ParallelPARC: A Scalable Pipeline for Generating Natural-Language Analogies

Oren Sultan, Yonatan Bitton, Ron Yosef, Dafna Shahaf
The Hebrew University of Jerusalem
{oren.sultan, yonatan.bitton, ron.yosef, dshahaf}@cs.huji.ac.il

Abstract

Analogy-making is central to human cognition, allowing us to adapt to novel situations – an ability that current AI systems still lack. Most analogy datasets today focus on simple analogies (e.g., word analogies); datasets including complex types of analogies are typically manually curated and very small. We believe that this holds back progress in computational analogy.

In this work, we design a data generation *pipeline*, *ParallelPARC (Parallel Paragraph Creator)* leveraging state-of-the-art Large Language Models (LLMs) to create complex, paragraph-based analogies, as well as distractors, both simple and challenging. We demonstrate our pipeline and create *ProPara-Logy*, a dataset of analogies between *scientific processes*. We publish a *gold-set*, validated by humans, and a *silver-set*, generated automatically. We test LLMs' and humans' analogy recognition in *binary* and *multiple-choice* settings, and found that humans outperform the best models (∼13% gap) after a light supervision. We demonstrate that our *silver-set* is useful for training models. Lastly, we show *challenging distractors* confuse LLMs, but not humans. We hope our pipeline will encourage research in this emerging field.

1 Introduction

Analogy-making is a central to human cognition. It allows us to abstract information and understand novel situations in terms of familiar ones (Minsky, 1988; Hofstadter and Sander, 2013; Holyoak, 1984), abilities that are still lacking in current AI systems. Research suggests that these abilities are essential for robust AI that can effectively generalize and adapt to diverse domains (Mitchell, 2021).

According to Gentner's Structure Mapping Theory (SMT) (Gentner, 1983), analogy is a *mapping* from entities in base $\mathcal{B}$ to entities in target $\mathcal{T}$, relying on *relational similarity*, not *object attributes*.

For example, in the analogy between an electrical circuit and a water pump, there is a mapping between *electrons → water*, *wire → pipe*. While object attributes are different (water is liquid, electrons are not), the *relations* are similar (electrons move through wires like water flows in pipes).

Despite the importance of analogy, relatively few analogy resources exist today. Most resources mainly focus on *word-analogies* ("A:B is like C:D"). We argue that this setting is too simplistic, often boiling down to a single relation ("PartOf", conjugation); in the real world, analogies are often complex, involving multiple entities and intricate relations between them. Real-world analogies are often described in natural language, adding to the complexity of the problem. A very recent work employed LLMs to generate analogies at scale between 2-sentence snippets (∼20 tokens) (Jiayang et al., 2023). However, resources of more complex analogies (e.g., full paragraphs) are few and *sparse* (18 samples max). We believe this lack of data hinders progress in computational analogy; in the past, high-quality datasets have led to a burst of novel research (e.g., ImageNet (Deng et al., 2009)).

In this work, we design a pipeline, *ParallelPARC (Parallel Paragraph Creator)* to scale up the process of generating analogies between paragraphs (see Figure 1), leveraging recent progress in LLMs. We release a *gold-set*, validated by humans, and a *silver-set*, which is automatically generated.

Coming up with non-trivial negative examples (non-analogous paragraphs) is a challenging task. Our pipeline generates, in addition to positives (analogies), both *simple negatives* (random paragraphs) and *challenging negatives* (distractors).

To demonstrate our pipeline, we create *ProPara-Logy*, a dataset of paragraphs describing *scientific processes* across various domains, meant for studying analogical reasoning. A sample in our data includes two processes, each described via a title ("How does a solar panel work?"), a domain ("En-
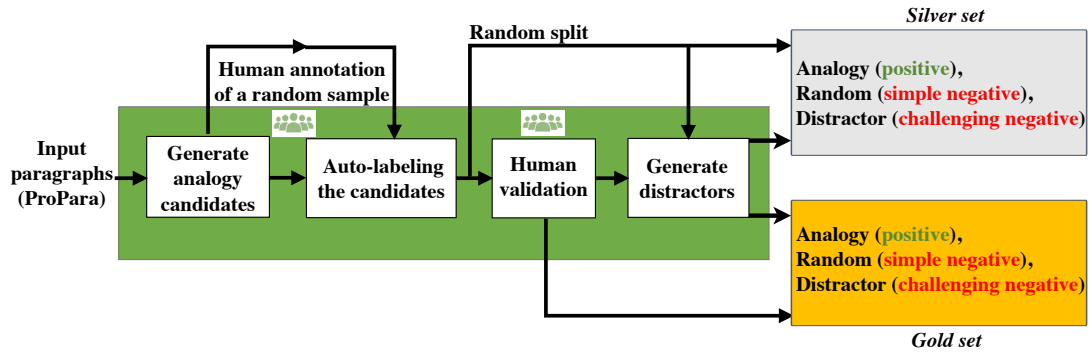
Figure 1: Our data generation pipeline. We generate analogy candidates, then collect human annotations on a random sample to be used as few-shot for an auto-labeling model. We run the model to label candidates at scale. We randomly split the data into *silver-set* and *gold-set*, which is validated by humans. In addition to *positives* (analogies), we include random target paragraphs (simple negatives), and generate distractors (challenging negatives).

gineering"), and a full paragraph. In addition, the data includes *similar relations* between the two processes, which is a core part in understanding why they could be analogous (See Figure 2).

We evaluate LLMs and humans on *binary* and *multiple-choice* analogical reasoning tasks on *ProPara-Logy*. We found that humans outperform the best models (∼13% gap) after a light supervision. We show the automatically-generated *silver-set* is useful for training models, and can significantly improve their performance. Finally, we demonstrate the distractors significantly reduce the performance of LLMs, but not of humans.

Our main contributions are:

- We develop a novel data pipeline to create complex, paragraph-based analogies.
- We demonstrate our pipeline and create the *ProPara-Logy* benchmark, a dataset for analogical reasoning over paragraphs describing processes in science. Our dataset is orders of magnitude larger than previous work, and could easily be expanded.
- Beyond the analogous paragraphs (positives), our dataset includes both simple and challenging distractors (negatives). It also includes useful information about the analogies, such as relations shared between the paragraphs.
- We use *ProPara-Logy* to evaluate humans and LLMs on our proposed analogical reasoning tasks, both in zero-shot and guided settings.
- We release data and code at `https://github.com/orensul/ParallelPARC`.

2 Existing Analogy Datasets

We now survey available analogy resources.
Word analogies. Many analogy resources focus on

word analogies ("A:B is like C:D") (Jurgens et al., 2012; Popov et al., 2017; Kmiecik et al., 2019; Rogers et al., 2016; Czinczoll et al., 2022). Such analogies are widely used in entrance tests like the SAT in the US or NCEE in China.

This area has gained popularity in the NLP community after Mikolov et al. (2013) show that word embeddings can model some relational similarities in terms of word vector offsets. This method can find analogies relying on certain simple types of relations, but struggles with complex relations (Linzen, 2016; Schluter, 2018; Ushio et al., 2021). More recently, several studies explored the use of LLMs in generating word analogies (Bhavya et al., 2022; Yuan et al., 2023a,b).

In addition to the word analogy itself (A, B, C and D), some resources include extra information, such as explanations (Chen et al., 2022). Other resources include multiple correct options, either close analogies (C, D are similar to A, B) or far (C, D are from a different domain than A, B) (Green et al., 2010). Some resources include wrong answers, but often quite simple (e.g., random words).

Visual analogies. This is the visual equivalent of word analogies (where A, B, C, D are images). There have been multiple attempts to represent *transformations* between pairs of images (Reed et al., 2015; Radford et al., 2016; Tewel et al., 2021), typically stylistic or geometric, and several resources published (Sadeghi et al., 2015; Bitton et al., 2022). As in word analogies, generating wrong answers is challenging. They are often created by either using random images or images that contain elements of the correct answer but exclude another element that is crucial for the analogy.

A different kind of resource is ARC (Chollet,

Base	Target	Similar Relations
Title: How does a solar panel work? **Domain**: Engineering **Paragraph**: **solar energy** *powers* an **electric current** within a solar panel. The **photovoltaic cells** within the panel *convert* the **energy** from the sun into electricity. The **electrical wires** then *spread* this **power** throughout the panel. The **electric current** is then used to *power* whatever the panel is connected to.	**Title**: How does photosynthesis occur? **Domain**: Natural Science **Paragraph**: Photosynthesis occurs when **sunlight** *powers* **chemical reactions** within the **chloroplasts** of a plant. The **chloroplasts** are able to *transform* the energy from the **sunlight** into usable energy for the **plant**. This energy is then used to produce **nutrients** for the plant, which are then *distributed* throughout the **plant**.	(**solar energy**, *powers*, **electric current**) (**sunlight**, *powers*, **chemical reactions**) (**photovoltaic cells**, *convert*, **energy**) (**chloroplasts**, *tranform*, **energy**) (**electrical wires**, *spread*, **power**) (**plants**, *distribute*, **nutrients**)

Figure 2: An example of an *analogous* sample from our dataset (generated by our pipeline). Two scientific processes, base and target, are described via a title, a domain, and a paragraph of natural-language text. A sample also includes *similar relations*, hinting at why the processes are analogous.

2019), where test-takers have to discern rules from *pixel grids* to deduce the correct output grid.

Paragraph-level analogies. Very recently, Jiayang et al. (2023) created a dataset of 24K story pairs. However, the pairs are short snippets (2 sentences, ~20 tokens), and well-aligned, making the setting overly simplistic. Moreover, their work does not assess directly whether a pair is analogous. There are few resources of analogies between *full paragraphs*, most notably stories from cognitive-psychology literature (Gentner et al., 1993; Wharton et al., 1994; Clement and Gentner, 1991). These datasets are manually curated and very small (18 samples max), rendering them inadequate for training models. Furthermore, the stories have a near-identical structure ("Mr. Newton was the manager of a company that made razors"/"Mr. Boyce was director of manufacturing shaving knives..."), again making the setting non-realistic.

Notably, the dataset of Gentner et al. (1993) also includes false analogy stories, which are similar to the base paragraph in terms of first-order relations, but dissimilar in higher-order relations (relations between the first-order relations). Jiayang et al. (2023) includes simple (random) negatives and hard negatives (snippets with similar entities).

A recent work focused on finding analogies between paragraphs describing processes (Sultan and Shahaf, 2022). Their method ranks pairs of paragraphs from a dataset, such that analogous pairs rank high. However, this is a noisy resource, as many non-analogies rank high, and many of the identified analogies are from very close topics.

3 Dataset Generation

Our goal is to develop a *pipeline* for generating high-quality data that could drive forward research

efforts in computational analogy. Figure 2 illustrates the format of data generated by our pipeline. A records contains two processes, base $\mathcal{B}$ and target $\mathcal{T}$. Each process is described via a title ("How does a solar panel work?"), a domain ("Engineering"), and a full paragraph.

In addition to expressive natural-language paragraphs, the data also includes *similar relations* between the two processes, which is a core element in identifying analogies (Figure 2, right).

Figure 2 shows a *positive example* (analogy). In addition to positives, our pipeline generates *simple negatives* and *challenging distractors*, designed to fool both humans and models.

The pipeline (see Figure 1) begins by using LLM for generating **analogy candidates** – paragraphs (and relations) that potentially describe analogous processes across diverse domains in science (§3.1). Then, we use **human annotators** to label a random sample of the candidates (§3.2), and use the annotated data as a few-shot for **automatic labeling** of candidates (§3.3). Then, we filter the data based on the automatic labels, and randomly split the filtered data into two disjoint sets: our *gold-set*, further **validated by humans**, and our *silver-set*, which is not (§3.4). Finally, we employ an LLM to generate challenging **distractors** (§3.5).

3.1 Analogy Candidates Generation

Our goal in this section is to generate analogy candidates from diverse scientific domains.

We employed GPT-3.5 (text-davinci-003)[1] (Brown et al., 2020) (see implementation details in Appendix A.2). We first naïvely tried to ask GPT

---

[1]We have chosen GPT-3.5 after experimenting with several newer models, and finding that it delivers high-quality results at a very reasonable cost.

repetitively for two analogous scientific processes (with no additional constraints or guidance). We found that GPT (1) tends to repeat itself, and (2) often creates analogies revolving around extremely similar topics.

To solve the problem of *repetitiveness*, we seeded GPT with $\mathcal{B}$ instead of asking for generating both $\mathcal{B}$ and $\mathcal{T}$. We used the ProPara dataset (Dalvi et al., 2018) of English paragraphs describing scientific processes, taking 390 titles from its training set. To solve the problem of *similar topics*, we tried to explicitly diversify the target paragraphs by asking for analogies in specific fields (e.g., zoology), but often no analogies were found. Ultimately, we selected several *broad* domains: *Engineering*, *Natural Science*, *Social Science* and *Biomedical and Health Science*. This provided a balance between diversity and specificity, and also allowed us to control the distribution of target domains.

We first tried using a single prompt for generating analogies. However, that led to paragraphs that were mostly identical to the input paragraph except for nouns ("The sediment is deposited again in a new place"/"Money is deposited again in a new place"), and artificially sounding sentences ("Money travels through the economy").

As noted earlier, analogy is often defined as a system of similar relations (Gentner, 1983). Thus, we decided to use relations as a stepping stone towards generating analogies; we developed two *separate prompts*, one for finding an analogous subject and identify similar relations, and another for taking the subject and relations and turning them into paragraphs in natural language (see Appendix A.2 Figures 4, 5). This approach has proven to be effective in practice. We experimented with one-shot and few-shot settings, and chose the one-shot prompt, which was more cost-effective.

We include relations in our data in addition to the paragraphs, subjects, and domains (Figure 2, right). We believe they are can also serve as potential *explanations*, highlighting the structural similarity between base and target paragraphs.

For each paragraph in ProPara, we generate 3 analogy candidates in 4 broad domains, resulting in 4680 samples. We filter out samples with less than 3 similar relations (less likely to be analogies), leaving us with 4288 candidates.

3.2 Human Annotation Task

In the previous section we generated analogy candidates. We now annotate a small portion of this data. Our goal is two-fold: (1) to estimate the proportion of analogies in the data, as well as identify issues with the generation process, and (2) to use the annotated data to train models.

We hired Amazon Mechanical Turk (AMT) workers who passed a rigorous qualification task. Workers received two paragraphs, base $\mathcal{B}$ and target $\mathcal{T}$, corresponding subjects, domains, and the similar relations generated by the LLM. The task is to determine whether the paragraphs are analogous *and* the similar relations are correct. If they are, the worker needs to select between close analogy (close topic, similar entities) or far analogy (unrelated topics). If there is an issue with the analogy or the relations, the worker marks it "for further inspection", along with a reason: dissimilar relations, misinformation, cyclic vs. non-cyclic process, or other (with a free-text explanation).

Note that two processes may be deemed analogous or not depending on the annotator's abstraction, which is affected by their domain knowledge. To ameliorate this, we explicitly instructed annotators to focus on relational similarity, between relations *as they are expressed in the texts*, and not take domain knowledge into account.

Three workers labeled each sample, for a reward of $0.5 per sample. See Appendix A.4 for more details about the annotation process.

3.3 Automatic Filtering and Labeling

Based on the annotations in Section 3.2, we estimate analogies to be less than 30% of the dataset.

Next, we decided to use part of our annotated data as few-shot examples for our *filtering model*. The goal is two-fold: (1) As the annotation process is long and costly, it could identify the most probable analogous candidates to show our annotators. (2) If the model performance matches humans, we could replace the human-in-the-loop and achieve a **fully automated** pipeline.

This task is complex, and thus we use GPT-4 (OpenAI, 2023), a state-of-the-art LLM (parameters in Appendix A.3). We input randomly selected annotated candidates (30 examples, maximum allowed tokens) into GPT, comprising two paragraphs, their subjects, similar relations, and a label indicating how many workers labeled it as an analogy (0-3). See Appendix A.3 for the prompt.

Following the in-context learning phase, we run the model on our unlabeled analogy candidates.

Base:	Target (Analogy):	Target (Distractor):
How do bats use echolocation? (Natural Sciences) Bats use echolocation to navigate and find food. **They emit high frequency sound waves** that bounce off of objects in their environment. The bats then **receive the echoes** and **interpret the information** to locate their prey and navigate their surroundings. Submarines interpret the echo to determine the distance and size of the object.	**How do submarines use sonar? (Engineering)** Submarines use sonar technology to detect objects in the water. **They emit sound waves**, which travel through the water and bounce off the objects. **The sound waves are then received back as an echo**. **Submarines interpret the echo** to determine the distance and size of the object.	**How do submarines use sonar? (Engineering)** **Submarines interpret the echo** to determine the distance and size of the object. **After interpreting the echo, they emit sound waves**, which travel through the water and bounce off the objects. **These sound waves are then received back as an echo**. Finally, submarines use sonar technology to detect objects in the water.

Figure 3: An example of the distractor creation process. On the left is the **Base** paragraph (about bats using echolocation). In the middle, a **Target** paragraph, which is analogous to the base paragraph. On the right is a **Target (Distractor)** paragraph, generated from the middle paragraph by switching the order of events: The emission of sound waves, followed by their reception as an echo, and submarines interpret the received echo. In the **Target (Distractor)**, the order is reversed, altering the cause-and-effect relations from the true analogy.

3.4 Human Validation

Our goal in this paper is to demonstrate how our pipeline can be used for creating datasets. We consider two types of datasets: a *silver-set*, automatically labeled, and a *gold-set*, validated by humans.

Thus, we returned to the task from Section 3.2. We show annotators both the most likely analogous candidates, as predicted by the model, but also the least likely candidates. This allows us to evaluate the filtering model where it is most certain. It also balances the data for the annotators. In addition, it is surprisingly hard to come up with hard negative examples. We believe that the least likely candidates (according to the model) hold the potential to be useful in future research (see Section 4).

Identifying analogies is a complex task. Therefore, in addition to the thorough qualification phase (Section 3.2), we also consistently monitored and provided clarifications to the annotators. To further ensure quality for our *gold-set*, we chose a strict setting: a sample is positive only if *all three* annotators agree it is an analogy. For our proof-of-concept, we wanted a *gold-set* with at least 300 positives. We randomly gave annotators small batches to label until reaching 310 positives. Annotators labeled 828 instances (not including the 130 from Section 3.2), for a total cost of $1,804.

Our annotators' agreement is 78.6%, where random chance is 25% (% of perfect agreement).
Filtering model evaluation. We also use the annotated data to evaluate the filtering model. We compare its predictions to workers' majority vote. Our model achieves an accuracy of 85.1%, f1-score of 83.4%. Importantly, it reaches 79.5% preci-

sion when predicting high likelihood of an analogy, which is significantly higher than the 30% base rate, and 90% precision when predicting low likelihood. These results show our model reliably replicates annotators on the high-confidence samples, rendering our approach *scalable*. Consequently, we release a *silver-set*, generated by applying the filtering model on the remaining candidates. This data could also be useful for training models (Section 5).

3.5 Distractors Generation

In addition to the 310 analogies in our *gold-set*, we create *simple* negatives from random ProPara paragraphs on different subjects[2] as $\mathcal{T}$. However, those are quite easy to tell apart from analogies; thus, we now focus on creating *challenging* negatives.

While many types of distractors are possible, we are inspired by the ideas in Gentner et al. (1993). There, story pairs match or not match at three levels: attributes, first-order and higher-order relations. We focus on the most *complex and challenging* setting – stories matching only in first-order relations. We leave other dimensions for future work.

Formulation. Let $\mathcal{B}$ and $\mathcal{T}$ be two analogous paragraphs. The intuition is to create distractor $\mathcal{T}'$ that keeps first-order relations of $\mathcal{T}$ (Figure 2, right) but changes the higher-order relations – i.e., relations between first-order relations, such as cause and effect, or temporal dependencies between events. To create $\mathcal{T}'$, we find two *dependent events* in $\mathcal{T}$ such that one must precede the other, and switch their order. See Figure 3 for an example, generated by

---

[2] We estimate two ProPara paragraphs on different subjects are analogous in ~1%, based on Sultan and Shahaf (2022).

our method: the relations are the same, but the submarines interpret the echo *before* emitting sounds returning as echos. See Appendix A.5 for details.

Generation. We use GPT-4 to automatically generate distractors with two separate prompts: (1) finding and replacing two dependent events, and (2) writing a *coherent* $\mathcal{T}'$. For the first task, we use one-shot. We ask GPT-4 to output a list of the events in $\mathcal{T}$ according to their order in time, and then replace two dependent events, along with an explanation. For the second task, we use few-shot. The input is an order of events and the output is a *coherent* paragraph. See details in Appendix A.5.

Evaluation. We now evaluate the generated distractors. A correct distractor should switch two dependent events, with a paragraph that is coherent and consistent with the new order. We begin with a sanity check of 10 distractors, involving three members from our team. The members reach a full consensus on the 10 samples. After reaching calibration, two team members proceeded to label 100 more distractors (50 each). The annotators found that 10 paragraphs could not have been made into good distractors (as they contain no dependencies). Out of the rest, 89% of the generated distractors were correct. For the wrong ones, in 5 samples the generated paragraph was not *coherent*, and in 5 the choice of events to replace was wrong.

We deduce the distractor generation is effective, and create distractors for both gold and silver sets.

4 Dataset Analysis

Our *gold-set* contains 310 positives (analogies), each with one corresponding simple (random) distractor and one challenging distractor. Our *silver-set* contains 403 positives, again with corresponding distractors. We note this is a proof-of-concept, and it is possible to construct larger sets if desired.

Gold-set analysis. We first computed the distribution of close and far analogies (based on majority vote). When all three annotators voted positive, 40% were far analogies. When at least one voted positive, the number increased to 47%. We conclude that our dataset is relatively balanced between close/far analogies. Not surprisingly, disagreements are more common for far analogies.

Table 1 shows different issues raised with the candidates. The most common is "dissimilar relations", indicating that GPT-3.5 has difficulties generating the relations. We note it is also quite easy for annotators to detect. "Other" was chosen

in approximately a quarter of the cases. An example reason provided is *inconsistent mapping* of entities. See Appendix A.6 for more reasons.

A Note on Scalability. The *silver-set* is generated automatically at scale. Our major annotation effort was to create the *gold-set*. For future users of the pipeline, we recommend the automatic route, with short annotation rounds for quality assessment.

A Note on Additional Data Released. Through the different stages of the pipeline, we collect information about candidates that does not make it into our gold or silver sets. We believe that this information might be beneficial for further research in this area. For example, *differences* in judgments might be interesting. In addition, our human annotators give *structured* feedback (see Section 3.2). If the annotators identified an issue with the generated relations, for example, it could still be the case that the *paragraphs themselves* are analogous (which is the reason we do not use them as negatives). Thus, we decide to make this data available to the community. We believe it opens up interesting avenues, from creating new types of distractors to teaching models how to automatically fix flawed analogies.

5 Evaluating Humans and LLMs

We use the data to develop the ProPara-Logy *benchmark* of analogy recognition. We propose *binary classification* and *multiple-choice* tasks. We evaluate the performance of both humans and state-of-the-art models, experimenting in zero-shot and guided settings (using labeled examples). Our research questions are:

RQ1: How well can humans and models recognize analogies?

RQ2: Is the *silver-set* useful for training models and improving their performance?

RQ3: Can the distractors fool humans and models?

5.1 Tasks

We propose two tasks. *Binary classification* offers a simple and clean formulation; *multiple-choice* is more similar to standardized test questions, adding an aspect of *ranking* among choices.

Binary classification. Given a pair of paragraphs base $\mathcal{B}$ and target $\mathcal{T}$, each describing a scientific process in natural language, the task is to decide whether the processes are analogous. The target paragraph could either be: (1) **Analogy** (positives), (2) **Random** ProPara paragraphs with a different subject than $\mathcal{B}$ (simple negatives, see footnote 2) or

Votes	Size	Drel	Minfo	Cyclic	Other
$= 0$	443	93%	16%	21%	22%
≤ 1	540	85%	23%	19%	27%
≤ 2	648	73%	28%	17%	29%

Table 1: Distribution of issues raised, by #positive annotations: dissimilar relations (Drel), misinformation (Minfo), cyclic vs. non-cyclic (Cyclic), and other. Annotators could choose more than one (hence sum >100%). Most of the issues are with (LLM-generated) relations.

(3) **Distractor** paragraphs (challenging negatives, see Section 3.5). In the benchmark, we balance the samples s.t. 50% of target paragraphs are analogies, 25% are simple negatives and 25% are distractors.
Multiple-choice. Given a base paragraph $\mathcal{B}$, along with four candidate paragraphs, the task is to identify the paragraph that is most analogous to $\mathcal{B}$. We use two different setups. (1) **Basic**: candidates are one analogous paragraph and 3 random paragraphs. (2) **Advanced**: In this setup, we increase the difficulty by including the distractor corresponding to the correct answer. However, this results in always having two extremely similar candidates (the analogous paragraph and its distractor), and both trained models and humans might realize that the correct answer always lies between them. To overcome this issue, we generate distractors both for the correct answer and for the random paragraph, and use them as our four candidates. This way, candidates include *two* pairs of similar paragraphs.

5.2 Baselines

We evaluate both *state-of-the-art LLMs* and *humans* in zero-shot and guided settings.
Models. We tested ChatGPT[3], GPT-4 (OpenAI, 2023), Gemini Pro (Team et al., 2023), FlanT5-small, FlanT5-XL, and FlanT5-XXL (Chung et al., 2022), all with their official implementations and default parameters. See Appendix A.7 for additional models, which performed poorly.

The families of models we have experimented with represent state-of-the-art. Our task poses significant NLP challenges, and recent work by Sultan and Shahaf (2022) suggests that more traditional models such as SBERT (Reimers and Gurevych, 2019) could only identify very close analogies with similar entities. Thus, we decided to focus on recent state-of-the-art LLMs and leave testing of additional models for future work.

We start experimenting in a *zero-shot* setting[4]. In the binary task, we use 620 instances (310 analogies, 155 distractors and 155 random) from our *gold-set*. In the multiple-choice we use the 310 analogous paragraphs as one of the candidates, adding three random paragraphs (basic setup), or a distractor, a random paragraph and a distractor generated for it (advanced setup). See prompts in Appendix A.7.

In addition to the zero-shot setting, we experimented with a **guided setting** to improve the performance of models and humans in the binary task using labeled examples. Where we could fine-tune models, we did. Other times, such as with GPT4 (the best model from zero-shot), we used few-shot examples. We experimented with several prompts, based on successes and failures of the model. Overall accuracy remained similar, and we chose a prompt that includes five mistakes (3 distractors, 1 analogy, 1 random); the rationale was to include more examples of common mistakes. See Appendix A.7.
Humans. In addition to the evaluation of LLMs, we are also interested in assessing the performance of humans on both tasks. We employ new AMT workers, who had not participated in creating the dataset. In both tasks, every instance is evaluated by 3 annotators. We publish the majority vote accuracy, and agreement as the % of perfect agreement. See Appendix A.7 for task instructions.

On the binary task, we run the experiment in two stages, mimicking the zero-shot and guided settings of the models. In the zero-shot setting, we show the crowdworkers 100 randomly sampled instances from the *gold-set*, including 50 positives (equally divided into close and far analogies), 25 simple negatives and 25 challenging distractors.

For the guided setting, we show workers examples based on their errors (similar to what we did with GPT-4). Then, we use another set of samples (with different base paragraphs) with 10 close analogies, 10 far analogies, 10 simple negatives, and 10 distractors. For the multiple-choice task, we show 25 instances for the basic setup, and another 50 for the advanced setup (using different base paragraphs). See Appendix A.7 for the tasks.

5.3 Results

Our results are summarized in Table 2 for the *bi-*

Row	Settings	Method	Overall	Per Target Type		
				Positives (50%)	Negatives (50%)	
				Analogy	Random	Distractor
1		Random Guess	50	50	50	50
2		GPT4	**79.5**	95.2	92.9	34.8
3		ChatGPT	68.2	53.5	96.8	69.0
4	Zero-shot	Gemini Pro	73.9	79.7	100	36.1
5		FlanT5-XXL	61.1	28.1	100	88.4
6		FlanT5-XL	59.7	25.1	100	88.4
7		FlanT5-small	49.3	0	97.4	100
8		Humans	79	58	100	100
9		GPT4 (in-context)	78	86.5	98.1	40.7
10	Guided	FlanT5-small (fine-tune)	74.4	87.1	96.1	27.1
11		Humans	**92.5**	95	100	80

Table 2: The **Overall** and **Per Target Type Accuracy (%)** of *LLMs* and *humans* in *zero-shot* and *guided* settings, on the *binary classification* task, evaluated on the *gold-set*. Out of the models, GPT4 achieves the best overall accuracy (**row 2**). Humans achieve better performance than models (∼13% gap in **Overall Accuracy**) after a cycle of learning from their mistakes (**row 2 vs. row 11**). Interestingly, FlanT5 models tend to output "not analogy" more than GPT-4, rendering their performance on the true negatives higher, but overall their performance is worse (**rows 5-7 vs. row 2**), see **Section 5.3, RQ1**. The training of FlanT5-small on the *silver-set* significantly improved its **Overall Accuracy** (**row 10 vs. row 7**), see **Section 5.3, RQ2**. Comparing challenging negatives (**Distractor**) with simple negatives (**Random**), we observe a performance decline in both humans and LLMs (**rows 2-6, 8-11**), except for FlanT5-small, which almost always predicts "not analogy". This reduction is statistically significant for models but not for humans. We additionally confirmed that the proportion of mistakes due to choosing the challenging distractor is much higher. For more details, see **Section 5.3, RQ3**.

nary classification task, and Table 3 for the *multiple choice* task.

RQ1: What is the performance of humans and models? In the binary task in zero-shot, GPT4 achieves the highest overall accuracy of 79.5%, succeeding on analogies and simple negatives but struggling with distractors. Gemini Pro follows with overall accuracy of 73.9%, then ChatGPT with overall accuracy of 68.2%. Not surprisingly, we can also see that Flan models get better as they grow bigger.

Humans achieve 79% overall accuracy ($\sigma = 0.04$), nearly matching the best model. Interestingly, humans achieved perfect accuracy on simple negatives and distractors, but were too strict and ruled out many correct analogies. Agreement was 70% (random chance 25%). Initially, we expected humans to outperform models. Thus, we set out to explore whether adding a guidance step helps. For the guided settings, we used the best model (GPT4) with few-shot examples of its mistakes. Similarly, we showed the crowdworkers their mistakes. We found that humans were able to improve signif-

icantly, achieving an overall accuracy of 92.5% ($\sigma = 0.014$) and agreement of 80%. We conclude the task is complex, but possible to explain. On the other hand, GPT4's performance is similar, even testing numerous prompt variations (Section 5.2).

We note that this task is harder than the annotation task of Section 3.2. Here, annotators only see the paragraphs (not the similar relations, subjects, or domains). Additionally, they have to decide whether the paragraphs are analogous, as opposed to going over a structured list of potential issues.

In the *multiple-choice* task, the best model is again GPT4, achieving overall accuracy of 95.5% (basic setup) and 83.2% (advanced setup).

For the multiple-choice task, we employed the same annotators from the binary task after guidance. In the basic setup, humans (majority vote) achieve a perfect accuracy of 100% ($\sigma =0.04$), and agreement of 88%. In the advanced setup, an accuracy of 96% ($\sigma = 0.04$), and agreement of 66% (chance agreement is 6.25%).

To conclude, humans achieve better performance than models (∼13% gap) after light supervision;

Row	Settings	Method	Basic	Advanced
1		Random Guess	25	25
2		GPT4	**95.5**	**83.2**
3		ChatGPT	74.2	59
4	Zero-shot	Gemini Pro	87.4	62.6
5		FlanT5-XXL	87.4	75.2
6		FlanT5-XL	68.4	55.5
7		FlanT5-small	32.9	32.9
8	Guided	Humans	**100**	**96**

Table 3: The **Accuracy (%)** of *LLMs* and *humans* in *zero-shot* and *guided* settings, on the *multiple choice* task, evaluated on the *gold-set*. The **Basic** setting uses simple negatives (random), while the **Advanced** includes challenging negatives (distractors). Humans achieve better performance than models (∼13% gap in the advanced setup); out of the models, GPT4 achieves the best results (**row 2**), see **Section 5.3, RQ1**. Distractors reduce performance in both humans and LLMs. This decline is statistically significant for the models, but not for humans (**rows 2-8 Advanced vs. Basic**), see **Section 5.3, RQ3**. Note that here we show the results of models only in zero-shot, as we already addressed **RQ2** – "Is the silver-set useful for training models?" in Table 2. We leave training of models for the *multiple choice* task for future work.

out of the models, GPT4 achieves the best results.

RQ2: Is the *silver-set* useful for training models? We employ FlanT5-small, which is a small model of only 80M parameters, fine-tune it on the *silver-set* (which was automatically generated) for the binary classification task, and test it on the *gold-set*. We choose FlanT5-small to test whether high accuracy can be achieved even with a small model. We use the same prompt from the zero-shot setting. See Appendix A.7 for details about training.

FlanT5-small's overall accuracy improved from 49.3% to 74.4% after fine-tuning, surpassing even the largest Flan model (FlanT5-XXL), in zero-shot (see Table 2). This result is statistically significant with a p-value of 1.3e-06 in the McNemar test, at the 0.05 level with Bonferroni correction.

RQ3: Are the distractors effective? In the binary classification task, we can see that both humans and LLMs (except FlanT5-small, which almost always predicted "not analogy") achieve nearly perfect accuracy on the simple negatives, but lower accuracy on the challenging distractors (see Table 2). In the multiple-choice task, we can see a drop in performance for both LLMs (except FlanT5-small) and humans when transitioning from basic setup without distractors to advanced with distractors (see Table 3). We use the McNemar test to as-

sess statistical significance, reaching p-values of 7e-08 for GPT4, 4.3e-14 for Gemini Pro, 6.3e-06 for ChatGPT, 1.5e-05 for FlanT5-XXL, and 0.0009 for FlanT5-XL (all statistically significant at the 0.05 level after Bonferroni correction). The drop in accuracy for humans was not significant.

Next, we compute the *percentage* of errors resulting from incorrectly choosing the distractor: In humans it is 100%, for GPT4 92.3%, Gemini Pro 75.0%, ChatGPT 66.9%, FlanT5-XXL 62.3%, FlanT5-XL 25.4%, and FlanT5-small 40.5% (random chance 25%). Thus, LLM mistakes mainly stem from selecting the distractor. In the case of humans, the *absolute* number of mistakes is quite small, so we cannot draw a firm conclusion.

6 Conclusions

Analogy-making is crucial for AI to generalize and adapt to unfamiliar contexts. We designed a pipeline, *ParallelPARC*, leveraging LLMs to generate complex analogies and distractors. We demonstrated our pipeline by creating *ProPara-Logy*, a dataset of analogies between scientific processes. *ProPara-Logy* is orders of magnitude larger than previous datasets of full paragraphs, and could easily be expanded via the pipeline.

Our experiments show humans outperform models after light supervision, and that even the best models are more sensitive to distractors than humans. We also show that automatically generated data is useful for training and improving models.

Our pipeline is easy to adapt to new domains, requiring only small changes in the prompts. We hope researchers will use it in domains where analogies have shown promise. For example, in education (Duit, 1991; Clement, 1993), analogies can be used to leverage students' existing knowledge to make abstract or challenging material easier to grasp (e.g., in biology, the heart is frequently compared to a pump to help students understand how it circulates blood throughout the body, similar to how a pump moves water); in computer-assisted creativity (Moreno et al., 2014; Hope et al., 2017), analogies can be used to inspire designers and engineers in solving new problems by using existing ideas from another field (e.g., NASA has embraced the principles of origami to develop foldable solar panels and satellite antennas). We hope this work will spur more NLP work on analogies, leading to novel tasks and benchmarks.

Ethical Considerations

Misuse of analogies. Research has revealed that people often find it difficult to discern nuances or limitations in presented analogies (Holyoak and Thagard, 1996). For example, in Swain (2000) an analogy is used to explain medical students the intricacies of the cardiovascular system by likening it to a city water supply. However, this analogy might also confuse them, as it fails to acknowledge crucial distinctions between water and blood, such as the existence of blood clots. Thus, one might wish to alert people who read analogies generated by our pipeline to this possibility, as well as the possibility of LLM hallucinations.

Crowdsourcing. Human annotations and evaluations were carried out through crowdsourcing (Amazon Mechanical Turk platform). The workers are native English speakers from the US. Workers were compensated at a rate of $15 per hour (higher than the minimum wage in their states). We set the price per HIT by calculating the average completion time for sample HITs.

Dataset. We used the ProPara dataset of paragraphs describing scientific processes in English, taking 390 titles from its training set (allenai.org/data/propara)[5] and generated the ProPara-Logy dataset. We removed all content in the ProPara-Logy that might contain information about the annotators, such as worker IDs. Note that our generated dataset focus is on the scientific domain, limiting cultural or situational biases.

Computation. Zero-shot experiments require about an hour to run both tasks on an NVIDIA A100 GPU, with the majority of the time spent on interactions with the GPT model's API. These experiments are conducted using Google Colab Pro+ on the Ubuntu version of Linux. Fine-tuning experiments, involving both training and inference of FlanT5-small, take less than 15 minutes on an NVIDIA RTX 6000 GPU. These experiments are run from the university cluster, operating on Debian GNU/Linux.

Limitations

Relying on closed models (e.g., OpenAI models). In closed models, the architecture, training data, and training methodologies are not available; furthermore, these models belong to a company and thus might be shut down or deprecated in the future. Nevertheless, these models are considered to be state-of-the-art, are widely in use and have gained significant attention from both experts and non-experts. Thus, we believe it is valuable to use them in this work, acknowledging their limitations. **Sensitivity to prompts.** It is known that LLMs are sometimes sensitive to small changes to the prompts.
Domains. In this work we focused on generated data for scientific processes across several (specific) domains. The results in other domains are yet to be explored.
Language. Our benchmark contains solely English texts. The results may differ in other languages.

References

Bhavya Bhavya, Jinjun Xiong, and ChengXiang Zhai. 2022. Analogy generation by prompting large language models: A case study of InstructGPT. In *Proceedings of the 15th International Conference on Natural Language Generation*, pages 298–312, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

Yonatan Bitton, Ron Yosef, Eli Strugo, Dafna Shahaf, Roy Schwartz, and Gabriel Stanovsky. 2022. Vasr: Visual analogies of situation recognition. In *AAAI Conference on Artificial Intelligence*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *ArXiv*, abs/2005.14165.

Jiangjie Chen, Rui Xu, Ziquan Fu, Wei Shi, Zhongqiao Li, Xinbo Zhang, Changzhi Sun, Lei Li, Yanghua Xiao, and Hao Zhou. 2022. E-kar: A benchmark for rationalizing natural language analogical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3941–3955.

---

[5]https://github.com/allenai/propara (Apache-2.0 license, no explicit intended use)

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

François Chollet. 2019. On the measure of intelligence. *ArXiv*, abs/1911.01547.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

C. Clement and D. Gentner. 1991. Systematicity as a selection constraint in analogical mapping. *Cognitive Science*, 15:89–132.

John J. Clement. 1993. Using bridging analogies and anchoring institutions to seal with students' preconceptions in physics.

Tamara Czinczoll, Helen Yannakoudakis, Pushkar Mishra, and Ekaterina Shutova. 2022. Scientific and creative analogies in pretrained language models. In *Conference on Empirical Methods in Natural Language Processing*.

Bhavana Dalvi, Lifu Huang, Niket Tandon, Wen tau Yih, and Peter Clark. 2018. Tracking state changes in procedural text: A challenge dataset and models for process paragraph comprehension. *NAACL*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Reinders Duit. 1991. On the role of analogies and metaphors in learning science. *Science Education*, 75:649–672.

Dedre Gentner. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2):155–170.

Dedre Gentner, Mary Jo Rattermann, and Kenneth D Forbus. 1993. The roles of similarity in transfer: Separating retrieval from inferential soundness. *Cognitive Psychology*, 25:524–575.

Adam E Green, David J M Kraemer, Jonathan A Fugelsang, Jeremy R Gray, and Kevin N Dunbar. 2010. Connecting long distance: semantic distance in analogical reasoning modulates frontopolar cortex activity. *Cerebral cortex (New York, N.Y. : 1991)*, 20(1):70—76.

Douglas R Hofstadter and Emmanuel Sander. 2013. *Surfaces and essences: Analogy as the fuel and fire of thinking*. Basic books.

Keith J Holyoak. 1984. Analogical thinking and human intelligence. *Advances in the psychology of human intelligence*, 2:199–230.

Keith J Holyoak and Paul Thagard. 1996. *Mental leaps: Analogy in creative thought*.

Tom Hope, Joel Chan, Aniket Kittur, and Dafna Shahaf. 2017. Accelerating innovation through analogy mining. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Cheng Jiayang, Lin Qiu, Tsz Ho Chan, Tianqing Fang, Weiqi Wang, Chunkit Chan, Dongyu Ru, Qipeng Guo, Hongming Zhang, Yangqiu Song, et al. 2023. Storyanalogy: Deriving story-level analogies from large language models to unlock analogical understanding. *arXiv preprint arXiv:2310.12874*.

David Jurgens, Saif Mohammad, Peter Turney, and Keith Holyoak. 2012. SemEval-2012 task 2: Measuring degrees of relational similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 356–364, Montréal, Canada. Association for Computational Linguistics.

Matthew J. Kmiecik, Ryan J. Brisson, and Robert G. Morrison. 2019. The time course of semantic and relational processing during verbal analogical reasoning. *Brain and Cognition*, 129:25–34.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 13–18, Berlin, Germany. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*.

Marvin Minsky. 1988. *Society of mind*. Simon and Schuster.

Melanie Mitchell. 2021. Abstraction and analogy-making in artificial intelligence. *Annals of the New York Academy of Sciences*, 1505.

Diana P. Moreno, Alberto A. Hernández, Maria C. Yang, Kevin N. Otto, Katja Hölttä-Otto, Julie S. Linsey, Kristin L. Wood, and Adriana Linden. 2014. Fundamental studies in design-by-analogy: A focus on domain-knowledge experts and applications to transactional design problems. *Design Studies*, 35:232–272.

R OpenAI. 2023. Gpt-4 technical report. *arXiv*, pages 2303–08774.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only.

Vencislav Popov, Penka Hristova, and Royce Anders. 2017. The relational luring effect: Retrieval of relational information during associative recognition. *Journal of Experimental Psychology: General*, 146:722–745.

Alec Radford, Luke Metz, and Soumith Chintala. 2016. Unsupervised representation learning with deep convolutional generative adversarial networks. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

Scott E. Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. 2015. Deep visual analogy-making. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1252–1260.

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.

Anna Rogers, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *North American Chapter of the Association for Computational Linguistics*.

Fereshteh Sadeghi, C. Lawrence Zitnick, and Ali Farhadi. 2015. Visalogy: Answering visual analogy questions. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1882–1890.

Natalie Schluter. 2018. The word analogy testing caveat. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 242–246, New Orleans, Louisiana. Association for Computational Linguistics.

Oren Sultan and Dafna Shahaf. 2022. Life is a circus and we are the clowns: Automatically finding analogies between situations and processes. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3547–3562, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

David P. Swain. 2000. The water-tower analogy of the cardiovascular system. *Advances in physiology education*, 24 1:43–50.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. 2021. Zero-shot image-to-text generation for visual-semantic arithmetic. *ArXiv preprint*, abs/2111.14447.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Asahi Ushio, Luis Espinosa-Anke, Steven Schockaert, and Jose Camacho-Collados. 2021. Bert is to nlp what alexnet is to cv: can pre-trained language models identify analogies? *arXiv preprint arXiv:2105.04949*.

Christine M Wharton, Keith J Holyoak, Peggy E Downing, Thomas E Lange, Thomas D Wickens, and Evan R Melz. 1994. Below the surface: Analogical similarity and retrieval competition in reminding. *Cognitive Psychology*, 26:64–101.

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*, abs/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*, abs/2305.05994.

A   Appendix

A.1   Reproducibility

A.1.1   Models

The models we used for evaluation are detailed in Section 5.2. Regarding the zero-shot experiments: after loading the models, it takes approximately one hour to run the models on both tasks on an NVIDIA A100 GPU. The majority of this duration is attributed to interactions with the GPT model's API. We run it using Google Colab Pro+ (the operating system is the Ubuntu version of Linux).

Regarding the fine-tuning experiments: both training and inference of FlanT5-small took less than 15 minutes on NVIDIA RTX 6000 GPU. We run it from the cluster of the university (the operating system is: Debian GNU/Linux). Trained models hyper-parameters (and the range of values we tried) are provided in Appendix A.7.2. Full implementation is provided in the attached code.

A.1.2   Statistics

The details about the pipeline generation are provided in Section 3. Dataset Statistics are provided in Section 4. A link to a downloadable version of the dataset is available in the code. A complete description of the annotation process is provided in Sections 3.2, and 3.4.

A.1.3   Code

The attached code includes the full implementation, dependencies, training code, evaluation code, pre-trained models, README files, and commands necessary to reproduce the results presented in the paper.

A.2   Analogy Candidates Generation

See Figures 4, and 5 for our solution using the two prompts. See Figure 6 for an example of what happens when we used one prompt for the whole task.

A.2.1   Model's parameters

For generating the analogy candidates, we use GPT-3.5 (text-davinci-003) (Brown et al., 2020) with temperature=0.7, max_tokens=1000, and top_p=1.

A.3   Automatic Filtering and Labeling

See Figure 7 for the prompt given to the auto-labeling model.

A.3.1   Model's parameters

For our auto-labeling model, we used GPT-4 (OpenAI, 2023) with the following parameters: temperature=0.5, max_tokens=4000, top_p=0.

A.4   Human Annotation

In this section we will give more details about the reasons for further inspection, and the annotation process.

A.4.1   Reasons for further inspection

Here is the list of some popular reasons we found:

- **Dissimilar relations** – when at least one line of relations consists of dissimilar relations. For example: (precipitation, *falls*, on the ground) like (rotor, *rotates*, generator) contains a pair of relations with dissimilar meaning for the verbs "falls" and "rotates".

- **Misinformation** – when one of the paragraphs (or the relations) contain misinformation. For example, one paragraph mentions "rain droplets rise to the atmosphere" instead of "falls to the ground".

- **Cyclic vs. non-cyclic process** – when one paragraph describes a cyclic process and the other not (e.g, one paragraph about the water cycle process which is cyclic, and another on human digestive system, which is not cyclic).

- **Other** – any other reason.

A.4.2   The annotation process

Human annotation task   We start by giving the workers the **instructions** for the task, which include a background on analogies, explanation about the task and the labels. See Figure 9, and Figure 10 for the instruction screens given to the workers in the Amazon Mechanical Turk platform. In addition to the instructions, we supplied 5 full examples (close analogy, far analogy, and 3 candidates for further inspection with different reasons). See Figures 11, 12, 13, 14 and 15 for the five examples. After the workers read the instructions for the task, they performed a **qualification exam** consists of 10 samples (equally divided between analogies and rejected samples). 7 out of 12 workers passed our performance bar – at least 8 out of 10 correct answers. Then, the workers start to annotate *analogy candidates*. The first phase is **initial annotation**, where our 7 highly-qualified workers labeled 130 samples from the *analogy candidates*. We chose

30 random samples with their label of how many workers vote for analogy (between 0 and 3) to feed as in-context few-shot samples to the GPT-4 auto-labeling model.

Human validation The next phase is the **validation**, in which we run our GPT-4 auto-labeling model in batches from the *analogy candidates*, and give the highly-qualified workers to label only samples where the model predicts full agreement. In this way, we filter the most probable analogies and candidates for further inspection.

Workers consent We obtained worker consent for all workers participating in the task. Workers have been told about the objective of the work, and how their annotations will be used. They have also been told they were annotating data generated by AI. Data collection has been approved by the Hebrew University board of ethics.

A.5 Distractors Generation

See Figures 16 and 17 for the two prompts to generate distractors.

A.5.1 Distractors formulation

Here is the formulation of our distractors. Let x and y be two events in $\mathcal{T}$ which describes an analogous process to $\mathcal{B}$, which is a paragraph from the ProPara dataset, in the form of procedural text. An *event* in paragraph is usually described in 1–2 sentences. Let t_x and t_y be the timestamps of events x and y in $\mathcal{T}$, such that $t_x < t_y$, and x must happen before y, in other words x is a prerequisite of y, or y is dependent on x which has to be presented before y. Our aim is to create a *coherent* paragraph $\mathcal{T}'$ such that y will be presented before x in the sequence of events. This distractor paragraph will include similar *first-order* relations, but dissimilar *higher-order* relations, which result in different *cause-and-effect-relationships* and possibly make $\mathcal{T}'$ *illogical*.

A.5.2 Model's parameters

We use GPT-4 with temperature=1.0 (for the one-shot prompt of creating new events order) and temperature=0.00001 (for the second few-shot prompt of creating the distractor paragraph). We used the other default parameters for both prompts.

A.6 Dataset Analysis

Here are the popular issues that annotators found as "Other".

- **Inconsistent mapping**: when the mapping that can be inferred by the supplied relations is inconsistent, which means one entity in the base is mapped to more than one entity in the target

- **Incorrect structure of relation**: the correct format for relations is: (entity1, verb, entity2), but some generated candidates had a wrong format (e.g, (verb, verb, entity)).

- **Relations and paragraphs misalignment**

A.7 Evaluating Humans and LLMs

See Figures 18 and 19 for the display screens to the crowdworkers in Amazon Mechanical Turk for the binary classification task and the multiple-choice task. See Figures 20 and 21 for the prompts given to both humans and models in the zero shot setting for both tasks. See Figure 22 for the prompt given to GPT4 in the supervised setting. This prompt includes five mistakes as few-shot examples from the zero-shot experiment.

A.7.1 Methods

In the evaluation of both the *binary classification* and *multiple-choice* tasks, we employ several state-of-the-art LLMs, including GPT-3.5 (Brown et al., 2020), GPT-4 (OpenAI, 2023), Gemini Pro (Team et al., 2023), FlanT5-XL (3B parameters), and FlanT5-XXL (Chung et al., 2022) (11B parameters). Other models we also considered, but we did not include are: Falcon, Flacon-instruct (Penedo et al., 2023), and Alpaca (Taori et al., 2023) with their 7B version, and Vicuna (Chiang et al., 2023), LLAMA, and LLAMA2 with 7B and 13B versions (Touvron et al., 2023a,b). We did not include the results of these models, since they failed to understand the task (chose the same candidate in the multiple-choice task or outputted an empty string).

A.7.2 FlanT5-small Fine-tune Parameters

We use the default AdamW (Loshchilov and Hutter, 2017) optimizer, a learning rate of 1e-5 (other learning rate values we tried are 1e-3), batch size of 16 (we tried a different batch sizes including 4, 8, and 32), and train for 7 epochs (we tried a different number of epochs in the range of 1 to 20). The metric is "overall accuracy" (remains relatively stable).

Your task is to find an analogy between BASE and TARGET.

Here are the instructions for the format of relations you should provide in SIMILAR_RELATIONS.

Every similar relation should be in the following format: (ENTITY1_BASE, VERB_BASE, ENTITY2_BASE)

like (ENTITY1_TARGET, VERB_TARGET, ENTITY2_TARGET).

ENTITY1_BASE and ENTITY2_BASE must be noun phrases from BASE.

ENTITY1_TARGET and ENTITY2_TARGET must be noun phrases from TARGET.

VERB_BASE and VERB_TARGET must be verbs with the same meanings.

Inputs: BASE, TARGET_DOMAIN

Outputs: TARGET, TARGET_FIELD, SIMILAR_RELATIONS

Inputs:

BASE: How does the electrical circuit works?

TARGET_DOMAIN: One of the fields of Engineering

Outputs:

TARGET: How does a mechanical system of water pump works?

TARGET_FIELD: Mechanical Engineering

SIMILAR_RELATIONS:

(battery, generates, electrical voltage) like (pump, generates, pressure)

(electrons, move through, copper wire) like (water, move through, pipe)

(resistor, decrease, voltage rate) like (valve, decrease, flow rate)

Figure 4: A one-shot prompt for finding a target analogous subject and generating the similar relations between base and target.

Your task is to write a paragraph given SUBJECT and RELATIONS.

PARAGRAPH has to include RELATIONS in the text.

Inputs: SUBJECT, RELATIONS

Outputs: PARAGRAPH

Inputs:

SUBJECT: How does the electrical circuit work?

RELATIONS:

(battery, generates, electrical voltage)

(electrons, move through, copper wire)

(resistor, decrease, voltage rate)

Outputs:

PARAGRAPH:

The battery generates electrical voltage.

This voltage creates a potential difference that causes electrons to flow through the circuit.

The electrical voltage causes electrons to move through the copper wire.

The electrons pass through the resistor.

The resistor presents a higher resistance to the flow of electrons,

which causes a decrease in the voltage of the circuit.

Figure 5: A one-shot prompt for writing a target paragraph given the subject and the relations in target.

$\mathcal{B}$: **How is sediment transported across the Earth?**	$\mathcal{T}$: **How is money transported across the economy?**
Sediment settles in a place due to gravity.	Money flows through the economy.
The sediment breaks down into small particles.	Money settles in different places.
Wind or water picks up the sediment.	Money breaks down into smaller denominations.
The sediment travels along the same route as the wind or water.	Investment or spending causes money to move.
The sediment is deposited at new locations by the wind or water.	Money is deposited into new accounts.
The sediment is picked up again by new wind or water.	The money is picked up again by new investment or spending.
The sediment travels further.	**Money travels through the economy**.
The sediment is deposited again in a new place.	**Money is deposited again in a new place.**

Figure 6: An example of an analogous target paragraph ($\mathcal{T}$) of "How is money transported across the economy" to a base paragraph ($\mathcal{B}$) which is about "How is sediment transported across the Earth?", using one prompt for the whole task of both finding the analogous target subject and writing the paragraph, generated by GPT-3.5. As we can see, using one-prompt lead to paragraphs which are mostly identical other than the nouns ("The sediment is deposited again in a new place"/"Money is deposited again in a new place"), and to artificially sounding sentences (e.g, "Money travels through the economy").

Analogies candidates Auto-labeling Prompt

Your task is to rate how analogous are paragraph pairs from 0 (non-analogous) to 3 (very analogous) based on whether they describe similar underlying processes or mechanisms.

SOURCE-SUBJECT: How do floods happen?
SOURCE-PARAGRAPH: Floods happen when there is excessive rainfall which increases the water levels in rivers and streams. When the water levels get too high, the rivers and streams will overflow their banks. Additionally, heavy rainfall can also cause groundwater to rise above ground. This can lead to flooding as well.
TARGET-SUBJECT: How does a social movement develop?
TARGET-PARAGRAPH: A social movement begins with the spread of ideas among people. As more individuals learn about the movement and join it, support for the cause grows. This support often includes donations, participation in protests, and other forms of support, which helps to further the cause of the social movement.
RELATIONS: (rainfall, increases, water levels) like (ideas, spread, among people).
(rivers, overflow, banks) like (individuals, join, the movement).
(groundwater, rises, above ground) like (support, grows, for the cause)
LABEL: 0

Figure 7: The beginning of the prompt we used for analogous paragraph's candidate auto-labeling, where no annotator classified the example as an analogy. This is one example out of 30 few-shot.

Figure 8: The display screen of the annotators in Amazon Mechanical Turk. The worker has to choose one of *close analogy* or *far analogy* in the case of analogy, or the reasons for possible issues in the generation. If the worker chooses *not analogy - other*, filling the text box with the other reason is mandatory. Note that this example is not analogy (in the current form), hence it should be postponed for further inspection. The issues raised are: *dissimilar relations* ("planted" vs. "replicated") and cyclic/non-cyclic (the target paragraph is a cyclic process, while the source paragraph is not)

5915

Analogy

Analogy is a **mapping** between entities in two domains **source** and **target** , that relies on **relational similarity,** rather than **object attributes.** For example, given a **source paragraph** on: "How does an electrical circuit work?", and a **target paragraph** on: "How does a mechanical water pump work?", one mapping between entities in the domains is: electrons → water. Note that their attributes are **different** (e.g., water can be in a state of liquid, while electrons are not considered to be in any state of matter), but we care on **relational similarity** (e.g., both share a similar relation that the electrons move through the copper wire, like the water moves through the pipe).
In this example the paragraphs describe an analogous process.

Some **mappings** between entities that play a similar role:
battery → pump
electrons → water
copper wire → pipe
resistor → valve

Similar **relations** include:
(battery, generates, electrical voltage) like (pump, generates, pressure)
(electrons, move through, copper wire) like (water, move through, pipe)
(resistor, decrease, voltage rate) like (valve, decrease, flow rate)

* Note that similar relations means that the verbs have a **similar meaning**, but does not have to be **identical**. For example, (battery, **creates**, electrical voltage) like (pump, **generates**, pressure)
is also a similar relation as **"creates"** and **"generates"** have a similar meaning.

* Note that some verbs in the relations could be **non-indicative** (e.g, **auxiliary verbs** such as "be", "do", "have", and **linking verbs**. For example if in one paragraph we have **"X was something"** and in the other paragraph we have **"Y was something"** it's not indicative enough to hint that X and Y is a good mapping.
If you found relations with these verbs only, so it's not a good analogy. We are looking for relations which are similar but with indicative verbs such as "flows", "destroy", "create" and many more indicative verbs.

The task

Given a **source paragraph, target paragraph,** and the **relations** between them, your task is to determine whether the pair of paragraphs describe an analogous process, such that there is a mapping between entities according to relational similarity (entities that play a similar role), as explained above.

Detailed instructions

Please read the Detailed Instructions in the next tab

Figure 9: The instructions screen of the annotators in Amazon Mechanical Turk. It includes a background on analogies, and explanation about the task.

Some reasons why paragraphs are *not* analogous

- **Misinformation:**
  when one of the paragraphs (**or the relations**) contain misinformation. For example, one paragraph mentions "rain droplets rise to the atmosphere" instead of "falls to the ground"

- **Dissimilar relations:**
  when at least one line of relations consists of dissimilar relations. For example:
  (precipitation, **falls**, on the ground) like (rotor, **rotates**, generator)
  This line contains a pair of relations with dissimilar verbs "falls" and "rotates".

- **Cyclic vs. non-cyclic process:**
  For example, one paragraph on water cycle (cyclic), and the other on human digestive system (non-cyclic).

- **Other:**
  Any other reason.

Types of Analogy

If two paragraphs describe analogous processes, you should specify the type of analogy.

Close analogy: when the topic is close, and the entities from a similar domain. For example, a **source paragraph** about "How does rain form?", and a **target paragraph** about "How does snow form?"

Far analogy: when the topics are unrelated, and the entities are different. For example, a **source paragraph** about "How does a solar panel work?", and a **target paragraph** about "What happens during photosynthesis?"

Figure 10: The detailed instructions screen of the annotators in Amazon Mechanical Turk. It includes the reasons why a sample is currently not analogous (in its current form), hence is given for further inspection, as well as the types of analogy.

Source domain:

Natural Sciences

Source subject:

What causes a volcano to erupt?

Source paragraph:

When magma heats up underground water, pressure begins to build up inside the volcano. This pressure forces the magma to push against the walls of the volcano, leading to an eruption. The eruption is caused by the pressure of the magma against the walls of the volcano, as well as the heat of the magma heating up the underground water.

Target domain:

Engineering

Target subject:

What causes a boiler to explode?

Target paragraph:

Steam heats the liquid inside the boiler, causing the pressure to build up. As the pressure increases, the steam begins to push against the walls of the boiler. The increased pressure can be too much for the boiler to take and if the pressure exceeds the boiler's capacity, it can cause an explosion.

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)

Figure 11: An example for a sample from the *analogy candidates* given to the workers in the phase of instructions. This example is labeled as a *far analogy*.

Source domain:

Natural Sciences

Source subject:

How do caverns form?

Source paragraph:

Caverns are formed when erosion carves rock. The erosive forces of wind, water, and ice can wear away at solid rock to create a cave. The cave is then further sculpted by the process of dissolution, which is when water dissolves the rock and creates a chamber. Sediment can also fill in a chamber, further adding to the formation of a cavern.

Target domain:

Natural Sciences

Target subject:

How do underground aquifers form?

Target paragraph:

Underground aquifers form through the process of permeation. The process of permeation carves out spaces in the sediment, allowing infiltration of water. This infiltration creates an aquifer, which is a type of underground reservoir. The water then fills the aquifer creating an underground water source.

Relations:
(erosion, carve, rock) like (permeation, carve, sediment)
(dissolution, create, cave) like (infiltration, create, aquifer)
(sediment, fill, chamber) like (water, fill, aquifer)

Figure 12: An example for a sample from the *analogy candidates* given to the workers in the phase of instructions. This example is labeled as a *close analogy*.

Source domain:

Natural Sciences

Source subject:

How does snow form?

Source paragraph:

Snow is formed when water vapor condenses into tiny droplets in the atmosphere. When the temperatures decrease, the droplets freeze and form snowflakes. The colder the temperatures, the more droplets are likely to freeze, resulting in snow precipitation.

Target domain:

Social Sciences

Target subject:

How does a political system work?

Target paragraph:

In a political system, people's ideas are condensed into a shared opinion. Ideas shape policy and the public's attitudes toward it. If public attitudes decrease the support for a policy, then it is likely that the policy will be changed or abandoned.

Relations:
(water vapor, condenses, tiny droplets) like (people's ideas, condenses, a shared opinion)
(droplets, **freeze** , snowflakes) like (ideas, **shape**, policy)
(temperatures, decrease, snow precipitation) like (public attitudes, decrease, support for a policy).
 * In the second line of relations, the verbs "freeze" and "shape" are dissimilar

Figure 13: An example for a sample from the *analogy candidates* given to the workers in the phase of instructions. This example is labeled with the reason of *dissimilar relations*.

Source domain:

Natural Sciences

Source subject:

How does snow form?

Source paragraph:

The water cycle is a cycle of continuous movement of water on the Earth. During the water cycle, evaporation creates water vapor from the oceans, lakes and rivers. This vapor then rises and condenses to form clouds. The clouds then release precipitation such as rain and snow, which returns water to the surface of the Earth. The runoff carries the water into streams and rivers, which will eventually return to the oceans and start the cycle again.

Target domain:

Natural Sciences

Target subject:

How do clouds form?

Target paragraph:

When water vapor condenses, tiny droplets form. As the temperature of the air decreases, the droplets become larger and heavier. The droplets eventually become too heavy for the air to hold and **rise higher into the atmosphere** , where they form clouds.

Relations:
(water vapor, condenses, snowflakes) like (water vapor, condenses, droplets)
(temperature, decreases, snowflakes) like (temperature, decreases, droplets)
(air, rises, snowflakes) like (**air, rises, droplets**)
 * Misinformation in both the target paragraph and the relations (in red).

Figure 14: An example for a sample from the *analogy candidates* given to the workers in the phase of instructions. This example is labeled with the reason of *misinformation*.

Source domain:

Natural Sciences

Source subject:

What happens during the water cycle?

Source paragraph:

The water cycle is a continuous cycle of water molecules being evaporated into the air from the oceans. Condensation then forms clouds of water vapor which eventually precipitate back to the land, adding water to rivers, lakes, and streams. Runoff carries this water from the land back to the oceans, completing the cycle. Evaporation takes water molecules from the oceans and begins the cycle again.

Target domain:

Biomedical and Health Sciences

Target subject:

How does the human body's digestive system work?

Target paragraph:

The human body's digestive system begins with ingestion, which is the act of taking food from the environment. Secretion adds enzymes to the stomach, which helps the stomach break down the food. Digestion further breaks down the food so that the body can absorb the nutrients. Finally, absorption carries the nutrients to the cells, providing them with the energy they need.

Relations:
(evaporation, takes water from, oceans) like (ingestion, takes food from, environment)
(precipitation, adds water to, land) like (secretion, adds enzymes to, stomach)
(condensation, forms clouds of, water vapor) like (digestion, breaks down, food)
(runoff, carries water to, rivers) like (absorption, carries nutrients to, cells)

Figure 15: An example for a sample from the *analogy candidates* given to the workers in the phase of instructions. This example is labeled with the reason of *cyclic vs. non-cyclic process*.

```
Events new order Prompt
```
Your task is given an INPUT_PARAGRAPH, create a list of events in the paragraph
INPUT_PARAGRAPH_ORDER according to their order in time. Then, find two events where the first
event must happen before the second event (a prerequisite).

Then, replace these events and create NEW_PARAGRAPH_ORDER which are the events in the new
order after the replacement, so the NEW_PARAGRAPH_ORDER is illogical.

You also need to write an EXPLANATION why NEW_PARAGRAPH_ORDER is illogical.

Inputs: INPUT_PARAGRAPH

Outputs: INPUT_PARAGRAPH_ORDER, NEW_PARAGRAPH_ORDER, EXPLANATION

Inputs:
INPUT_PARAGRAPH:
When you get a cut on your hand, the first step is to clean the wound with antiseptic.
This is to help prevent infection. Then, a bandage is applied to cover the injury and protect it from further
harm.
As time passes, the healing process begins, in which the incision is gradually closed up.

Outputs:
INPUT_PARAGRAPH_ORDER:
[1]: A cut occurs on your hand.
[2]: The wound is cleaned with antiseptic to help prevent infection.
[3]: A bandage is applied to cover the injury and protect it from further harm.
[4]: The healing process begins and the incision is gradually closed up.

NEW_PARAGRAPH_ORDER:
[1]: A cut occurs on your hand.
[2]: A bandage is applied to cover the injury and protect it from further harm.
[3]: The wound is cleaned with antiseptic to help prevent infection.
[4]: The healing process begins and the incision is gradually closed up.

EXPLANATION:
By looking at INPUT_PARAGRAPH_ORDER, I found that "[2]: The wound is cleaned with antiseptic to
help prevent infection."
must happen before "[3]: A bandage is applied to cover the injury and protect it from further harm."
because the wound should be cleaned first before a bandage can be applied.
After a bandage is applied and cover the wound, the wound cannot be cleaned, since the bandage cover
the wound, and we have no access to the wound, so it's illogical.

Figure 16: A one-shot prompt for generating the events order in the paragraph, as well as new events order after
replacing two dependent events, along with explanation of choosing the specific events.

`Writing a coherent paragraph Prompt`

Your task is to concatenate the EVENTS_ORDER according to the temporal order in EVENTS_ORDER which is wrong, to create illogical paragraph.

Inputs: EVENTS_ORDER
Outputs: OUTPUT_PARAGRAPH

Inputs:
EVENTS_ORDER:
[1]: A cut occurs on your hand.
[2]: A bandage is applied to cover the injury and protect it from further harm.
[3]: The wound is cleaned with antiseptic to help prevent infection.
[4]: The healing process begins and the incision is gradually closed up.

Outputs:
OUTPUT_PARAGRAPH:
[1] When you get a cut on your hand, [2] the first step is to apply a bandage to cover the injury and protect it from further harm. [3] After a bandage is applied, the next step is to clean the wound with antiseptic to help prevent infection. [4] As time passes, the healing process begins, in which the incision is gradually closed up.

Figure 17: The beginning of a few-shot prompt for writing a new coherent paragraph according to the new events order after replacement of two dependent events. We show here the first out of five-shot examples in the prompt.

View instructions

Source paragraph:

A virus is a microorganism that invades cells and replicates itself. The virus reproduces itself within the cells, using the cell's resources to make more copies of itself. This process can cause damage to the cell, resulting in disease. The virus is responsible for the disease, as it causes the cells to malfunction or die.

Target paragraph:

Bacteria invades the tissues of the body, causing potential infections. The bacteria's presence then leads to illness by releasing toxins that damage the host's cells and tissues. After causing illness, the bacteria multiplies and can start spreading to other body parts.

○ **Analogy**   ○ **Not Analogy**

Submit

Figure 18: The screen for the crowdworkers in AMT for the binary classification task.

Source paragraph:
Plants obtain water from their surroundings through their roots. The roots absorb water molecules from the soil and this water is then transported to the leaves and other parts of the plant. The water molecules then travel up the plant through the xylem. At the same time, the stomata on the leaves transpire water molecules into the atmosphere. This process helps to provide water to the plant and cool it down.

Target paragraph:
The human body obtains water through the intestine, which absorbs the water from food and drink. Water is also released from the body as sweat through the skin. The capillaries transport water to the cells in the body, and water is perspired through the skin when the body becomes too hot. The water is then used to regulate body temperature and to lubricate the joints and organs.

‹ 01 ›

Please choose the candidate which is an analogous target paragraph:
○ 1   ○ 2   ○ 3   ○ 4

Submit

Figure 19: The screen for the crowdworkers in AMT for the multiple-choice task. The annotator can press the button (right) and scroll to different target paragraphs (four in total).

Binary Classification Task Prompt

In this task, you'll be given two paragraphs that describe scientific processes. Your goal is to decide whether the processes are analogous. Analogy is a mapping in which the objects of one process are structurally aligned with the objects of another. It is based on similarity of the relationships between the objects and the roles they play throughout the process, and not on the similarity between object attributes. For example, there is an analogy between a paragraph about "How does an electrical circuit work?", and a paragraph about "How does a mechanical water pump work?". In this analogy, electrons are mapped to water: both start at some state (low voltage/low pressure), then move through something (wire/pipe), and change their state (high voltage/high pressure) because of another object (battery/pump). Similar first order relations between the domains include: (battery, creates, electrical voltage) like (pump, generates, pressure) (electrons, move through, copper wire) like (water, flows through, pipe). On the other hand, if for example the second paragraph about the pump is describing that: first the water flows inside the pipe, and following this the pump creates pressure, it changes the cause and effect relationship (higher order relation) to be different from the first paragraph about the electrical circuit, and in this case, the processes are not analogous.

Answer "1" if the two paragraphs describe analogous processes, and "0" if not.

Figure 20: The prompt given for both humans and LLMs in the binary classification task

Figure 21: The prompt given for both humans and LLMs in the multiple-choice task

In this task, you'll be given two paragraphs that describe scientific processes. Your goal is to decide whether the processes are analogous. Analogy is a mapping in which the objects of one process are structurally aligned with the objects of another. It is based on similarity of the relationships between the objects and the roles they play throughout the process, and not on the similarity between object attributes. For example, there is an analogy between a paragraph about "How does an electrical circuit work?", and a paragraph about "How does a mechanical water pump work?". In this analogy, electrons are mapped to water: both start at some state (low voltage/low pressure), then move through something (wire/pipe), and change their state (high voltage/high pressure) because of another object (battery/pump). Similar first order relations between the domains include: (battery, creates, electrical voltage) like (pump, generates, pressure) (electrons, move through, copper wire) like (water, flows through, pipe). On the other hand, if for example the second paragraph about the pump is describing that: first the water flows inside the pipe, and following this the pump creates pressure, it changes the cause and effect relationship (higher order relation) to be different from the first paragraph about the electrical circuit, and in this case, the processes are not analogous. Answer "1" if the two paragraphs describe analogous processes, and "0" if not.

Inputs: First Paragraph, Second Paragraph

Outputs: Answer

First Paragraph:

A wind-powered power station generates electricity by using wind turbines that capture kinetic energy from the wind. This energy is then converted by a generator into electricity, which then flows through power lines to be used in homes and businesses. The wind turbine captures the kinetic energy of the wind and converts it into electrical energy by spinning a generator, which then causes electricity to flow through the power lines.

Second Paragraph:

Solar energy is captured by the solar panels. The electricity generated can then be used to power various electrical appliances. Afterward, the generator converts solar energy into electricity. Finally, electricity flows through wires to reach the appliances.

Answer: 0

First Paragraph:

Floods happen when heavy rain saturates the soil, causing water to accumulate in low-lying areas. The excess water can cause the ground to become unstable, leading to flooding.

Second Paragraph:

A heavy snowfall saturates the mountain slope. This instability then causes the snow to break loose. After the snow breaks loose, it accumulates on the steep slopes. As the snow accumulates, it becomes increasingly unstable. Finally, the avalanche is created.

Answer: 0

First Paragraph:

Bats use echolocation to navigate and find food. They emit high frequency sound waves that bounce off of objects in their environment. The bats then receive the echoes and interpret the information to locate their prey and navigate their surroundings. The echo provides the bats with information about the shape, size, and distance of the object.

Second Paragraph:

Submarines interpret the echo to determine the distance and size of the object. After interpreting the echo, they emit sound waves, which travel through the water and bounce off the objects. These sound waves are then received back as an echo. Finally, submarines use sonar technology to detect objects in the water.

Answer: 0

First Paragraph:

Floods happen when there is an excessive amount of rainfall in a certain area. The rain causes the ground to be saturated, leading to flooding. The flood water can damage buildings and crops, as well as cause disruption to transport and other infrastructure. In addition, rivers can overflow their banks due to the high levels of water, leading to even further flooding.

Second Paragraph:

The wind causes vibration and can damage structures, so engineers must design bridges to withstand the forces the wind exerts. The wind can produce a force that pushes the bridge sideways and could cause it to collapse if not designed properly. Engineers must build bridges in such a way that the wind does not exert too much force on the bridge, and that the bridge is able to withstand the vibration caused by the wind."

Answer: 1

First Paragraph:

Igneous rocks are formed from molten material. This molten material is known as magma and it solidifies into rock as it cools. As the magma cools, crystals form within it, creating the igneous rock. The combination of the cooling of magma and the formation of crystals is what creates igneous rock.

Second Paragraph:

People come together to form a social movement. The organization of people, who have a common goal, creates a movement. Social movements are formed from collective action, as individuals come together to fight for a shared cause. By uniting, people can accomplish goals that they cannot achieve on their own.

Answer: 1

Figure 22: The few-shot prompt given to GPT4 on the binary classification task. It includes 5 examples of mistakes made by GPT4 in the zero-shot experiment. Three on distractors, one on analogy, and one on random.