya et al. (2024) similarly report that model-generated explanations often lack coherence or drift from the intended analogy. Overall, explanation generation remains underexplored, particularly regarding how input representations affect quality.

**Evaluation.** Evaluating analogy quality remains challenging. Lexical metrics such as ROUGE and BLEURT fail to capture relational soundness (Zhai et al., 2024), while more structured evaluation frameworks still fall short of fully capturing analogy quality (Bhavya et al., 2024). Human evaluation, although more reliable, is costly (Jiayang et al., 2023; Czinczoll et al., 2022). LLM-as-a-judge methods offer a scalable alternative (Zhai et al., 2024), but their alignment with human judgments in analogy-specific settings remains mediocre.

### 3 Methodology

We present a systematic evaluation of the proposed modular pipeline for educational analogy generation, analysing how model choice and input configuration influence performance at each stage. The pipeline is decomposed into four stages aligned with our research questions: source finding (RQ1), sub-concept generation (RQ2), explanation generation (RQ3), and evaluation (RQ4). Figure 2 provides an overview of the experimental setup. To illustrate the full pipeline, consider the example in Figure 1, where the target system *atom* is explained using the source *solar system*. In the source-finding stage, candidate source systems are retrieved based on their similarity to the target. Next, sub-concept generation aligns components such as *nucleus* → *sun* and *electrons* → *planets*. The explanation generation stage then produces a natural-language explanation linking these correspondences. Finally, the analogy is evaluated based on coherence, mapping soundness, and explanatory power. This example demonstrates how each stage contributes to the final analogy.

### 3.1 Datasets

We use two datasets with explicit sub-concept annotations, enabling controlled evaluation across pipeline stages.<sup>1</sup>

**SCAR** (Yuan et al., 2023a) comprises 400 system-level analogy instances across 13 scientific domains, created by human annotators. Each instance pairs a target system with a source system and is annotated with background descriptions, multiple sub-concept mappings, and explanations for these mappings (1,614 total annotated mappings). A representative example is provided in Appendix A.

**ParalleIPARC** (Sultan et al., 2024) extends the ProPara dataset (Tandon et al., 2018) by transforming scientific-process descriptions into analogy pairs through a pipeline of LLM generation, automatic labelling, and human validation. It introduces 310 gold analogies, each annotated with relational alignments between corresponding properties in the format (entity, verb, entity) mapped to (entity, verb, entity). Please refer to Appendix A for an example.

### 3.2 Models

**Closed Setting.** For source retrieval in the closed setting, we evaluate a diverse set of embedding models: OpenAI’s embedding-small and embedding-large,<sup>2</sup> E5-large,<sup>3</sup> BGE-large,<sup>4</sup> MPNet-base,<sup>5</sup> MiniLM-L6,<sup>6</sup> and SPECTER.<sup>7</sup> These models cover complementary paradigms: proprietary general-purpose embeddings (OpenAI), open-source retrieval-oriented models (E5-large, BGE-large), sentence-similarity baselines (MPNet-base, MiniLM-L6), and a domain-specialised scientific encoder (SPECTER), enabling comparison across models that differ in scale, training objective, and domain specificity.

**Open Setting.** For open-ended generation across all three stages, we use the DSPy framework (Khat-tab et al., 2022, 2024) for consistent input/output

<sup>1</sup>Throughout the paper, we adopt the field names from SCAR for clarity, and align corresponding fields from other datasets to match this structure.

<sup>2</sup><https://platform.openai.com/docs/guides/embeddings>

<sup>3</sup><https://huggingface.co/intfloat/e5-large>

<sup>4</sup><https://huggingface.co/BAAI/bge-large-en>

<sup>5</sup><https://huggingface.co/sentence-transformers/all-mpnet-base-v2>

<sup>6</sup><https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>

<sup>7</sup><https://huggingface.co/allenai/specter>

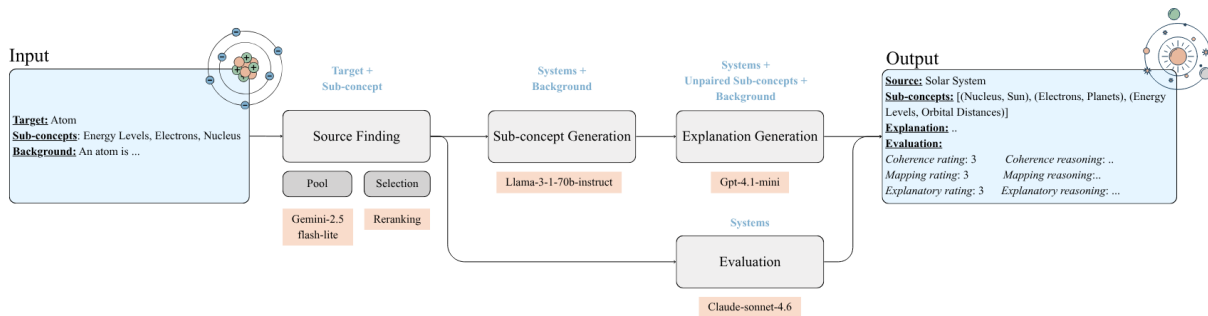


Figure 2: Overview of the proposed modular pipeline for educational analogy generation. Given a target system with its sub-concepts and background description, the framework proceeds through four stages: (1) source finding, where a pool of candidate sources is generated and selected; (2) sub-concept generation; (3) explanation generation; and (4) evaluation using an external LLM-as-a-judge. Each stage operates under input configurations tailored to its requirements, as determined by our empirical results. The output illustrates the final generated analogy, including aligned sub-concepts, explanation, and evaluation scores.

interfaces. We evaluate 12 LLMs across six families: OpenAI’s GPT-4.1 (mini, nano), GPT-OSS (20B, 120B), Meta’s LLaMA 3.1 (8B, 70B, 405B) Instruct, Google’s Gemini 2.5 Flash-Lite, xAI’s Grok-4-Fast, DeepSeek-R1, and Alibaba’s Qwen3 (14B, 32B). Models were selected based on OpenRouter’s public rankings in academia and science,<sup>8</sup> balancing benchmark performance and cost-efficiency. All models were run at temperature 0.2 for consistency. Models were accessed via OpenAI, OpenRouter, and DeepInfra APIs.<sup>9</sup>

### 3.3 Source Finding (RQ1): Pool & Selection

#### Closed Setting: Embedding-based Retrieval.

Following prior work (Czinczoll et al., 2022; Yuan et al., 2023a), we embed all unique source systems from SCAR and ParallelPARC using the seven models from Section 3.2 and retrieve the top-20 candidates per target via cosine similarity. We test four input configurations with increasing context: (1) **target only** to assess surface-level retrieval; (2) **+background** to test whether definitional context helps; (3) **+sub-concepts** to assess the role of structural alignment; and (4) **sub-concepts & background** to evaluate the effect of combining both.

To go beyond similarity-based ranking, we introduce an LLM reranking stage: the top-10 candidates from the best-performing embedding configuration are passed to LLaMA-3.1-70B-Instruct, which selects the top-3 (Appendix B). We evaluate five reranking configurations: (1) **target only** as a baseline; (2) **+background** to test definitional

context; (3) **+gold sub-concepts** to assess structural alignment with gold annotations; (4) **+sub-concepts & background** to combine both signals; and (5) **+generated sub-concepts** to test whether generated sub-concepts can substitute for gold annotations in downstream selection (see Subsection 3.4).

We evaluate retrieval using  $Hit@K$  (Eq. 1), which measures whether at least one gold source appears in the top- $K$  results. This evaluation directly measures the quality of the retrieval component by assessing whether embedding models successfully retrieve appropriate source systems for a given target.

$$Hit@K = \frac{1}{N} \sum_{i=1}^N \mathbb{I}(\text{rank}_i \leq K) \quad (1)$$

**Open Setting: LLM-based Generation.** In the open setting, each of the 12 LLMs generates 20 ranked candidate sources from scratch, evaluated on SCAR only due to its educational relevance and computational constraints. Following closed-setting results, we test two input configurations: (1) **target only** and (2) **+sub-concepts**.

Since multiple candidates are generated, we evaluate three selection strategies: (1) **baseline top-1**, selecting the first output from an LLM; (2) **embedding-based top-1**, selecting the candidate with the highest cosine similarity to the target using `text-embedding-3-small`, strong in closed-setting retrieval and cost-efficient; and (3) **re-ranker top-1**, applying the LLM-based reranking framework from Subsection 3.3 with the **name & subconcepts** configuration, selected based on its strong performance in the closed setting.

<sup>8</sup><https://openrouter.ai/rankings>

<sup>9</sup>OpenAI: <https://openai.com>, OpenRouter: <https://openrouter.ai>, DeepInfra: <https://deepinfra.com>

We report both *exact Hit@K* (Eq. 1) and *semantic Hit@K*. A generated source is considered correct if its embedding similarity to any gold source exceeds 0.374, corresponding to the upper tertile of pairwise gold similarities in SCAR (see Appendix C for more details). To avoid evaluation bias, we use all-MiniLM-L6-v2 for semantic Hit@K scoring rather than text-embedding-3-small, which is used in the embedding-based Top-1 selection step. Using the same model for both selection and evaluation might unfairly favour that configuration over others.

### 3.4 Sub-concept Generation (RQ2)

Before tackling the harder generation task, we revisit sub-concept matching – where models align given sub-concepts rather than produce new ones – to establish how well current LLMs capture structural correspondences and how far the field has advanced since Yuan et al. (2023a). Figure 3 highlights the distinction between the two tasks.

Inputs	Sub-concept Matching	Sub-concept Generation
Target System	Atom	
Target Sub-concept	[Energy levels, Electrons, Nucleus]	[Energy levels, Electrons, Nucleus]
Source System	Solar System	
Source Sub-concept	[Planets, Orbital Distance, Sun]	?
Output	[(Nucleus, Sun), (Electrons, Planets), (Energy Levels, Orbital Distances)]	[(Nucleus, Sun), (Electrons, Planets), (Energy Levels, Orbital Distances)]

Figure 3: Sub-concept matching and generation stage (RQ2). Given a target system (e.g., *atom*) and a source system (e.g., *solar system*), the model performs sub-concept matching using gold sub-concepts, and sub-concept generation when source sub-concepts are unavailable, producing aligned sub-concept pairs.

**Sub-concept Matching.** We re-evaluate sub-concept matching on SCAR, using its reported system accuracy (62% for GPT-4 vs. 86% for human annotators) as a reference point. Models receive randomly shuffled lists of target and source sub-concepts and must identify the correct correspondences. Then, we test two scenarios: (1) **without background**, where models rely only on sub-concept labels to recognise structural alignments; and (2) **with background** (see Listing 1 in Appendix A). We compare performance across the 12 LLMs to determine if the gap between model and human performance can be narrowed. For evaluation, we use *System Accuracy* (Yuan et al., 2023a), which assigns 1 if all pairs are correct and 0 other-

wise.

**Sub-concept Generation.** To test whether LLMs can generate source sub-concepts in open settings, we test two configurations: given the target system, its sub-concepts, and the source system name, models must produce the corresponding source sub-concepts either (1) **without background** or (2) **with background**.

To evaluate, we use *Semantic Match Accuracy* (SMA). It uses cosine similarity between embeddings built with all-MiniLM-L6-v2.<sup>10</sup> A fixed 0.7 threshold is adopted as a conservative heuristic for MiniLM embeddings, balancing semantic sensitivity with false-positive robustness.

$$SMA = \frac{\text{count}(\text{similarity} \geq 0.7)}{\text{total concepts}} \quad (2)$$

### 3.5 Explanation Generation (RQ3)

The third stage evaluates whether LLMs can articulate why an analogy holds, given a target and source. Each model generates a natural-language explanation. To isolate the contribution of different input signals, we test six configurations: (S1) **names only**, (S2) **+background**, (S3) **+unpaired sub-concepts**, (S4) **+unpaired sub-concepts & background**, (S5) **+paired mappings**, and (S6) **+paired mappings & background** (see the prompts in Appendix D). Throughout the results, we refer to these configurations as S1–S6 for brevity.

We evaluate generated explanations against gold references using cosine similarity over all-mpnet-base-v2 (Reimers and Gurevych, 2019) embeddings known for their ability to compare sentences to each other, capturing semantic equivalence beyond surface variation, and then we compare performance across input settings.

### 3.6 Evaluation (RQ4)

**LLM-as-a-judge.** To close the pipeline, we turn to evaluation – employing Claude Sonnet 4.6<sup>11</sup> as an external LLM judge, independent of all 12 pipeline models to reduce evaluation bias. Analogies are scored on three dimensions (using 1–3 scale): **coherence** (does the pairing make intuitive sense?), **mapping soundness** (are source–target

<sup>10</sup>all-MiniLM-L6-v2:<https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>

<sup>11</sup><https://openrouter.ai/anthropic/claude-sonnet-4.6>

correspondences structurally valid?), and **explanatory power** (does the analogy help a learner understand the target?). We use few-shot prompting with 10 calibrated examples spanning the full score range, as preliminary experiments without examples yielded less stable judgments. The full prompt is provided in Appendix E. We apply this judge across all 12 models to identify which one produces the highest-quality analogies overall.

**Human Evaluation.** To validate the LLM-based ratings, we collect human judgments from seven graduate students in AI and Business.<sup>12</sup> Prior to annotation, participants were provided with detailed instructions describing the task, including definitions of the three evaluation dimensions (coherence, mapping soundness, and explanatory power), along with illustrative examples to calibrate their understanding. Annotators evaluate 45 analogy pairs – 15 target concepts across five domains (check Appendix J for all analogies tested), each paired with three candidate sources – scoring each on the same three dimensions used by the LLM judge. In addition to scoring individual dimensions, annotators rank candidate analogies based on overall learning usefulness, capturing whether the analogy is intuitive and helpful for understanding the target concept. This ranking provides a qualitative, learner-oriented perspective that complements the dimension-level scores, yielding a total of 165 annotated datapoints.<sup>13</sup>

We measure inter-annotator reliability using Krippendorff’s  $\alpha$  (Krippendorff, 2011) for ordinal ratings, Kendall’s  $W$  (Kendall, 1990) for ranking concordance across annotators, and Spearman’s  $\rho$  to compare human scores against LLM judge scores.

## 4 Results and Analysis

### 4.1 Source Finding (RQ1)

**Closed Setting.** Embedding-based retrieval shows consistent but dataset-dependent trends. On SCAR, adding sub-concepts yields the largest gains: OpenAI-Large improves from 0.710 to 0.792 and OpenAI-Small from 0.677 to 0.780, with the average increase from 0.557 to 0.673. Adding background provides no

<sup>12</sup>Annotation form: <https://myriam2002.github.io/analogy-annotation/>

<sup>13</sup>Confidence scores were collected for ranking but not used in the current analysis; incorporating them to weigh annotator reliability is left for future work.

benefit (avg. 0.629), likely because sub-concepts already capture structural alignment, and extra context introduces noise. On ParallelPARC, performance is near-saturated across settings (name-only avg. 0.876), with sub-concepts + background performing best (OpenAI-Large: 0.942, OpenAI-Small: 0.932, avg. 0.908), as its backgrounds provide richer relational signal than SCAR descriptions. Across both datasets, OpenAI embeddings consistently lead, while SPECTER lags (SCAR avg. 0.437; ParallelPARC avg. 0.781), reflecting weaker alignment with the task. Full results are in Appendix I.

For selection, we compare five configurations: re-ranking with names only, background, gold sub-concepts, gold sub-concepts + background, generated sub-concepts, and embedding-only retrieval. Re-ranking with gold sub-concepts achieves the best Hit@1 (24.5%), followed by background (22.2%), gold + background (20.8%), embedding-only (19.5%), generated sub-concepts (18.2%), and names only (17.5%). At Hit@2 and Hit@3, embedding-only performs best (30.5%, 40.0%), while gold sub-concepts rank third (31.8%, 35.8%), indicating that embeddings better retrieve candidates, whereas LLM re-ranking excels at top-1 selection. Generated sub-concepts outperform the name-only baseline (18.2% vs 17.5% at Hit@1), showing that the pipeline produces meaningful structure without gold annotations, though a gap remains. Overall, embeddings narrow the search space, while LLM reasoning over sub-concepts is key for precise selection.

**Open Setting.** Adding sub-concepts to the generation prompt yields only marginal gains (avg. 0.4–1.6 pp across all K and models), suggesting limited benefit compared to their substantial impact in retrieval, while incurring additional prompt cost, making target-only prompting the more practical default. The two exceptions are LLaMA-405B and Qwen3-32B, which show a slight decrease when sub-concepts are added. In absolute terms, the strongest Hit@20 results are achieved under the target with sub-concepts mode, with Gemini-2.5 (75.4%) and GPT-OSS-120B (75.3%) leading. However, top models cluster within a few percentage points of each other, suggesting that scale alone does not determine generation quality. Per-model results are in Appendix H.

Model	Matching		Generation	
	+BG	-BG	+BG	-BG
Grok-4	0.76	<b>0.78</b>	0.42	0.31
DeepSeek-R1	0.70	0.76	0.45	0.33
GPT-OSS-120B	0.69	0.72	0.34	0.27
Qwen3-14B	0.68	0.70	0.42	0.31
GPT-OSS-20B	0.66	0.68	0.39	0.28
GPT-4.1-mini	0.66	0.68	0.42	0.31
Qwen3-32B	0.65	0.71	0.40	0.31
Llama-70B	0.57	0.60	<b>0.47</b>	0.32
Llama-405B	0.56	0.62	<b>0.47</b>	0.33
Gemini-2.5	0.54	0.64	0.42	0.29
Llama-8B	0.40	0.46	0.44	0.27
GPT-4.1-nano	0.37	0.46	0.34	0.27

Table 1: System accuracy (Matching) and semantic match accuracy (Generation) with (+BG) and without (-BG) background descriptions. Bold indicates the best score per column. Gray shading indicates the higher-scoring mode per task for each model. Models are sorted by matching accuracy.

## 4.2 Sub-concept Generation (RQ2)

Table 1 contrasts sub-concept matching and generation across models. For matching, performance is slightly higher *without* background descriptions (avg. 0.650 vs. 0.603), suggesting additional context hinders alignment. Grok-4 leads at 0.78, surpassing the prior state-of-the-art of 0.62 (Yuan et al., 2023a), with gains attributed to model choice and prompt design (Appendix F). GPT-4.1-nano performs worst (0.37) despite not being the smallest model. These results motivate the harder generation setting explored next.

For generation, background descriptions provide a clear benefit, raising average semantic match accuracy from 0.298 to 0.412. The LLaMA family leads, with LLaMA-3.1-70B and LLaMA-3.1-405B both reaching 0.47, while GPT-based variants consistently occupy the lower half. Notably, GPT-4.1-nano and GPT-OSS-120B (0.34) underperform the smaller GPT-OSS-20B (0.39), confirming that scale alone does not guarantee stronger generation.

Based on these findings, we adopt LLaMA-3.1-70B with background descriptions for all subsequent generation experiments.

## 4.3 Explanation generation (RQ3)

The most apparent pattern is the large jump in similarity when sub-concepts are introduced: average scores rise from 0.724 in S1-S2 to 0.854 with unpaired sub-concepts (S3-S4) and 0.830 with paired sub-concepts (S5-S6). By contrast, adding back-

ground descriptions has a negligible effect across all three setting pairs (at most +0.006), consistent with our earlier matching findings. The best setting overall is S5 (systems + paired sub-concepts).

GPT-4.1-Mini is the strongest model overall (mean 0.827) and is the best performer in four of the six settings (S3-S6), indicating strong generalisation across sub-concept conditions. In the system-only settings (S1-S2), where scores are uniformly low and differences between models are small (range  $\approx 0.03$ ), no single model stands out, with GPT-OSS-120B and LLaMA-70B marginally leading in S1 and S2, respectively. Check Appendix G for the detailed heatmap.

## 4.4 Evaluation (RQ4)

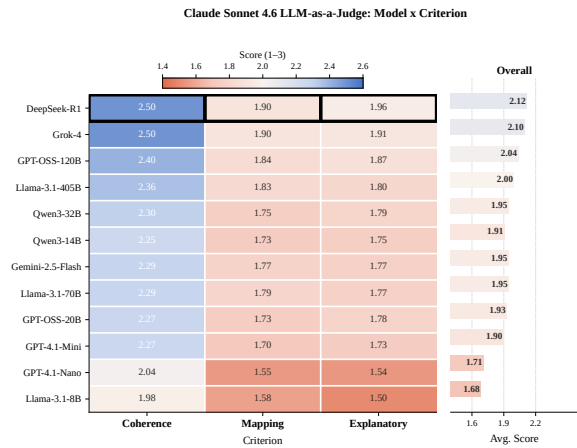


Figure 4: Per-criterion scores assigned by Claude Sonnet 4.6 acting as an LLM judge. Scores are on a 1–3 scale across three criteria: Coherence, Mapping, and Explanatory power. Black borders indicate the top-performing model per criterion.

**LLM-as-a-judge.** Figure 4 shows LLM-as-a-judge scores across generation models. DeepSeek-R1 and Grok-4 achieve the highest overall performance (2.12 and 2.10), while GPT-4.1-Nano and LLaMA-3.1-8B perform the worst (1.71 and 1.68). The remaining models cluster closely in the middle, indicating relatively small differences across most systems. Across all models, coherence scores are consistently higher than mapping and explanatory scores, suggesting that models often produce intuitively plausible analogies even when structural mappings are incomplete or explanations are less informative.

Table 2 compares selection strategies under LLM-as-a-judge evaluation. The baseline and re-ranking approaches achieve close average scores

Dimension	Baseline		Embeddings		Reranker	
	mean	std	mean	std	mean	std
Coherence	<b>2.463</b>	0.652	1.956	0.748	2.437	0.677
Mapping	1.839	0.560	1.550	0.583	<b>1.878</b>	0.561
Explanatory	<b>1.902</b>	0.622	1.484	0.581	1.894	0.640
Avg. Score	2.068	0.541	1.663	0.575	<b>2.069</b>	0.550

Table 2: Comparison of analogy quality across selection strategies using LLM-as-a-judge scores (1–3 scale).

(2.068 vs. 2.069), but differ in their strengths: the baseline yields slightly higher coherence, while re-ranking improves mapping quality. This indicates that LLM-based selection refines structural alignment without substantially affecting overall performance.

**Human Evaluation.** Inter-annotator agreement is measured using Krippendorff’s  $\alpha$ . Agreement is fair for *coherence* (0.382) and *explanatory power* (0.381), and lower for *mapping soundness* (0.342) – the most technically demanding dimension. Agreement is highest for the overall ranking (0.479) – where annotators rank the three candidate sources per target by overall learning usefulness – reflecting that relative ordering is easier to agree on than assigning consistent absolute scores across all three dimensions; for instance, two annotators may agree that source A is best, yet still differ on whether to rate its coherence a 2 or a 3. Appendix K provides detailed ranking statistics.

Spearman correlation ( $\rho$ ) between Claude and human annotators varies across both annotators and dimensions. Agreement is highest for *explanatory power* (mean  $\rho = 0.511$ , range 0.269–0.678) and lowest for *mapping soundness* (mean  $\rho = 0.342$ , range 0.124–0.507), consistent with LLM-as-a-judge trends. Coherence falls between these extremes (mean  $\rho = 0.368$ , range 0.130–0.581), reflecting moderate agreement on whether analogies are intuitively meaningful. Annotator-level variability is also notable: the weakest alignment consistently comes from the annotator whose academic background differed most from the rest of the group, which may be partially explained by the domain composition of the evaluation set, where five of the fifteen target concepts were drawn from the computer science domain.

Table 3 compares Claude’s implicit ranking of analogies with human rankings. In contrast to the dimension-level results, ranking agreement is generally stronger, with most annotators showing moderate-to-high correlation ( $\rho$  between 0.46 and

Annotator	$\rho$	p-value	Significant
ANN-6JJP46	0.474	0.0010	True
ANN-6LG7WY	0.705	0.0000	True
ANN-DUNVTH	0.307	0.0404	True
ANN-H2NTMS	0.694	0.0000	True
ANN-KKN85E	0.463	0.0014	True
ANN-Q777H6	0.128	0.4012	False
ANN-VM9J6N	0.694	0.0000	True

Table 3: Spearman correlation between Claude’s implicit ranking and human rankings.

0.70) and statistically significant results ( $p < 0.05$ ). The strongest alignment is observed for ANN-6LG7WY ( $\rho = 0.705$ ), while ANN-Q777H6 again shows no significant agreement. These results indicate that Claude more reliably captures relative preferences between analogies than assigning consistent absolute scores, suggesting that LLM-as-a-judge is better suited for comparative evaluation settings than for fine-grained scoring.

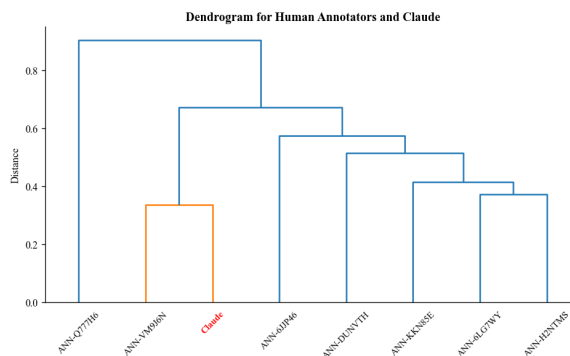


Figure 5: Hierarchical clustering of human annotators and Claude based on their scoring patterns across evaluation dimensions. Distances reflect dissimilarity in judgments (computed from Spearman correlations). Claude clusters closely with a subset of annotators (e.g., ANN-VM9J6N), while other annotators (e.g., ANN-Q777H6) form more distant branches.

Figure 5 further examines agreement patterns through hierarchical clustering of annotators and Claude. The dendrogram reveals that Claude groups closely with a subset of annotators, suggesting similar evaluation behavior, while others form more distant clusters, indicating systematic differences in scoring. In particular, annotators with lower correlation scores (e.g., ANN-Q777H6) appear as outliers, consistent with the results in Table 3. This analysis reinforces that, although Claude aligns well with the majority of annotators, human judgments themselves are not fully consistent, highlighting the inherent subjectivity of analogy evaluation.

## 5 Conclusion

We presented a modular pipeline for educational analogy generation decomposed into source finding, sub-concept generation, explanation generation, and evaluation, systematically evaluated across 12 LLMs and 7 embedding models on SCAR and ParallelPARC. A central finding is that the best model and configuration differ across stages: Grok-4-Fast leads sub-concept matching without background descriptions, while LLaMA-3.1-70B with background is best for generation; GPT-4.1-Mini with unpaired sub-concept mappings achieves the strongest explanation quality; and OpenAI large embeddings with sub-concepts lead closed-setting retrieval, whereas the target-only configuration is more practical in the open setting. Sub-concept grounding consistently improves retrieval precision and explanation quality but provides limited benefit in open-ended source generation. For evaluation, DeepSeek-R1 and Grok-4-Fast produce the highest-quality analogies, and Claude Sonnet 4.6 aligns more reliably with human rankings than absolute scores, suggesting LLM-as-a-judge is better suited to comparative settings. Taken together, these findings demonstrate that no single model dominates across all stages, reinforcing the necessity of a modular, stage-aware approach to analogy generation.

### Limitations

Several limitations of this work warrant acknowledgment. First, all experiments are conducted exclusively on English-language data, leaving the pipeline’s behaviour on morphologically richer or structurally different languages unexplored. In such settings, both embedding-based retrieval and generation quality may differ, and evaluation metrics such as Semantic Match Accuracy (SMA) may require recalibration.

Second, the 0.7 cosine similarity threshold used for SMA is a fixed heuristic selected for MiniLM-L6 embeddings rather than empirically validated. Its suitability for more abstract or diverse sub-concepts remains unclear, and alternative thresholds could meaningfully affect generation results.

Third, while each pipeline stage is evaluated independently, we do not evaluate the pipeline end-to-end. In a full pipeline setting, errors are likely to accumulate and propagate across stages. For example, an incorrect source selected during the retrieval stage would lead to flawed sub-concept mappings

and explanations downstream. This makes source finding a particularly critical stage, as early errors can disproportionately affect final analogy quality. Future work should investigate strategies to mitigate such compounding effects, such as iterative refinement, confidence-aware filtering, or alternative candidate selection mechanisms.

Fourth, the current architecture relies primarily on dense embedding models for retrieval and does not explore hybrid search strategies. Combining dense and sparse retrieval methods, or incorporating re-ranking models more extensively, could improve the robustness and quality of source selection, particularly in open-ended settings.

Fifth, analogy quality is inherently subjective: an analogy that is effective for one learner may not be equally meaningful for another. While this subjectivity is partially reflected in moderate inter-annotator agreement, our evaluation does not explicitly model user-dependent preferences. This limits the ability to assess how well generated analogies generalise across diverse learner populations. Future work should investigate how to construct learner profiles and incorporate them into the pipeline to generate personalised analogies tailored to factors such as cultural background, age, prior knowledge, and individual interests or hobbies.

Sixth, all pipeline stages are evaluated in a single-pass setting. We do not measure output variance across runs, which means that reported results reflect point estimates rather than stable averages. Although a low temperature (0.2) is used, some variability is expected, particularly for smaller models. Moreover, since ParallelPARC was used in only one stage of the pipeline (embedding-based retrieval) and the remaining stages relied solely on SCAR, the generalizability of the results across datasets may be limited.

Finally, the human evaluation study involves six out of seven graduate-level annotators from a single institution, which may introduce demographic and disciplinary homogeneity. Inter-annotator agreement is moderate, reflecting the inherent subjectivity of analogy evaluation and suggesting that a larger, more diverse annotator pool could yield different conclusions. In addition, open-setting source finding is evaluated only on SCAR due to computational constraints, limiting the generalisability of these findings to other domains of different difficulty.

## Ethical Considerations

We do not anticipate any ethical risks related to this study.

## References

- Briana Bettin and Linda Ott. 2023. Pedagogical prisms: toward domain isomorphic analogy design for relevance and engagement in computing education. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, pages 410–416.
- Bhavya Bhavya, Chris Palaguachi, Yang Zhou, Suma Bhat, and ChengXiang Zhai. 2024. Long-form analogy evaluation challenge. In *Proceedings of the 17th International Natural Language Generation Conference: Generation Challenges*, pages 1–16.
- David E Brown and John Clement. 1989. Overcoming misconceptions via analogical reasoning: Abstract transfer versus explanatory model construction. *Instructional science*, 18(4):237–261.
- Tamara Czinczoll, Helen Yannakoudakis, Pushkar Mishra, and Ekaterina Shutova. 2022. Scientific and creative analogies in pretrained language models. *arXiv preprint arXiv:2211.15268*.
- Stina Eriksson, Niklas Gericke, and Karin Thörne. 2024. Analogy competence for science teachers. *Studies in Science Education*, pages 1–29.
- Dedre Gentner. 1981. Are scientific analogies metaphors? Technical report.
- Cheng Jiayang, Lin Qiu, Tsz Ho Chan, Tianqing Fang, Weiqi Wang, Chunkit Chan, Dongyu Ru, Qipeng Guo, Hongming Zhang, Yangqiu Song, and 1 others. 2023. Storyanalogy: Deriving story-level analogies from large language models to unlock analogical understanding. *arXiv preprint arXiv:2310.12874*.
- Maurice G Kendall. 1990. *Rank Correlation Methods*, 5th edition. Oxford University Press.
- Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive NLP. *arXiv preprint arXiv:2212.14024*.
- Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2024. Dspy: Compiling declarative language model calls into self-improving pipelines.
- Jeongyeon Kim, Sangho Suh, Lydia B Chilton, and Haijun Xia. 2023. Metaphorian: Leveraging large language models to support extended metaphor creation for science writing. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, pages 115–135.
- Klaus Krippendorff. 2011. Computing krippendorff’s alpha-reliability. *Annenberg School for Communication Departmental Papers*.
- Nianqi Li, Siyu Yuan, Jiangjie Chen, Jiaqing Liang, Feng Wei, Zujie Liang, Deqing Yang, and Yanghua Xiao. 2024. Past meets present: Creating historical analogy with large language models. *arXiv preprint arXiv:2409.14820*.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-bert: Sentence embeddings using siamese bert-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Zekai Shao, Siyu Yuan, Lin Gao, Yixuan He, Deqing Yang, and Siming Chen. 2025. Unlocking scientific concepts: How effective are llm-generated analogies for student understanding and classroom practice? In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–19.
- Oren Sultan, Yonatan Bitton, Ron Yosef, and Dafna Shahaf. 2024. Paralleparc: A scalable pipeline for generating natural-language analogies. *arXiv preprint arXiv:2403.01139*.
- Niket Tandon, Bhavana Dalvi Mishra, Joel Grus, Wentau Yih, Antoine Bosselut, and Peter Clark. 2018. Reasoning about actions and state changes by injecting commonsense knowledge. *arXiv preprint arXiv:1808.10012*.
- Xiao Ye, Andrew Wang, Jacob Choi, Yining Lu, Shreya Sharma, Lingfeng Shen, Vijay Tiyyala, Nicholas Andrews, and Daniel Khashabi. 2024. Analobench: benchmarking the identification of abstract and long-context analogies. *arXiv preprint arXiv:2402.12370*.
- Siyu Yuan, Jiangjie Chen, Xuyang Ge, Yanghua Xiao, and Deqing Yang. 2023a. Beneath surface similarity: Large language models make reasonable scientific analogies after structure abduction. *arXiv preprint arXiv:2305.12660*.
- Siyu Yuan, Jiangjie Chen, Changzhi Sun, Jiaqing Liang, Yanghua Xiao, and Deqing Yang. 2023b. Analogykb: Unlocking analogical reasoning of language models with a million-scale knowledge base. *arXiv preprint arXiv:2305.05994*.
- ChengXiang Zhai, Suma Bhat, and Heng Ji. 2024. Long-form analogy extraction, generation, and evaluation with large language models.
- Xiaofei Zhou, Yunfan Gong, Yushan Zhou, Yufei Jiang, and Zhen Bai. 2025. Co-design of analogical and embodied representations with children for child-centered ai learning experiences. *International Journal of Human-Computer Studies*, 199:103462.

## A Detailed Record of Datasets

Listing 1 and 2 shows a representative JSON example of the two datasets.

```
{
  "id": 3,
  "lang": "en",
  "system_a": "Respiratory system",
  "system_b": "Engine",
  "mappings": [
    ["oxygen", "fuel"],
    ["lungs", "combustion chamber"],
    ["breathing muscles", "piston"]
  ],
  "system_a_domain": "Biology",
  "system_b_domain": "Physics",
  "system_a_background": "The respiratory system ...",
  "system_b_background": "An engine or motor ...",
  "Explanation": [
    "Oxygen corresponds to fuel: ...",
    "Lungs correspond to the combustion chamber: ...",
    "Breathing muscles correspond to the piston: ..."
  ]
}
```

Listing 1: Representative JSON example from SCAR

```
{
  "sample_id": 1,
  "source_subject": "What causes a volcano to erupt?",
  "source_domain": "Natural Sciences",
  "target_domain": "Engineering",
  "target_subject": "What causes a boiler to explode?",
  "target_field": "Chemical Engineering",
  "relations": [
    "(magma, heats, underground water) like (steam, heats, liquid)",
    "(pressure, builds, inside the volcano) like (pressure, builds, inside the boiler)",
    "(magma, pushes, against the walls of the volcano) like (steam, pushes, against the walls of the
    ↪ boiler)"
  ],
  "source_paragraph": "When magma heats up underground water, pressure begins to build up inside the
  ↪ volcano, leading to an eruption.",
  "target_paragraph": "Steam heats the liquid inside the boiler, causing the pressure to build up
  ↪ until the boiler can no longer contain it.",
  "analogy_type": "far analogy",
  "sum_vote_analogy": 3.0
}
```

Listing 2: Representative JSON from ParallelPARC

## B Prompt for LLM Re-ranker

You are selecting the best analogous source concepts for a scientific analogy.

Your task:

1. Analyze the target concept and its properties
2. Review each candidate source and its generated analogous properties
3. Select the 3 candidates whose properties BEST map to the target properties

Selection criteria:

- Strong structural/functional correspondence between source and target properties
- The source concept should be familiar and help explain the unfamiliar target
- Prefer sources with clear, well-mapped properties over vague ones

Return the EXACT names of your top 3 selected candidates.

""

Listing 3: Prompt used to select the top-3 analogies from the top-10 potential sources. This was wrapped in DSPY function as instruction, alongside specific input and output variables.

## C Threshold Decision for Semantic Hit@K

To define a threshold for semantic Hit@K, we analyse the distribution of source-to-source similarity scores in the SCAR dataset. Specifically, we compare similarity values between *correct* source pairs (i.e., gold analogies) and *incorrect* pairs obtained by randomly shuffling sources.

Figure 6 shows that correct source pairs tend to exhibit higher similarity than incorrect ones, but with substantial overlap between the two distributions. This overlap indicates that a fixed high threshold (e.g., 0.5) would be overly restrictive, excluding many valid analogies that are structurally sound but not highly similar at the surface level.

To balance precision and coverage, we adopt a data-driven threshold of 0.374, corresponding to the upper tertile (top one-third) of similarity scores among gold source pairs. This choice captures a substantial portion of valid analogies while filtering out a large fraction of incorrect ones. As shown in the figure, this threshold lies above the mean similarity of incorrect pairs (0.180) and close to the mean of correct pairs (0.368), providing a principled trade-off between false positives and false negatives.

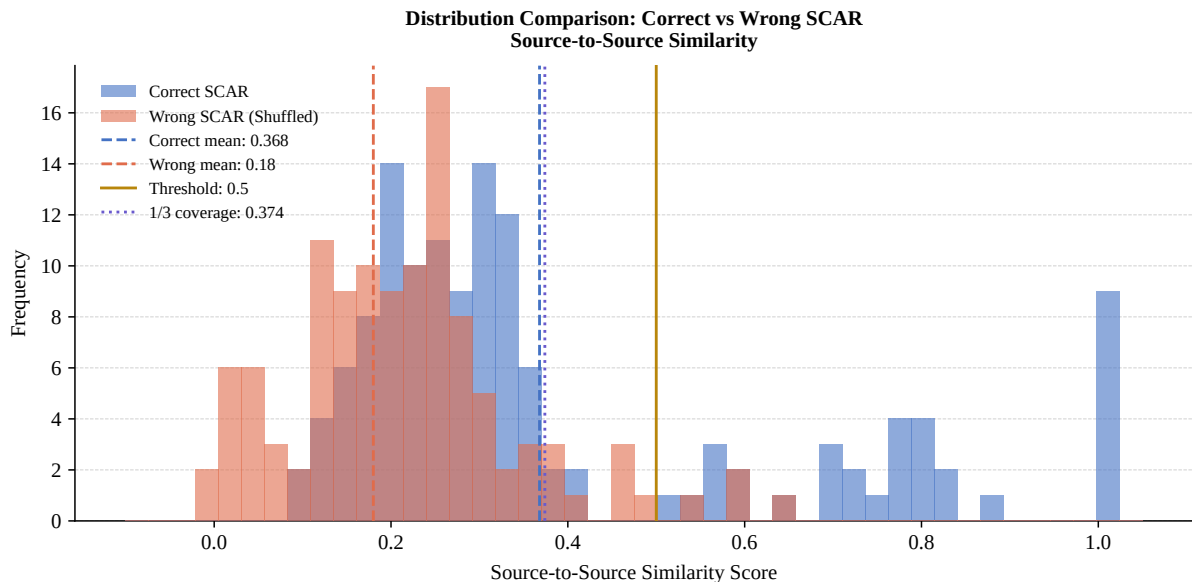


Figure 6: Distribution of source-to-source similarity scores in SCAR for correct (gold) and incorrect (shuffled) pairs. Dashed lines indicate the mean similarity for each group, while the dotted line shows the chosen threshold (0.374), corresponding to the upper tertile of gold similarities. A higher fixed threshold (e.g., 0.5) would exclude many valid analogies, motivating the use of a data-driven cutoff.

## D Explanation Generation Prompts

This appendix lists the DSPy signatures used for the six explanation-generation settings. Each signature corresponds to one input configuration.

**Setting 1: System Names Only** This configuration provides only the target and source system names.

```
class ExplGen_Base(dspy.Signature):
    """Generate an analogy explanation using system names only."""
    target: str = dspy.InputField(
        desc="Target system"
    )
    source: str = dspy.InputField(
        desc="Source system"
    )
```

```

)
explanation: str = dspy.OutputField(
    desc="Explanation of the analogy"
)

```

**Setting 2: System Names with Background Descriptions** This setting augments system names with background descriptions for both systems.

```

class ExplGen_BaseDesc(dspy.Signature):
    """Generate an analogy explanation using names and descriptions."""
    target: str = dspy.InputField(
        desc="Target system"
    )
    target_desc: str = dspy.InputField(
        desc="Background description of the target"
    )
    source: str = dspy.InputField(
        desc="Source system"
    )
    source_desc: str = dspy.InputField(
        desc="Background description of the source"
    )
    explanation: str = dspy.OutputField(
        desc="Explanation of the analogy"
    )

```

**Setting 3: System Names with Unpaired Sub-concepts** This configuration provides both systems with unpaired lists of sub-concepts.

```

class ExplGen_Sub(dspy.Signature):
    """Generate an analogy explanation using unpaired sub-concepts."""
    target: str = dspy.InputField(
        desc="Target system"
    )
    target_sub: list[str] = dspy.InputField(
        desc="Target sub-concepts"
    )
    source: str = dspy.InputField(
        desc="Source system"
    )
    source_sub: list[str] = dspy.InputField(
        desc="Source sub-concepts"
    )
    explanation: list[str] = dspy.OutputField(
        desc="Explanation after implicitly pairing sub-concepts"
    )

```

**Setting 4: System Names with Unpaired Sub-concepts and Descriptions** This setting extends Setting 3 by adding background descriptions for both systems.

```

class ExplGen_SubDesc(dspy.Signature):
    """Generate an analogy explanation using unpaired sub-concepts and descriptions."""
    target: str = dspy.InputField(
        desc="Target system"
    )
    target_desc: str = dspy.InputField(
        desc="Background description of the target"
    )
    target_sub: list[str] = dspy.InputField(
        desc="Target sub-concepts"
    )
    source: str = dspy.InputField(
        desc="Source system"
    )
    source_desc: str = dspy.InputField(
        desc="Background description of the source"
    )

```

```

source_sub: list[str] = dspy.InputField(
    desc="Source sub-concepts"
)
explanation: list[str] = dspy.OutputField(
    desc="Explanation after implicitly pairing sub-concepts"
)

```

**Setting 5: Paired Sub-concept Mappings** This configuration provides explicit one-to-one sub-concept alignments between the target and source systems.

```

class ExplGen_Pair(dspy.Signature):
    """Generate an analogy explanation using paired sub-concepts."""
    target: str = dspy.InputField(
        desc="Target system"
    )
    source: str = dspy.InputField(
        desc="Source system"
    )
    pairs: list[list[str]] = dspy.InputField(
        desc="Paired target-source sub-concepts"
    )
    explanation: list[str] = dspy.OutputField(
        desc="Explanation for each paired sub-concept"
    )

```

**Setting 6: Paired Sub-concept Mappings with Descriptions** This setting combines explicit sub-concept pairings with background descriptions for both systems.

```

class ExplGen_PairDesc(dspy.Signature):
    """Generate an analogy explanation using paired sub-concepts and descriptions."""
    target: str = dspy.InputField(
        desc="Target system"
    )
    target_desc: str = dspy.InputField(
        desc="Background description of the target"
    )
    source: str = dspy.InputField(
        desc="Source system"
    )
    source_desc: str = dspy.InputField(
        desc="Background description of the source"
    )
    pairs: list[list[str]] = dspy.InputField(
        desc="Paired target-source sub-concepts"
    )
    explanation: list[str] = dspy.OutputField(
        desc="Explanation for each paired sub-concept"
    )

```

## E Prompt for the LLM-as-a-Judge

You are an expert evaluator of scientific analogies.

Given a target concept and a chosen source analogy, evaluate whether this is a good analogy. A good analogy uses a FAMILIAR source concept to explain an UNFAMILIAR target concept through meaningful structural or functional parallels.

For each of the three dimensions below, first provide a brief reasoning explaining your assessment, then give the numeric score (1, 2, or 3).

ANALOGY\_COHERENCE: Does the pairing make intuitive sense?

- 3: The connection is immediately clear and natural -- most people would see it without explanation. The source and target share an obvious structural or functional parallel.
- 2: A meaningful connection exists but requires some explanation to see. The link is real but not self-evident; a sentence or two is needed to establish it.
- 1: No meaningful connection exists, or the pairing is random, forced, or misleading.

MAPPING\_SOUNDNESS: Could properties/mechanisms of the source map to the target?

- 3: Rich, consistent structural or functional correspondences exist.  
Multiple source properties map precisely onto target properties.  
The mapping holds across the main components of both domains.
- 2: Some valid mappings exist, but coverage is partial.  
Core correspondences work, but important aspects of the target are not represented, or some mappings are approximate or strained.
- 1: No valid mappings are possible, or the apparent mappings are entirely superficial or misleading. Source and target are fundamentally incompatible.

EXPLANATORY\_POWER: Would this analogy help a learner understand the target?

- 3: The analogy clearly illuminates the target concept and supports correct reasoning.  
A learner could use it to predict or explain target behavior.
- 2: The analogy provides partial insight with notable limitations.  
It conveys the general idea but cannot support deeper reasoning, or it risks creating minor misconceptions.
- 1: The analogy fails to aid understanding and would likely confuse or mislead a learner.

CALIBRATION EXAMPLES -- use these to anchor your scoring.

Score each dimension independently. The scale is 1-3 only:

3 = strong / clearly works

2 = partial / works with caveats

1 = doesn't work / poor or misleading

Example 1 (scores: 3 / 3 / 3)

target: "electric circuit" | analogy: "water flowing through pipes"

coherence=3:

The connection is immediately clear without any explanation needed.

A driving force pushes a substance through a constrained pathway in both cases -- this structural parallel is instantly visible.

mapping=3:

Multiple precise correspondences hold consistently:

voltage -> pressure | current -> flow rate | resistance -> pipe restriction

battery -> pump | closed loop -> closed pipe system

Relationships between variables (increase pressure -> increase flow) are preserved.

explanatory=3:

A learner can use the analogy to reason about Ohm's law qualitatively and make correct predictions about circuit behavior.

Example 2 (scores: 3 / 2 / 3)

target: "cell" | analogy: "factory"

coherence=3:

The analogy captures organized division of labor within a bounded system.

Most people immediately see why a cell resembles a factory.

mapping=2:

nucleus -> control center | ribosomes -> production workers

mitochondria -> energy supply | membrane -> boundary/security

explanatory=3:

Helps learners grasp coordination and specialization clearly.

Example 3 (scores: 2 / 2 / 2)

target: "mathematical function" | analogy: "machine"

coherence=2:

The shared idea (input -> transformation -> output) is real but requires explanation.

mapping=2:

input -> raw material, rule -> mechanism, output -> product

explanatory=2:

Useful for initial intuition but limited for deeper reasoning.

Example 4 (scores: 1 / 1 / 1)

target: "neural network" | analogy: "human brain"

coherence=1:  
Superficial similarity ("neurons") without structural alignment.

mapping=1:  
No valid correspondences across core mechanisms.

explanatory=1:  
Likely to mislead understanding.

Example 5 (scores: 1 / 1 / 1)  
target: "chemical reaction" | analogy: "a novel"

coherence=1:  
No meaningful connection.

mapping=1:  
No systematic correspondences.

explanatory=1:  
No instructional value.

Example 6 (scores: 3 / 1 / 2)  
target: "democracy" | analogy: "majority vote in a classroom"

coherence=3:  
Both involve majority-based decisions.

mapping=1:  
Institutional aspects have no counterpart.

explanatory=2:  
Partial understanding only.

Example 7 (scores: 2 / 3 / 3)  
target: "atom" | analogy: "solar system"

coherence=2:  
Spatial similarity exists but requires explanation.

mapping=3:  
nucleus -> sun | electrons -> planets

explanatory=3:  
Supports initial reasoning about structure.

Example 8 (scores: 3 / 2 / 1)  
target: "photosynthesis" | analogy: "solar-powered factory"

coherence=3:  
Energy conversion is clearly analogous.

mapping=2:  
sunlight -> energy supply | chloroplast -> factory

explanatory=1:  
Oversimplifies processes.

Example 9 (scores: 1 / 3 / 3)  
target: "compound interest" | analogy: "a snowball rolling downhill"

coherence=1:  
Domains differ fundamentally.

mapping=3:  
principal -> snowball | growth -> accumulation

explanatory=3:  
Clearly conveys accelerating growth.

```

Example 10 (scores: 3 / 1 / 2)
target: "ecosystem" | analogy: "a family"

coherence=3:
  Interdependence is clear.

mapping=1:
  No structural ecological correspondences.

explanatory=2:
  Partial intuition only.

```

Listing 4: Prompt used to evaluate the analogies from different settings on different dimensions. This was wrapped in DSPY function as instruction, alongside specific input and output variables.

## F Ablation results for Sub-concept Matching

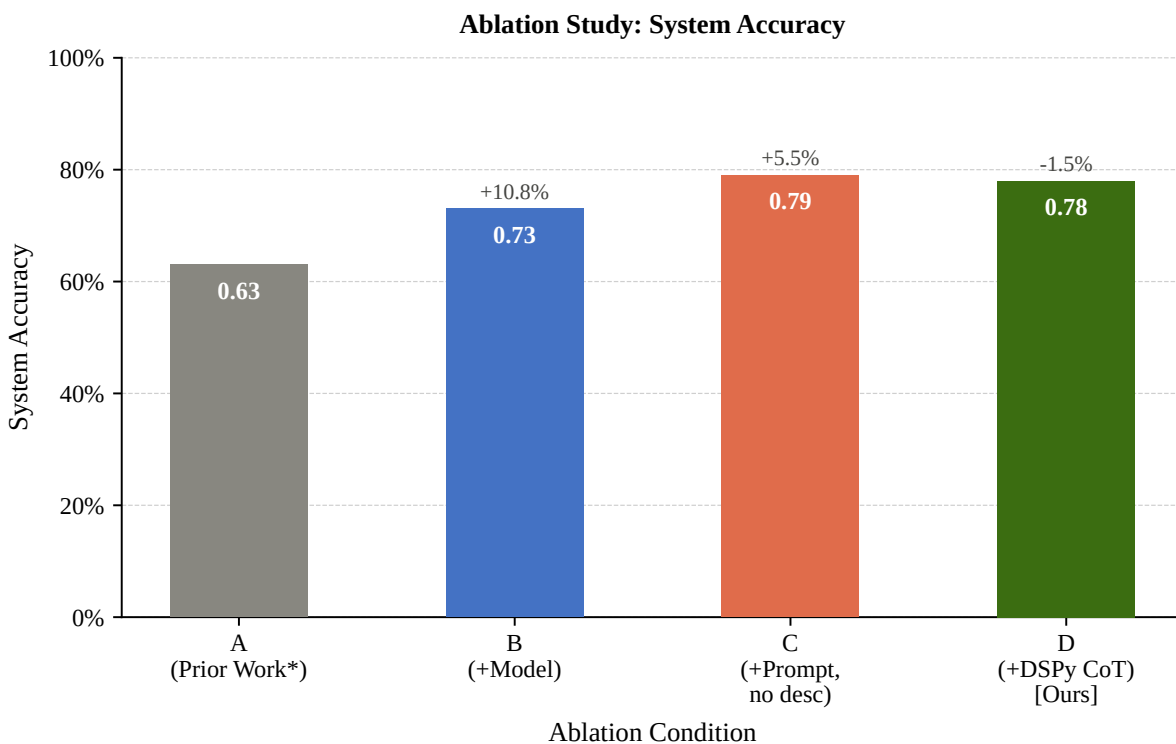


Figure 7: Ablation study on sub-concept matching system accuracy, isolating the contribution of each component relative to prior work (Yuan et al., 2023a). Upgrading the model to our best performing Grok-4 alone (B) yields the largest gain (+10.8%), followed by prompt redesign (C, +5.5%). Wrapping with DSPy Chain-of-Thought (D, our final system) introduces a marginal drop (-1.5%), confirming that automated prompt optimisation does not improve over the hand-crafted prompt for this task.

Figure 7 shows the ablation study to figure out what aspect actually led to the improved results.

## G Explanation Heatmap Results

Figure 8 presents SBERT similarity scores across twelve LLMs and six experimental settings for the explanation generation stage.

## H Open Setting Source Finding: Per-Model Hit@20 Comparison

Figure 9 presents per-model Hit@20 scores in the open setting, comparing target-only and target with sub-concept prompting across all twelve models.

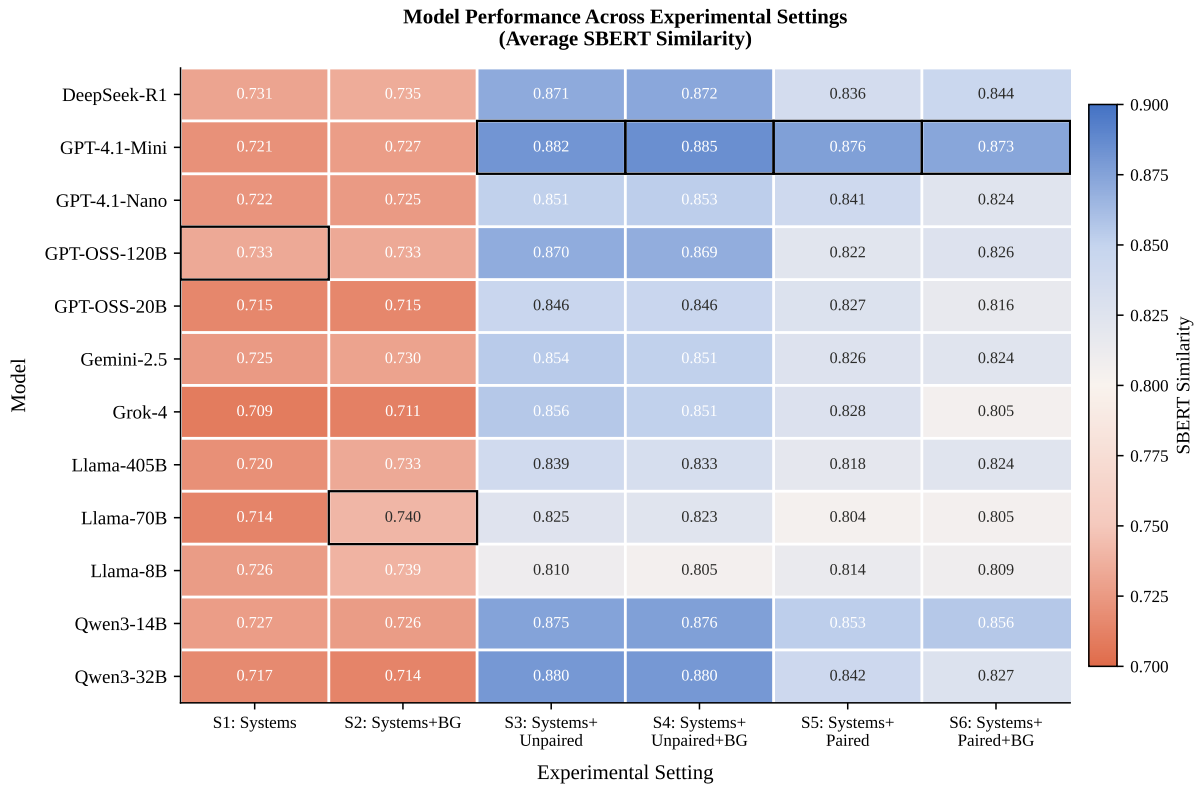


Figure 8: Average SBERT similarity scores across twelve models and six experimental settings. Settings S1-S2 provide only system-level context (without and with background descriptions, respectively), while S3-S4 additionally supply unpaired sub-concepts and S5-S6 supply paired sub-concepts. Black borders indicate the best-performing model per setting. GPT-4.1-Mini consistently leads in sub-concept-enriched settings (S3-S5), while DeepSeek-R1 and Qwen3 variants are competitive in paired settings (S5-S6). Scores rise substantially when sub-concepts are introduced (S3-S6 vs. S1-S2), suggesting that grounding the comparison in explicit sub-concept structure is a stronger driver of similarity than background context alone.

## I Embedding-based Retrieval Results

Figure 10 and 11 show the heatmap of the 7 embedding models across 4 input configurations.

## J Annotation Dataset

Table 4 presents the annotation dataset, comprising 15 target concepts spanning five domains, each paired with three source analogies evaluated under target-only and target with sub-concept conditions. Furthermore, Figures 12- 21 provide an overview of the web-based annotation form used in our human evaluation study. The form guides annotators through rating each of three candidate source analogies per target concept on three dimensions – coherence, mapping soundness, and explanatory power – on a 1–3 scale, before ranking the three sources by overall learning usefulness and reporting their confidence in the ranking.

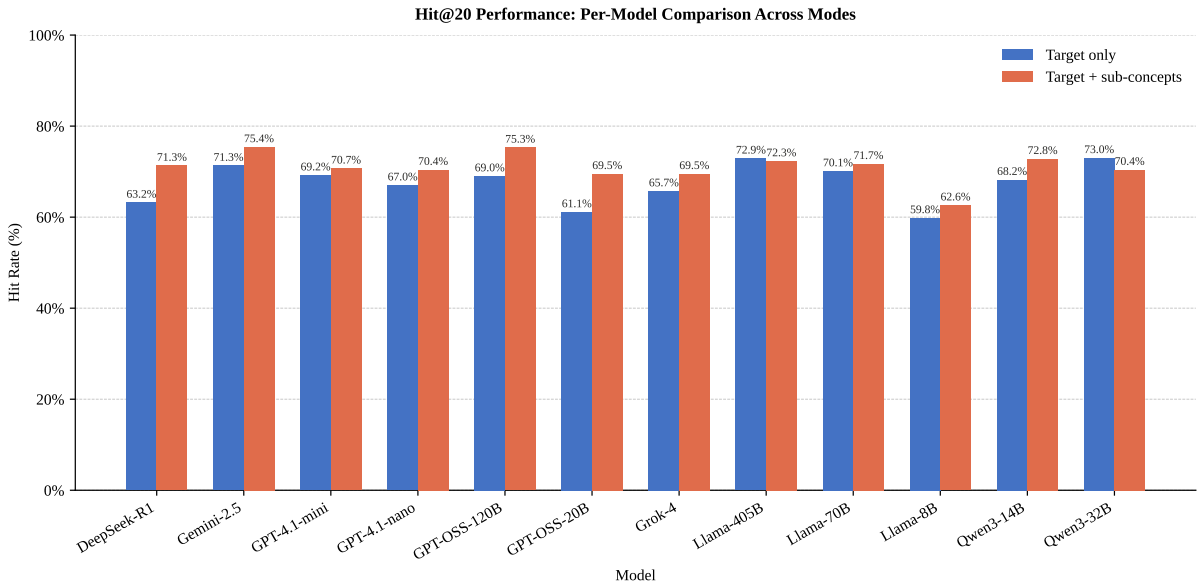


Figure 9: Per-model Hit@20 comparison in the open setting under Target-Only and Target + Sub-Concept modes. Results are reported using semantic matching and highlight the effect of incorporating sub-concepts on top-20 retrieval performance.

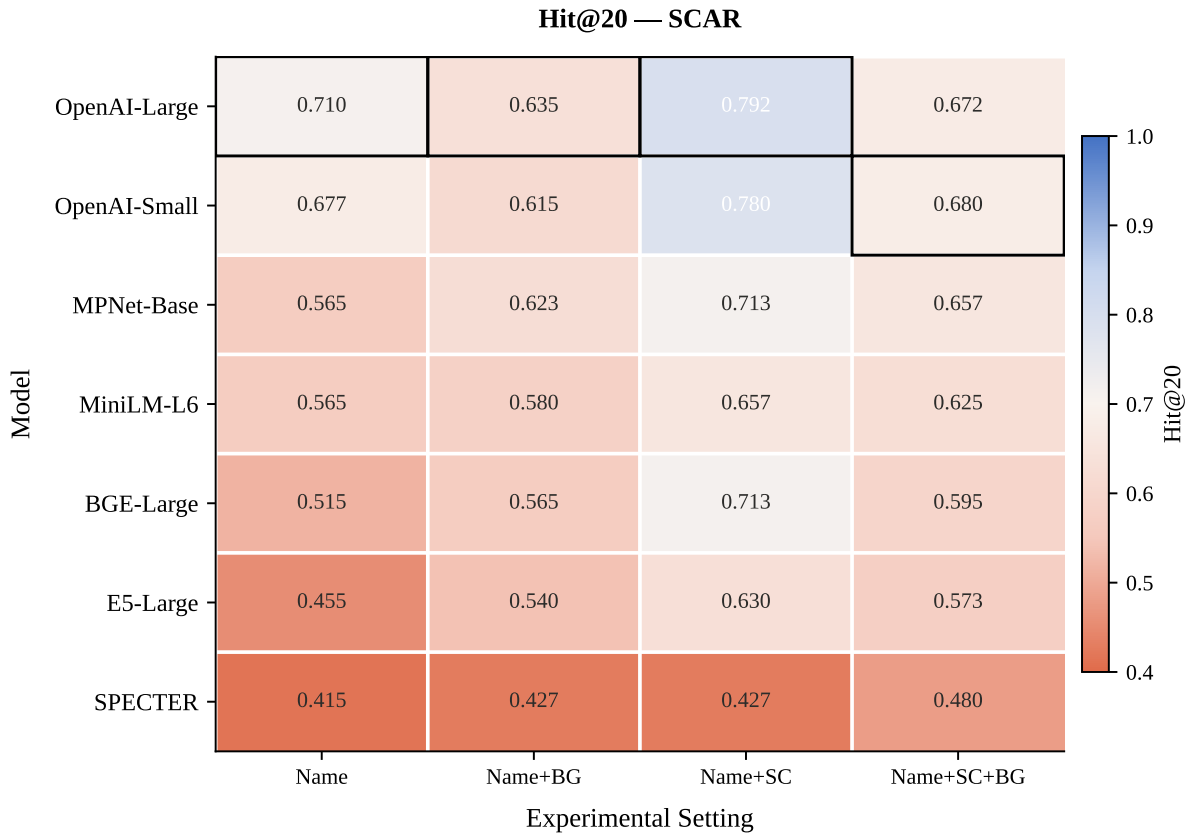


Figure 10: Hit@20 performance on the SCAR dataset across embedding models and input configurations. Incorporating sub-concepts consistently improves retrieval compared to name-only inputs, while adding background provides mixed benefits. OpenAI embeddings achieve the strongest overall performance.

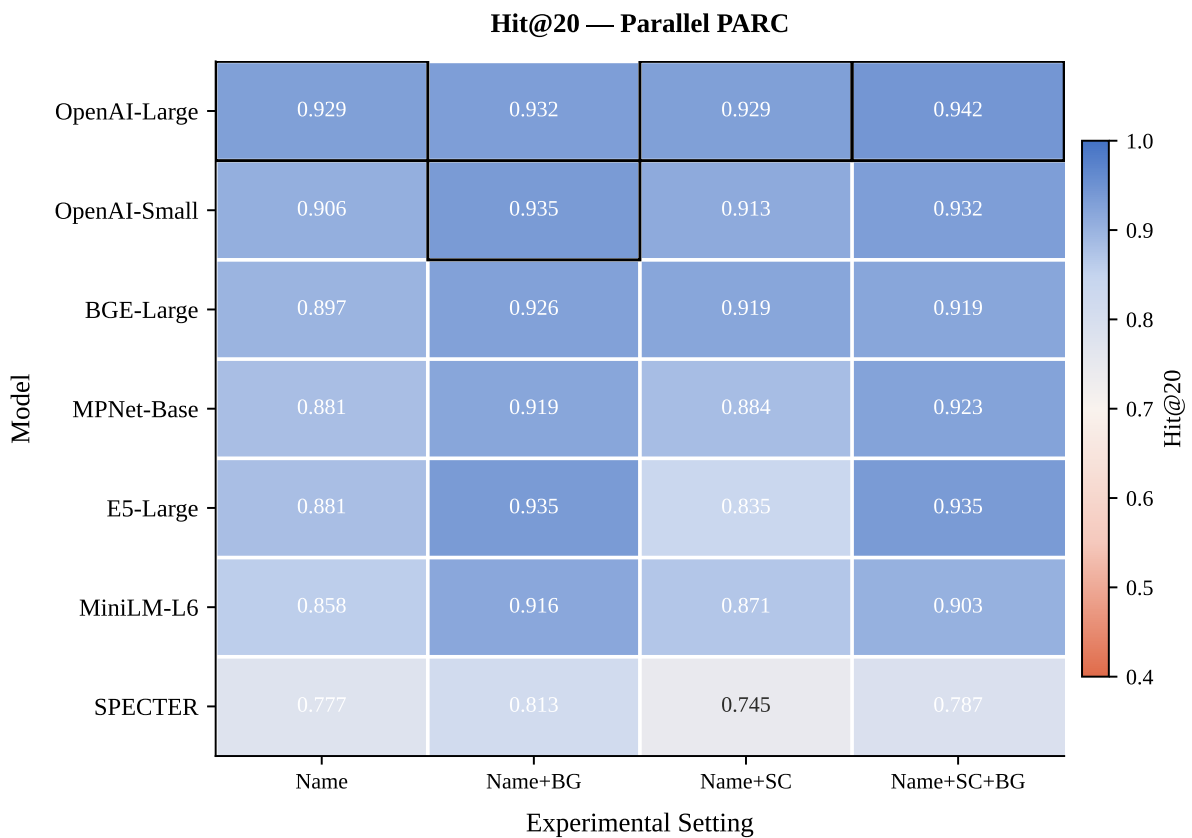


Figure 11: Hit@20 performance on the ParallelPARC dataset. Retrieval performance is generally higher than SCAR, reflecting the more structured nature of process-based analogies. Combining sub-concepts with background yields the best results across most models, indicating the value of richer contextual representations.

Table 4: Annotation dataset: 15 target concepts across five domains, each paired with three source analogies. T = Target-only, T+S = Target + Sub-concepts.

#	Target	Domain	Source Analogy	Model	Mode
1	Gas Diffusion	Chemistry	Perfume spreading Smoke dispersion Gasoline	GPT-OSS-120B Qwen3-14B Qwen3-14B	T T+S T
2	Encryption	CS	locked box Sealed envelope Locker	GPT-OSS-120B Grok-4-fast GPT-4.1-nano	T T+S T+S
3	Industrial Revolution	History	Steam engine era Automation Industrialization	GPT-OSS-120B LLaMA-8B LLaMA-405B	T T+S T
4	GAN Algorithm	CS	Art forgery Artist and Critic Teacher	Grok-4-fast Grok-4-fast GPT-4.1-nano	T T+S T
5	Bill of Rights	Law	Rulebook New-car warranty School curfew	Qwen3-32B GPT-OSS-20B GPT-OSS-20B	T T T
6	AdaBoost Algorithm	CS	Teamwork iterative training Gears in a machine	Gemini-2.5 GPT-OSS-120B LLaMA-70B	T+S T T+S
7	Platelets	Biology	Tiny repair crew Band-Aid Beads	Gemini-2.5 Grok-4-fast GPT-4.1-mini	T T+S T
8	Cell	Biology	factory House Garage	Grok-4-fast LLaMA-405B LLaMA-8B	T T T+S
9	Economics	Economics	Marketplace Household Budget Ecosystem	Qwen3-14B LLaMA-70B Qwen3-14B	T T T+S
10	Chemical Reaction	Chemistry	Baking soda + vinegar Digestion Boiling water	Qwen3-14B Qwen3-32B GPT-4.1-mini	T T+S T
11	Photosynthesis	Biology	Solar panels Greenhouse farming Fermentation Process	LLaMA-405B Grok-4-fast LLaMA-70B	T+S T T+S
12	Random forest	CS	Voting committee Jury deliberation Tree	Gemini-2.5 GPT-OSS-120B GPT-4.1-nano	T+S T+S T
13	Isotope Dating	Chemistry	Hourglass Atomic Clock Half-Life	DeepSeek-R1 LLaMA-405B LLaMA-70B	T+S T T
14	Renaissance	History	spring bloom Ancient Rome Enlightenment	GPT-OSS-120B retrieval-based LLaMA-70B	T Embedding T+S
15	EM Algorithm	CS	Detective investigation guess-and-check Miner	GPT-OSS-120B GPT-OSS-120B retrieval-based	T T+S Embedding

# Welcome to the Analogy Quality Annotation Study

Your expert judgment helps us validate how well automated systems evaluate analogies for education.

**⚠ Important:** Please complete the form in one sitting, as refreshing the page will reset your progress.

## BACKGROUND

We are building a system that generates **analogies** to help learners understand complex concepts through familiar comparisons. Large Language Models (LLMs) serve as automated judges to rate analogy quality — but we need **human evaluation** to verify their reliability.

## YOUR TASK

You will see **15 target concepts**, each paired with **3 candidate source analogies**.

### All 15 Targets

Rate each analogy on 3 quality dimensions (1–3 scale), rank the 3 analogies by learning usefulness, and indicate your confidence in the ranking.

Estimated time: **60–90 minutes**. A floating reference guide will be available throughout.

← Back

Continue →

Figure 12: (a) Welcome screen introducing the task and study background.

## Key Definitions & Rating Scale

Read these carefully — they define exactly what you're evaluating.

### WHAT IS AN ANALOGY?

An **analogy** maps a *target concept* (the unfamiliar topic being learned) onto a *source concept* (something more familiar). A good analogy preserves structural and functional relationships, helping learners transfer understanding from the known to the unknown.

Example: "A *cell is like a factory*" — Cell is the target, Factory is the source.

### THE THREE DIMENSIONS

#### 1. COHERENCE

**Does the pairing make intuitive sense?** When you hear "Target is like Source," does the connection feel natural and immediately understandable?

#### 2. MAPPING

**Do the key parts of the target correspond to parts of the source?** Read both descriptions carefully and try to identify counterparts between important components, processes, or relationships of the target and the source.

*How to identify mappings:* Go through the target description and list its key elements (components, roles, processes, relationships, inputs, outputs). Then, for each element, ask: *is there something in the source description that plays a similar role or serves a similar function?* The more elements you can pair up, the higher the mapping score. Elements left without a counterpart count against it.

*Tip — look for shared roles* (e.g., something that acts as a barrier in both), *shared processes* (e.g., both involve something flowing or transforming), and *shared relationships* (e.g., one part controlling another in both).

*Note on cross-type analogies:* Some analogies compare a process to an object or device (e.g., comparing a biological process to a machine). This is perfectly fine — what matters is whether the **functional roles and**

Figure 13: (b) Key definitions and rating scale for the three evaluation dimensions.

*Note on cross-type analogies:* Some analogies compare a process to an object or device (e.g., comparing a biological process to a machine). This is perfectly fine — what matters is whether the **functional roles and relationships** align, not whether both are the same kind of entity. A device can have parts that map onto the stages or components of a process and vice versa.

### 3. EXPLANATORY POWER

**Would this analogy actually help someone learn the target concept?** Does it make the abstract concrete, reveal causal relationships, or enable the learner to reason about how the target works?

A key factor is the **complexity gap** between source and target. The most useful analogies explain complex or unfamiliar concepts through simpler, more concrete, more everyday sources — moving the learner from the known to the unknown. An analogy where the source is just as complex or technical as the target may have good structural correspondence, but limited teaching value because the learner has no easier foothold to start from. For example, explaining one scientific process through another equally specialized process offers less educational benefit than explaining it through something a non-expert already understands intuitively.

*When scoring Explanatory Power, consider:* is the source meaningfully simpler or more familiar than the target? A "High" score typically requires the source to give the learner an easier entry point — not just a structurally parallel one.

#### RATING SCALE

SCORE	LEVEL	GUIDELINE
3	High	Strong and effective. Little to no weakness on this dimension.
2	Medium	Partial or acceptable. Some elements work but with notable gaps.
1	Low	Weak or broken. The analogy fails on this dimension.

Figure 14: (c) Explanatory power dimension definition and full rating scale.

### RANKING CONFIDENCE

After ranking the three source analogies for each target, you will indicate your **confidence in the ranking**. For some triplets, the quality differences are stark and the order is obvious; for others, the analogies are very close in quality and the ranking feels like a coin flip.

LEVEL	MEANING
<b>High</b>	The quality differences are clear — this ranking order is obvious.
<b>Medium</b>	You have a reasonable sense of the order, but some positions were close calls.
<b>Low</b>	The analogies are very similar in quality — the ranking could easily go another way.

### WORKED EXAMPLES (NOT PART OF YOUR ANNOTATION)

Three examples showing how to rate analogies. Click each tab to study a different target. Note: in these examples we describe the reasoning behind each score — during your annotation, you should apply similar reasoning by reading the source and target descriptions and identifying parallels yourself.

Atom      DNA      Electricity

**Target: Atom** (Physics) — The fundamental unit of matter, consisting of a dense central nucleus (protons and neutrons) surrounded by electrons in orbital shells, held together by electromagnetic forces.

**"Solar System" → Coh 3 · Map 3 · Exp 3**

**Coh 3:** The image of a central mass with orbiting bodies clicks immediately with the idea of a nucleus surrounded by electrons.

**Map 3:** Nearly every key component of the atom has a natural counterpart in the solar system — the central body, the orbiting objects, the governing force, and the orbital structure all find clear parallels.

**Exp 3:** A learner can use this analogy to reason about spatial layout, the role of forces, and relative scale.

**"Marble in a Stadium" → Coh 2 · Map 2 · Exp 2**

**Coh 2:** Conveys emptiness and scale contrast, but the full atomic structure isn't immediately apparent.

**Map 2:** The small object inside a vast space maps onto one aspect (the nucleus in mostly empty space), but several key parts of the atom — the orbiting particles, shells, and forces — lack counterparts.

Figure 15: (d) Ranking confidence scale and worked examples panel.

Atom
DNA
Electricity

**Target: Atom** (Physics) — The fundamental unit of matter, consisting of a dense central nucleus (protons and neutrons) surrounded by electrons in orbital shells, held together by electromagnetic forces.

**"Solar System" → Coh 3 · Map 3 · Exp 3**

**Coh 3:** The image of a central mass with orbiting bodies clicks immediately with the idea of a nucleus surrounded by electrons.

**Map 3:** Nearly every key component of the atom has a natural counterpart in the solar system — the central body, the orbiting objects, the governing force, and the orbital structure all find clear parallels.

**Exp 3:** A learner can use this analogy to reason about spatial layout, the role of forces, and relative scale.

**"Marble in a Stadium" → Coh 2 · Map 2 · Exp 2**

**Coh 2:** Conveys emptiness and scale contrast, but the full atomic structure isn't immediately apparent.

**Map 2:** The small object inside a vast space maps onto one aspect (the nucleus in mostly empty space), but several key parts of the atom — the orbiting particles, shells, and forces — lack counterparts.

**Exp 2:** Teaches one property (emptiness) well but misses the dynamic multi-part relationship.

**"Lego Brick" → Coh 1 · Map 1 · Exp 1**

**Coh 1:** A solid uniform block misrepresents the internal structure of an atom entirely.

**Map 1:** No components of the source correspond to the key parts of an atom.

**Exp 1:** Could actively mislead learners. Provides zero insight into atomic behavior.

**Ranking:** 1st Solar System · 2nd Marble in a Stadium · 3rd Lego Brick

← Back
Continue →

Figure 16: (e) Worked example for the *Atom* target with scored candidate sources.

# You're All Set

Here's your anonymous session identifier and a few final notes.

## YOUR ANONYMOUS ID

This auto-generated code identifies your session. It is not linked to any personal information — it simply lets us distinguish different annotators' results.

**ANN - HWQCC4**

**⚠ Reminder:** Please complete the form in one sitting, as refreshing the page will reset your progress.

**Tip:** The [Reference Guide](#) button in the top-right corner will be available on every annotation page. Use it anytime to revisit the dimension definitions, rating scale, or worked examples.

## REMINDERS

- For each target, you'll see the concept name, domain, and a description explaining what it is.
- Each source analogy also includes a description explaining what it is.
- Rate each source on 3 dimensions, rank the sources, then indicate your confidence in the ranking.
- You **must drag** the ranking to confirm your preferred order — even if you agree with the default arrangement.
- Source order is randomized. There are no trick questions. Take your time.

[← Back](#)

[Continue →](#)

Figure 17: (f) Annotator onboarding screen with anonymous session ID and task reminders.

RATE & RANK · TARGET 1 OF 15

CHEMISTRY

## Gas Diffusion

The process by which gas molecules spread from regions of high concentration to low concentration through random molecular motion (Brownian motion) in a medium such as air, until equilibrium is reached.

SOURCE ANALOGY 1

**"Gas Diffusion is like *smoke dispersion*"**

Smoke released from a fire or cigarette starts as a dense, visible plume and gradually thins out as the particles spread through the surrounding air. The movement is influenced by air currents, convection, and gravity — larger particles may settle downward while finer ones drift more freely.

<b>Coherence</b> Does the pairing click?	1 · Low	2 · Med	3 · High
<b>Mapping</b> Do key parts correspond?	1 · Low	2 · Med	3 · High
<b>Explanatory Power</b> Helps someone learn? Is the source simpler than the target?	1 · Low	2 · Med	3 · High

Figure 18: (g) Rating interface for *Gas Diffusion* — Source Analogy 1.

**SOURCE ANALOGY 2**

### "Gas Diffusion is like *gasoline*"

Gasoline is a volatile liquid fuel refined from petroleum, composed of a mixture of hydrocarbons. It is used primarily in internal combustion engines, where it vaporizes and mixes with air before being ignited by a spark plug to produce mechanical energy.

<b>Coherence</b> Does the pairing click?	1 · Low	2 · Med	3 · High
<b>Mapping</b> Do key parts correspond?	1 · Low	2 · Med	3 · High
<b>Explanatory Power</b> Helps someone learn? Is the source simpler than the target?	1 · Low	2 · Med	3 · High

Figure 19: (h) Rating interface for *Gas Diffusion* — Source Analogy 2.

**SOURCE ANALOGY 3**

## "Gas Diffusion is like *perfume spreading*"

When perfume is sprayed, scent molecules leave the nozzle as a concentrated burst and then gradually disperse in all directions through the surrounding air. Over time the scent reaches every corner of a room, becoming fainter near the source as molecules spread outward through random collisions with air particles.

<b>Coherence</b> Does the pairing click?	<input type="radio"/> 1·Low	<input type="radio"/> 2·Med	<input type="radio"/> 3·High
<b>Mapping</b> Do key parts correspond?	<input type="radio"/> 1·Low	<input type="radio"/> 2·Med	<input type="radio"/> 3·High
<b>Explanatory Power</b> Helps someone learn? Is the source simpler than the target?	<input type="radio"/> 1·Low	<input type="radio"/> 2·Med	<input type="radio"/> 3·High

Figure 20: (i) Rating interface for *Gas Diffusion* — Source Analogy 3.

**⚠ Drag to confirm**

### Which analogy is most useful for learning?

Drag to reorder. #1 = most useful for a learner encountering this concept.

- 1 smoke dispersion
- 2 gasoline
- 3 perfume spreading

---

#### How confident are you in this ranking?

Consider how distinguishable the analogies are from each other in overall quality.

<b>Low</b> Very close in quality — ranking could easily go another way	<b>Medium</b> Reasonable sense of the order, but some positions were close	<b>High</b> Quality differences are clear — this order is obvious
---	---	--

← Back Continue →

Figure 21: (j) Drag-to-rank interface for ordering sources by learning usefulness with confidence selection.

## K Human Evaluation Rank Agreement

Target	Analogy 1	Analogy 2	Analogy 3	Kendall's W	p-value
adaboost algorithm	gears in a machine	iterative training	teamwork	0.735	0.0058
bill of rights	new-car warranty	rulebook	school curfew	0.388	0.0663
cell	factory	garage	house	0.551	0.0211
chemical reaction	baking soda + vinegar	boiling water	digestion	0.878	0.0021
economics	ecosystem	household budget	marketplace	0.020	0.8669
em algorithm	detective investigation	guess-and-check	miner	0.755	0.0051
encryption	locked box	locker	sealed envelope	0.143	0.3679
gan algorithm	art forgery	artist and critic	teacher	0.878	0.0021
gas diffusion	gasoline	perfume spreading	smoke dispersion	0.796	0.0038
industrial revolution	automation	industrialization	steam engine era	0.061	0.6514
isotope dating	atomic clock	half-life	hourglass	0.633	0.0119
photosynthesis	fermentation process	greenhouse farming	solar panels	0.388	0.0663
platelets	band-aid	beads	tiny repair crew	1.000	0.0009
random forest	jury deliberation	tree	voting committee	1.000	0.0009
renaissance	ancient rome	enlightenment	spring bloom	0.061	0.6514

Table 5: Inter-annotator agreement for analogy ranking using Kendall's  $W$ . Each row corresponds to a target concept with three candidate analogies. Higher  $W$  indicates stronger agreement among annotators, while the p-value tests statistical significance.

Table 5 reports inter-annotator agreement using Kendall's coefficient of concordance ( $W$ ) for each target concept. Annotators ranked three candidate analogies per target. Higher values of  $W$  indicate stronger agreement.

## L Full Human and LLM-as-a-judge Results Heatmap

Figures 22, 23 and 24 show the full human and Claude sonnet 4.6 agreement.

## M Word Overlap Analysis in Embedding-based Retrieval

Figure 25 shows the impact of lexical overlap on embedding-based source retrieval.

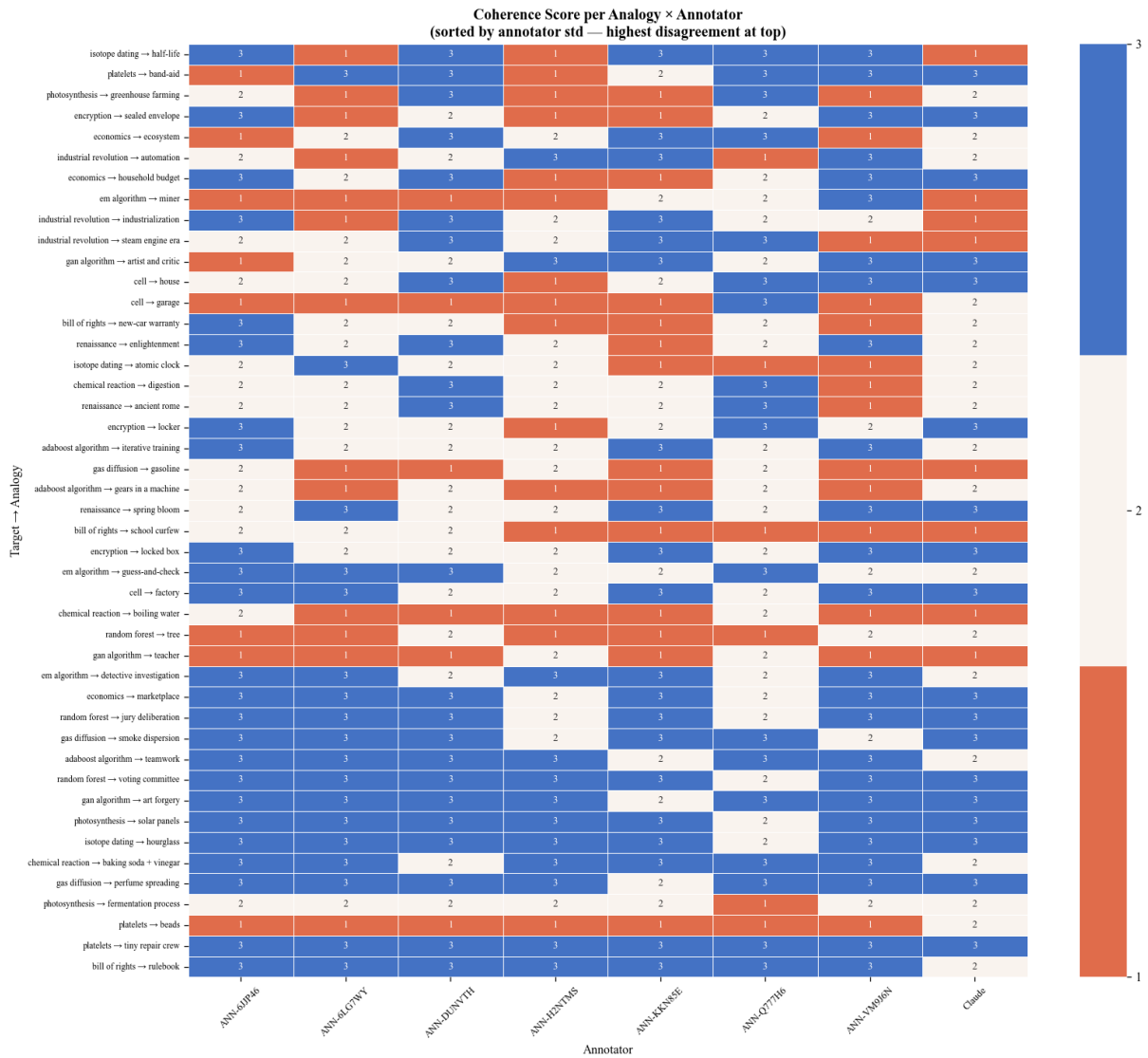


Figure 22: Mean coherence scores per analogy across eight annotators (seven human annotators and Claude), sorted by inter-annotator standard deviation (highest disagreement at top). Scores are on a 1–3 scale. High-disagreement analogies such as *isotope dating* → *half-life* and *platelets* → *band-aid* reveal cases where structural clarity is perceived differently across annotators. Low-disagreement analogies at the bottom (e.g., *platelets* → *tiny repair crew*, *bill of rights* → *rulebook*) receive consistently high scores, indicating clear and intuitive mappings. Claude’s ratings (rightmost column) generally align with the human majority, with occasional deviations in mid-range cases.

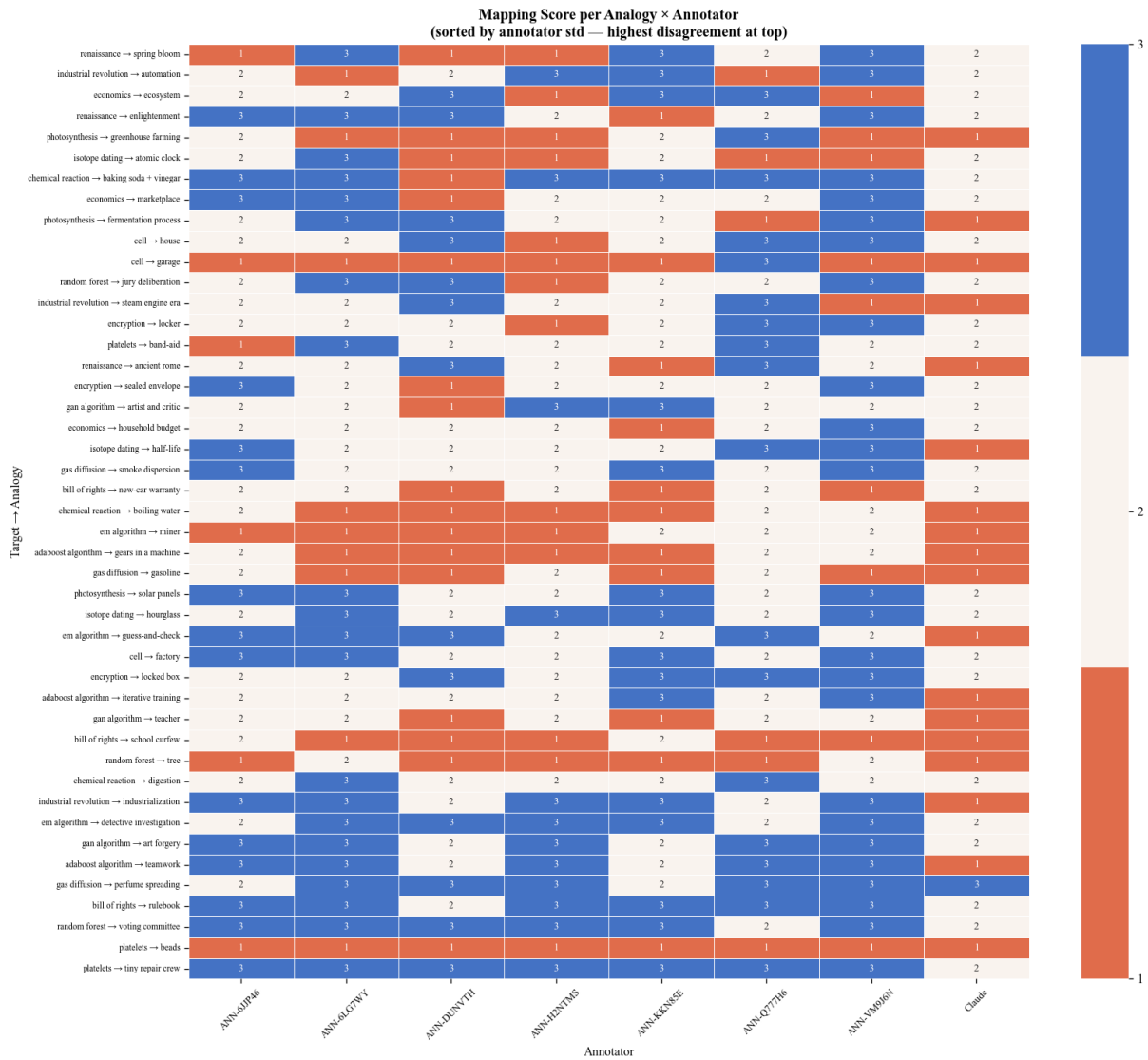


Figure 23: Mean mapping quality scores per analogy across eight annotators, sorted by inter-annotator standard deviation. Mapping quality—capturing how well source sub-concepts correspond to target sub-concepts—shows greater disagreement than coherence, particularly for analogies such as *renaissance -> spring bloom* and *industrial revolution -> automation*. Low-disagreement cases (e.g., *platelets -> beads*, *platelets -> tiny repair crew*) tend to receive uniformly low scores, suggesting that weak mappings are easier to agree on than strong ones. Claude’s scores broadly follow human judgments, with a slight tendency toward mid-range values in ambiguous cases.

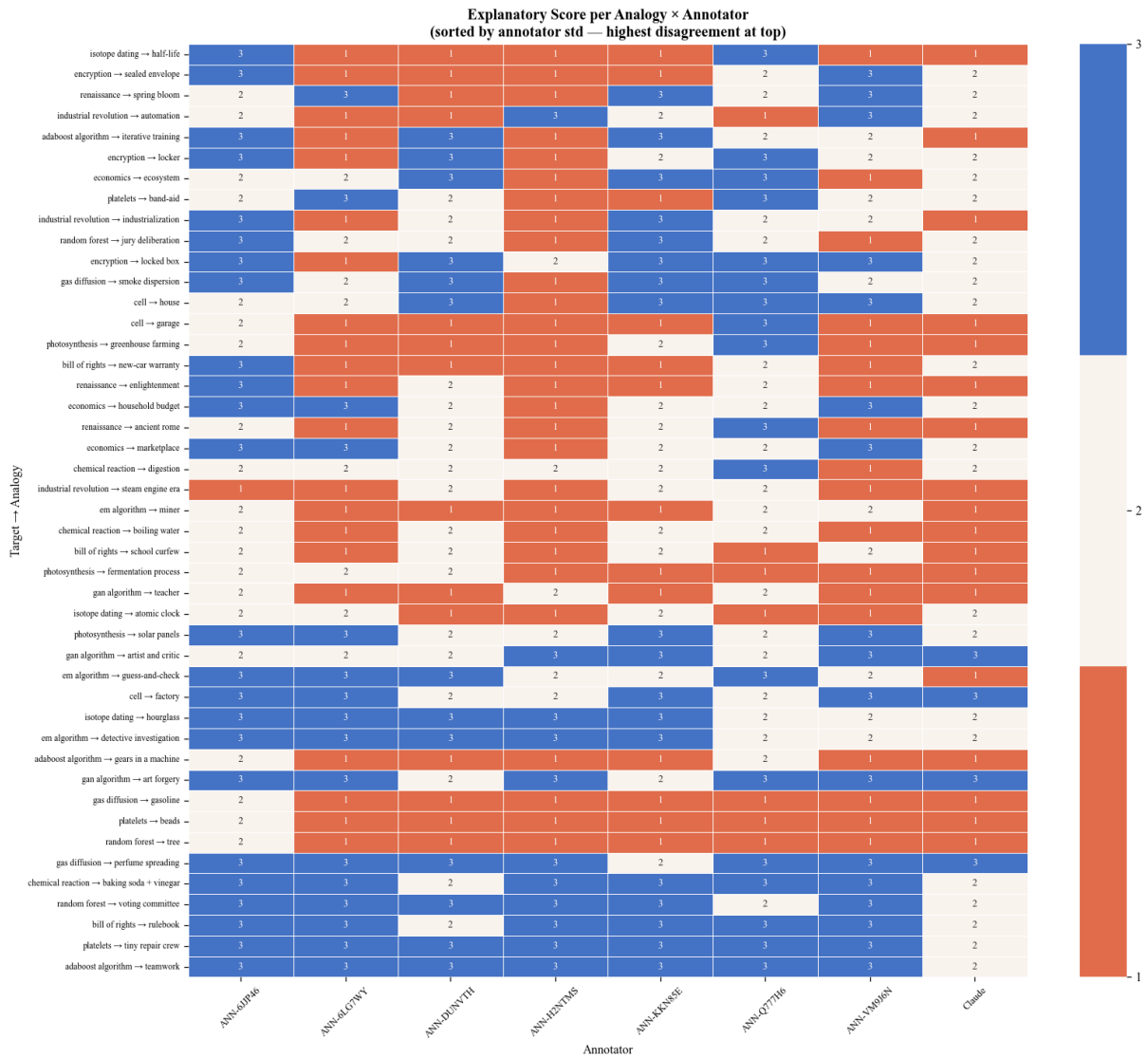


Figure 24: Mean explanatory power scores per analogy across eight annotators, sorted by inter-annotator standard deviation. Explanatory power—measuring how well an analogy supports understanding—exhibits the highest disagreement, especially for analogies such as *isotope dating* → *half-life* and *encryption* → *sealed envelope*. Consistently high-scoring analogies (e.g., *platelets* → *tiny repair crew*, *adaboost algorithm* → *teamwork*) indicate strong pedagogical value. Analogies involving less familiar concepts (e.g., *gas diffusion* → *gasoline*, *random forest* → *tree*) cluster in lower-score regions, suggesting more limited explanatory effectiveness.

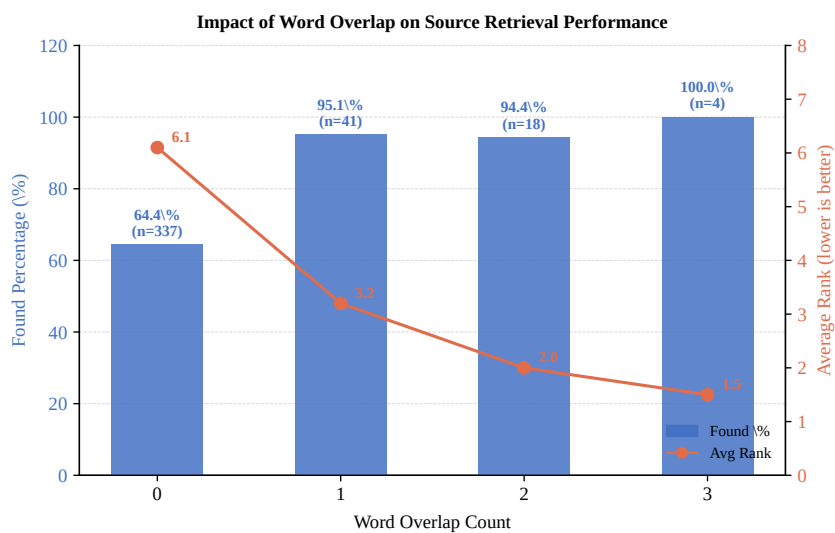


Figure 25: **Impact of lexical overlap on embedding-based source retrieval.** This figure shows how the number of overlapping words between target and source names affects retrieval performance, measured by the percentage of times the gold source is retrieved and its average rank (lower is better). As lexical overlap increases, both retrieval success and ranking quality improve, indicating that embedding-based retrieval is strongly influenced by surface-level word overlap.