CAM: A Large Language Model-based Creative Analogy Mining Framework

Bhavya Bhavya (https://orcid.org/0000-0003-0509-3173), University of Illinois at Urbana-Champaign, USA, bhavya2@illinois.edu **(mailto:bhavya2@illinois.edu)**
Jinjun Xiong (https://orcid.org/0000-0002-2620-4859), University at Buffalo, USA, jinjun@buffalo.edu **(mailto:jinjun@buffalo.edu)**
Chengxiang Zhai (https://orcid.org/0000-0002-6434-3702), University of Illinois at Urbana-Champaign, USA, czhai@illinois.edu **(mailto:czhai@illinois.edu)**

DOI: https://doi.org/10.1145/3543507.3587431 (https://doi.org/10.1145/3543507.3587431)
WWW '23: Proceedings of the ACM Web Conference 2023 (https://doi.org/10.1145/3543507), Austin, TX, USA, April 2023

Analogies inspire creative solutions to problems, and facilitate the creative expression of ideas and the explanation of complex concepts. They have widespread applications in scientific innovation, creative writing, and education. The ability to discover creative analogies that are not explicitly mentioned but can be inferred from the web is highly desirable to power all such applications dynamically and augment human creativity. Recently, Large Pre-trained Language Models (PLMs), trained on massive Web data, have shown great promise in generating mostly known analogies that are explicitly mentioned on the Web. However, it is unclear how they could be leveraged for mining creative analogies not explicitly mentioned on the Web. We address this challenge and propose Creative Analogy Mining (CAM), a novel framework for mining creative analogies, which consists of the following three main steps: 1) Generate analogies using PLMs with effectively designed prompts, 2) Evaluate their quality using scoring functions, and 3) Refine the low-quality analogies by another round of prompt-based generation. We propose both unsupervised and supervised instantiations of the framework so that it can be used even without any annotated data. Based on human evaluation using Amazon Mechanical Turk, we find that our unsupervised framework can mine 13.7% highly-creative and 56.37% somewhat-creative analogies. Moreover, our supervised scores are generally better than the unsupervised ones and correlate moderately with human evaluators, indicating that they would be even more effective at mining creative analogies. These findings also shed light on the creativity of PLMs [1].

CCS Concepts: • **Information systems** → **Web mining**;

Keywords: analogy mining, creativity, large language model

ACM Reference Format:
Bhavya Bhavya, Jinjun Xiong, and Chengxiang Zhai. 2023. CAM: A Large Language Model-based Creative Analogy Mining Framework. In *Proceedings of the ACM Web Conference 2023 (WWW '23), April 30--May 04, 2023, Austin, TX, USA*. ACM, New York, NY, USA 12 Pages. https://doi.org/10.1145/3543507.3587431 **(https://doi.org/10.1145/3543507.3587431)**

1 INTRODUCTION

Analogies make connections between concepts as well as their attribute values and are at the foundation of creativity and cognition [8, 24]. They are a form of combinational creativity (unfamiliar combination of familiar ideas) that exploit shared conceptual structure and are widely used in science as well as art. [9]. For example, analogies can facilitate creative designs as they might help identify non-obvious, useful connections [4]. Similarly, they also inspire scientific discovery [20], such as, Bohr-Rutherford solar system model of the atom and William Harvey's description of the heart as a pump. Analogies are also useful for explaining a complex concept to a learner since they map the complex concept to a more familiar concept that is easier to understand [23].

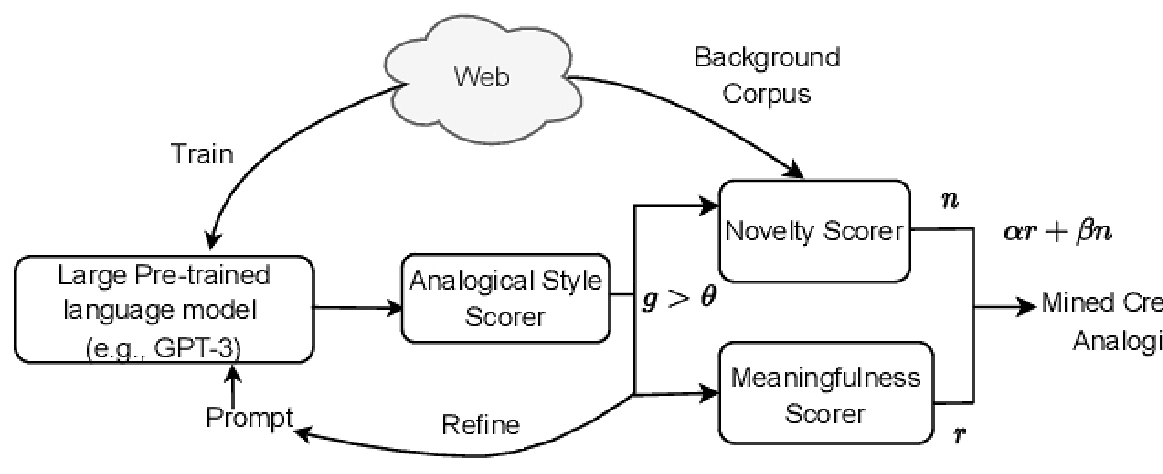


Figure 1: CAM framework for creative analogy mining.

Thus, it is naturally desirable to facilitate the discovery (or creation) of analogies. However, how to generate such creative artifacts at a large scale remains an open question. According to the Stage-and-Loop-based theories of creativity [37], the process of making creative products is multi-stage and iterative. For example, Wallas [62] suggests a four-stage theory based on case studies of idea generation in scientists: *preparation*, *incubation*, *inspiration*, and *verification*. These stages correspond to gathering information related to the task, pondering the information and making connections, getting an idea based on a novel way of looking at the information, and analyzing and polishing the idea into a finished product.

For gathering information, the Web is an excellent resource since it contains all kinds of information and knowledge. It also contains many examples of analogies that are mentioned directly in a webpage while many others could be inferred from multiple webpages using combinational creativity; we refer to such analogies as *creative* analogies. Accordingly, being able to mine creative analogies from the Web is a highly attractive goal because it would enable us to create a large repository of analogies. This repository could then be used to support various applications, such as, creative writing. Moreover, as the Web grows, the analogy repository can grow naturally over time. However, it is impossible for humans to manually discover creative analogies due to the sheer scale of the Web and difficulty in analogical inference across webpages.

Table 1: Sample creative analogy mined using our framework.

Target	Domain	Analogy
Candidate Generation	Machine Learning	A good analogy for understanding candidate generation is to think of it as sifting through a pile of sand looking for gold. The "candidate" in this analogy would be the gold, and the machine learning algorithm is the person doing the sifting. In order to find candidates, the algorithm needs to have some way of identifying which pieces of sand are worth further inspection. This might be done by checking how shiny each piece of sand is, or by measuring its weight.

Thus, we study how to automatically mine a new kind of knowledge from the Web that has not yet been studied, i.e., mining creative analogies to augment human creativity. Although much research has been done on analogy modeling [18, 43], how to mine the Web to extract analogies at large scale remains an open challenge. Therefore, we propose to study this problem in this paper.

Recently, large language models, such as GPT-3 [10] have been trained on massive Web data. In a way, they have already undergone the "preparation" stage of creativity from the Web to complete a variety of tasks. Indeed, a recent work has even demonstrated great promise of prompting large, pre-trained neural language models to generate analogies [7]. However, the analogies generated in that work are generally known analogies, meaning that they have appeared in the Web data used to train the language models. In this paper, we investigate whether PLMs can also be used to generate creative analogies that are not explicitly mentioned on the Web.

Specifically, we propose a novel Creative Analogy Mining (CAM) framework that leverages large-scale pre-trained language models (PLMs) with effectively designed prompts and multiple scoring functions to quantitatively measure three desirable properties of a creative analogy, i.e., its analogical style, coherent meaningfulness, and novelty. The basic idea of CAM is to prompt a PLM appropriately to elicit the knowledge about concepts learned by the PLM (via training on the Web data) and encourage the PLM to perform analogical reasoning so that it would generate not only analogies explicitly discussed on the Web but also potentially novel analogies. This roughly corresponds to the "incubation" and "inspiration" stages of creativity. The scoring functions can then be used to rank and identify the creative analogies generated by the PLM, roughly corresponding to the "verification" stage of creativity. The CAM framework also includes an iterative process to potentially mine more creative analogies by iterative prompting. Figure 1 illustrates this framework (section 4). Our framework is general enough to be used either in an unsupervised or supervised way.

We evaluate the proposed CAM framework by human evaluation of analogies mined from three domains, i.e., cybersecurity, machine learning, and middle school science. We find that the unsupervised instantiation of the framework is effective to generate 13.7% highly creative analogies and 56.37% somewhat-creative analogies. Table 1 shows a sample analogy mined using our framework that was rated as being creative by humans. We also conduct comprehensive experiments to assess our scoring functions against human ratings of the generated analogies, and find that our supervised scoring functions correlate moderately with human ratings, indicating that a human-in-the-loop or co-creative system would be even more effective at mining creative analogies.

Overall, the promising results of CAM not only imply that it can be used to power interesting applications such as providing creative analogy examples, e.g., to explain a concept for improved online learning, but also suggest that PLMs can be used as a powerful creativity mining tool for performing many interesting text mining and discovery tasks on the Web, opening up a novel paradigm of mining the Web at scale that can naturally adapt to the Web content via continuously training/adapting the PLMs to the new Web data.

2 RELATED WORK

In this section, we discuss related work in three main areas: Computational Creativity, Analogy Discovery, and Web and Text Mining.

2.1 Computational Creativity

Computational creativity is an active research area that aims to study the development of computational systems that demonstrate creative behavior [13]. Data mining and machine learning approaches have commonly been used to generate a variety of creative artifacts [19, 56], including both non-text and text (e.g., images [33], metaphors [60], drugs and proteins [29]). For text-based combinational creativity, typically statistical pattern mining approaches have been used, for example, for metaphor generation based on word-co-occurrence [59, 60].

However, such methods have limited power and recently, pre-trained language models have shown a greater success in generating creative text, such as riddles [40], satire [68], and poems [6], as they have already been trained on massive text data. In this direction, one line of work fine-tunes the models on curated datasets [40, 68]. Here, the loss function is optimized to generate the desired creative texts and often requires a lot of additional data. On the other hand, prompting pre-trained language models [42] is more desirable as it can be performed in a zero-shot or few-shot manner. In this paradigm, the given task is directly framed as a language modeling task, where given a textual input or prompt with unfilled slots, the language model is tasked to fill those slots to obtain the output.

There is some work that leverages prompt-based PLMs as a fully automatic tool (e.g., for analogy generation [7]) or as part of a human-machine collaborative tool for supporting humans in creative tasks, such as idea generation [16] and story-writing [54]. However, unlike our work, they do not systematically evaluate and refine the generated artifcats in a unified framework. But, evaluation is a challenging yet important component of creative systems [37]. In the generate-then-test setting, the creative process is divided into separate phases for generating and evaluating the creative artifacts respectively [19]. For example, PIERRE [44] is a recipe generation system that uses a genetic algorithm for generation and neural networks trained on user ratings for evaluation.

We propose a similar idea of using multiple, light-weight scoring functions to evaluate the creativity of the PLM-generated analogy for a desired application. Technically, our idea is similar to using a trained model, called verifier, to assess the prompt-based PLM model outputs [12, 70]. Leveraging user feedback has also recently been used in a reward model and reinforcement learning-based framework to make large language models better at following user intent [47]. Our proposed supervised scorers and user evaluation data can be easily integrated in such a framework. Importantly, to the best of our knowledge, we are the first ones to apply a PLM-based generate and evaluate framework for a creative data mining task that can generate insights to augment human creativity.

2.2 Analogy Discovery

Much work has been done on discovering analogies mostly in the context of idea mining [4, 64]. Our work can be positioned as in the category of mining ideas from text information [55], including, e.g., patents, literature articles [2, 51, 71], social media data [36], information on the Web [2], using various techniques such as topic modeling [3, 63], deep learning [25], information retrieval [11], bibliometric analysis [46], prompting pre-trained language models [7, 57]. Among all the previous work, from the perspective of Web mining at scale, the work [7] is most promising. Our work is a direct extension of the work [7] to enable mining of creative analogies from the Web that are unseen on the Web.

Virtually all the work on analogy mining attempts to capture semantic similarity in some way. For example, Hope et al. [25] formulated the analogy mining task as to retrieve similar text (product descriptions) for scientific innovation based on similarities between their learned representations. In contrast, we use a unified framework based on prompting a pre-trained large-scale language model, which enables our framework to both retrieve/extract explicitly mentioned analogies on the Web and generate novel analogies.

Our work can also be combined with other lines of work, including combining computational analysis with human input (e.g., discovering business ideas using crowdfunding [38]) and application-specific mining (e.g., analogy search engine for product design [21]), to enable them to use more knowledge from the Web.

2.3 Web and Text Mining

Mining Web data has been extensively studied since two decades ago [34] with much work focused on mining text content on the Web [1, 28]. Commonly used text mining techniques include information extraction [45], topic modeling [17], sentiment analysis [69], and neural networks [65]. Our work can be viewed as exploring a new kind of text mining algorithms by leveraging pre-trained large-scale neural language models. While such language models have been widely used for improving text representation in many application tasks, few studies have attempted to use them as a text mining tool. Specifically, such PLMs are trained using massive amounts of text data on the Web that contain knowledge about all kinds of topics. However, due to the non-interpretability of neural LMs, the knowledge is "hidden." Our main idea is to use various prompts to encourage those models to explicitly express the relevant knowledge that we want in the form of generated text, which we can then iteratively evaluate and refine for our text mining needs. As shown in our evaluation results, such a novel strategy of text mining appears to be quite promising and powerful.

3 PROBLEM FORMULATION

We formally define the analogy mining problem as follows.

Given a fixed corpus of data source (which can be as big as the entire web-scale data), for any given target concept *t* expressed as a phrase, the analogy mining problem is to identify a source concept expressed as a phrase *s* that is analogous, or comparable in a meaningful way, to the given target concept *t*, and provides an explanation or justification *Y*, expressed as a set of natural language sentences, on how the two concepts are connected, i.e., what the two concepts' analogical mappings are. For example, in the analogy in table 1, "sifting though a pile of sand" is the source concept for the target concept "candidate generation" and the rest of the text is the explanation. For simplicity, we denote *X* as the input to the analogy mining problem (which includes the target concept *t*), and *Y* as the output (which contains the source concept *s*). In case of *creative* analogy mining, the goal is to discover a creative analogous concept *s* which is not explicitly mentioned in the existing corpus, but the identified analogical explanation *Y* is meaningful.

Note that mining of creative analogies requires some form of inference and is thus significantly more challenging than simply extracting analogies from large amounts of text data. Evaluation of creative analogies also poses special challenges since we cannot possibly enumerate all the creative analogies in advance.

To make the problem more tractable, we must identify specific criteria that define creative analogies so that we can assess the mined analogies based on their overall creativity. Following existing work on assessing creativity [50], we aim to mine analogies that are *typical* (i.e., possess properties of typical analogies), *meaningful* and *novel*. Accordingly, we define the following criteria for assessing creative analogies in this work:

- ***Analogical Style:*** The most basic requirement to assess any analogy is whether it contains an analogy-like text or not.
- ***Meaningfulness:*** A meaningful analogy is defined to be factually correct and reasonable, e.g., containing truthful facts about the source and target concepts and valid analogical mappings.
- ***Novelty:*** A creative analogy must be novel or original when compared to other analogies explicitly mentioned in the corpus.

How to computationally measure these desirable properties is the main technical challenge that we need to address. Below we discuss how we tackle this challenge in the proposed CAM framework.

4 THE CAM FRAMEWORK

Inspired by the existing theories on computational creativity [37], we propose an iterative generation-evaluation framework for mining analogies based on PLMs as shown in Figure 1.

The framework consists of the following three main steps.

- **Step 1 (Generate)**: The PLM with proper prompts is used to generate potential analogies.
- **Step 2 (Evaluate)**: We then apply several scoring functions to automatically assess the quality of the generated text responses in terms of the three criteria of creative analogies 3. Specifically, we design scoring functions to assess the responses generated in Step 1, in terms of the analogical style (*g*), meaningfulness (*r*), and novelty (*n*), and identify the high quality analogies. The output of this step gives us the mined creative analogies.
- **Step 3 (Refine)**: The low quality responses from Step 2 are fed back into the PLM in another round of prompt-based generation to potentially elicit more creative analogies. The second round of prompts could be either manually designed or learned based on feedback from scorers (for example, by using reinforcement learning [14]). This step can be iteratively applied until a stopping criterion is reached, such as the automatic scores assigned to the text responses in the following rounds do not improve.

Thus, our framework is more general than the existing work on generating analogies using PLMs [7], which only covers Step 1.

The three components as discussed above could be flexibly combined to create an end-to-end pipeline for creative analogy mining. For example, the various scorers could be used as sequential filters to create a high-*precision* model that selects the most creative analogies for an application. As shown in Figure 1, one option is to apply a specified threshold θ on analogical style function *g* to filter out the text in non-analogical style. Further, we can linearly combine the meaningfulness score function *r* with the novelty function *n* to rank the generated analogies and select the top-ranked ones. As there is generally a trade-off between meaningfulness and novelty, e.g., completely random text would be highly novel but not meaningful, their combination could be optimized based on the application. To improve the *recall* (i.e., discover as many creative analogies as possible), the entire pipeline could be used iteratively.

4.1 Unsupervised Instantiation of CAM

As it is naturally desirable to be able to generate creative analogies even without having access to any human assessment or supervision, which enables analogy generation at large scale, we first propose an unsupervised instantiation of the CAM framework.

4.1.1 Prompting PLMs for Analogy Generation: Given the promise of using prompts for this task[7], we use a few manually designed, zero-shot textual prompts for PLMs to generate analogies. This is a generative task to generate an analogy *Y* given an input *X* (i.e., the prompt) that maximizes the conditional probability, $p(Y|X)$.

4.1.2 Scoring Functions: Standard metrics for evaluating natural language like ROUGE [41], METEOR [5], etc. are not appropriate for evaluating creative analogies because they do not measure the three criteria of creative analogies (section 3). Thus, we design an unsupervised scoring function each for those criteria.

- ***Analogical Style Scorer:*** We define the scoring function $g(Y)$ as the output of a classifier in terms of its classification probability,

  $$g(Y) = p(Analogy|Y), \tag{1}$$

  where $g(Y)$ measures the probability that Y is an analogy-like text. Such a classifier requires examples of analogies and non-analogies for training. Instead of using manual labels, we use zero-shot prompting of pre-trained language models to generate **synthetic** examples of both styles of text. Details of prompts for both types of text generations are shown in Appendix A. We then fine-tune a PLM on these generated examples.

- ***Meaningfulness Scorer:*** Although existing work has studied how to evaluate the factual correctness of generated text, (e.g, for abstractive summarization [22]), it is unclear how to directly use them for assessing correctness of novel analogies. Recently, self-consistency has recently been quite successful in solving reasoning problems (e.g., math problems) by consensus on the final answer tokens generated by pre-trained language models using an ensemble of prompts[39, 66]. Thus, we leverage a **Self-Agreement (SA)**-based function $r(Y)$ to measure the agreement or similarity between Y_i (the analogy of interest) and a set of K variations of analogies Y_k generated for the same target concept t under various model configurations (denoted by subscript k). As a higher consensus among all the variations suggests a higher probability of correctness, given K analogies for a target concept, we estimate the correctness and validity of a generated analogy in a given setting Y_i as follows:

  $$r(Y_i) = \sum_{k \in K, k \neq i} \frac{1}{K-1} sim(Y_i, Y_k) \tag{2}$$

  where *sim* can be any reasonable text similarity function.
  We obtain the variations of an analogy by sampling text for the same target from multiple prompts and temperatures.

- ***Novelty Scorer:*** This function evaluates how original the generated response is when compared to existing explicitly mentioned analogies in the corpus about the same target concept. Specifically, we design a **semantic distance**-based function, $n(Y)$, which defines the minimum cosine distance between Y and the top K analogies (denoted as w_k for $\forall k \in K$) obtained from a background corpus of the same target concept, i.e.,

  $$n(Y) = \min_{k \in K}(1 - sim(Y, w_k)) \tag{3}$$

  In order to retrieve such a background corpus, we design the following search queries to query the Web using search engines:

  - ***Source-specific queries for Web analogy retrieval:*** To precisely retrieve analogies from the Web that are most similar to an existing analogy Y, we construct queries containing both the generated source (as present in Y) and the target concepts, joined by analogy indicating phrases, such as “like”, “similar to”, etc. For example, <*target*> [2] is like <*src*> (Appendix A.2, table 11). When the PLM-generated text Y contains content more than the source concept, we also need a parser to identify the source concept in Y. To that end, we design a one-shot prompt (Appendix A, table 9) and leverage PLMs to do so.

  - ***General queries for Web analogy retrieval:*** The goal here is to cast a wider net and identify some commonly found analogies from the Web of the target concept. These queries contain only the target concept and analogy indicating phrases, e.g., <*target*> is like, (Appendix A.2, table 10).

Finally, based on the designed scoring functions, the Creative Analogy Ranking Score of Y is defined as follows:

$$\alpha * r(Y) + \beta * n(Y), \tag{4}$$

where α and β are interpretable hyperparameters that could be tuned or set based on the application needs.

4.1.3 PLM Self-Editing via Prompt Refinement. We propose a simple **self-editing** strategy for improving the quality of generated responses. Here, the text generated by the PLM in the first iteration, Y, is concatenated with X, either the original prompt or a new prompt explicitly designed to instruct the model to improve a specific aspect of Y based on the feedback from the scorer (e.g., “Write a more novel analogy for <target>”.). Thus, the response generated in the next iteration is conditioned on both the response from the first iteration, and the instruction prompt. This would encourage the PLM to edit and improve their original response.

Although this idea could potentially be used to edit and improve all aspects of a creative analogy, we do a preliminary exploration of refinement based on the feedback from the analogical style scorer only. Specifically, if the generated response is classified as being not analogy-like, it is concatenated with the original prompt and the augmented prompt is used for the next round of generation.

4.2 Supervised Instantiations of Scorers

Given some human annotations (e.g., based on a likert-scale) for analogies based on the three criteria, it is feasible to train a supervised model to automatically score other analogies. Thus, we design the following two types of supervised methods.

- **Absolute rating:** Here, the input is a single analogy and the model is trained to output a discrete rating for an aspect, like novelty, on a scale (e.g., 1-4). Formally, it is defined as follows:

  $$s(Y_i) = p(l|Y), \tag{5}$$

  where $s(Y)$ measures the probability that Y has a rating of l.

- **Ranking by pairwise-comparisons**: Inspired by ranking losses [30], which learn to discriminate between positive and negative examples, we propose **pairwise ranking as a generative task** to achieve a similar goal of discriminating between analogies of varying qualities based on some scale. Instead of using a loss function, the generative formulation allows us to flexibly use black-box models, like GPT-3, that are available to fine-tune for generative tasks only. Specifically, the input consists of two analogies as part of the prompt and the model task is to identify the better (e.g., more meaningful or novel) analogy out of them (i.e., first or second) and generate labels 1 or 2 accordingly. Formally,

  $$s(Y_i) = p(l_i > l_j), \tag{6}$$

  where $s(Y_i)$ measures the probability that Y_i has a rating greater than that of another analogy Y_j on a given scale. In general, there can also be ties, but we ignore that case for simplicity.

5 EXPERIMENTS

In this section, we describe the implementation details, the datasets used, problem setup, and results of our experiments.

We aim to investigate the following main research questions: 1. How effective is the CAM framework in mining creative analogies? 2. How well do our designed scoring functions work for assessing multiple aspects of creative analogies? 3. Does adding supervision from human ratings improve the performance compared to unsupervised scoring functions? 4. Can prompt refinement (self-editing) improve the quality of the generated creative analogies?

5.1 Implementation

In this section, we describe the implementation details of major components of our CAM framework. More details about hyperparameters and training set construction are in Appendix A.3.

5.1.1 PLM generator: For generating analogies of all target concepts, following [7], we use the largest InstructGPT model [48] (text-davinci-0010) [3], i.e., GPT-3 aligned to follow human instructions, provided by OpenAI API[4]. We also used the same hyper-parameters and prompts as theirs, namely, five synonymous prompts zero-shot prompts directly asking the PLMs for analogies (Table 7, Appendix A.2). For example, “Explain <*target*> using an analogy.” By design, these prompts do not explicitly instruct the model to generate creative analogies as we aim to investigate if PLMs can do that even without significant prompt engineering. For ML and cybersecurity, we added the domain after the target concept in parentheses to help contextualize the PLM and avoid any ambiguity.

5.1.2 Scoring Functions: Implementation details of the scorers that assess the generated analogies are described below.

- **Analogical Style Scorer:** For developing the Analogical Style Scorer, we fine-tuned BERT for sequence classification [15] using the HuggingFace library[5] on the synthetic dataset (section 4.1.2). Probability threshold to filter non-analogies was set to 0.5.
- **Similarity score:** For all similarity computations, we leverage one of the state-of-the-art models for computing semantic similarity of longer text like sentences, Sentence-BERT [49].
- **Semantic distance-based Novelty Scorer:** We leverage Microsoft Bing API [6] for search-engine based retrieval and retrieve the snippets of the top 10 results for each query.
- **Creativity Ranking Score:** α and β in Equation 4 are set to 0.5.
- **Supervised Meaningfulness and Novelty Scorers:** Open AI GPT-3 (*curie*) model is fine-tuned using default parameters for all the scorers. We report average results from three different runs. The test set is comprised of analogies of 100 randomly sampled target concepts from each domain, such that there is no overlap between the train and test set target concepts. A single model is trained on the train data from all

the domains, and is tested on each domain separately. We design simple prompts for both the fine-tuning tasks (see Appendix A.2.1).

Motivated by the practical utility of the mined analogies, we identified three domains with abstract concepts that could potentially be explained well using analogies: *Science*, *Cybersecurity*, and *Machine Learning*. These three domains are different because we expect more information, including analogies, about science concepts to be online compared to machine learning and cybersecurity.

For the science domain, we used the same set of 109 target concepts introduced in [7]. These were compiled from analogical questions on Chegg.com [7]. For cybersecurity and machine learning, we did not find any existing resources of concepts with analogies. Instead, we crawled online glossaries[8] [9] [10] to identify target concepts.

5.2 Problem Setup

The problem of evaluating automatically generated creative analogies in specialized domains is itself challenging. This is because (1) all possible correct answers cannot be pre-specified, (2) manual evaluation is subjective and cognitively demanding. Thus, we primarily use human evaluation to assess our framework. Moreover, large-scale expert evaluation is infeasible because it is expensive and time-consuming. Thus, we mainly rely on crowd annotation, as is also standard in other natural language evaluation tasks [52].

Since previous work has shown that the largest InstructGPT model is the best among all the InstructGPT models at generating meaningful analogies [7], we evaluate our overall framework with this model alone and study the effectiveness of the various scoring functions. The following baselines and evaluation metrics are used to compare our models against crowd ratings.

- **Baselines:** We use the following baselines for all experiments.
  - **Majority** assigns the most common value (i.e., mode) of the reference human ratings to every sample.
  - **Random** randomly assigns a rating between 1 – 4 to a sample.
- **Evaluation Metrics:** Following existing work on evaluating the automatic metrics for assessing natural language generations (e.g., [53]), we use Kendall's Tau-b (τ) [31] to measure overall agreement of automatic scores with human ratings. Additionally, we also measure the top-k performance using Normalized Discounted Cumulative Gain (NDCG@k) [26] because our scoring functions are ultimately used to rank and select the most creative analogies.

5.3 Human Evaluation of Unsupervised CAM

To evaluate the unsupervised CAM framework (section 4.1), human evaluation was conducted via Amazon Mechanical Turk.

5.3.1 *Study Design.* We compiled a total of 747 target concepts, 109, 240, and 398 target concepts from the science, cybersecurity, and ML domains, respectively. A total of 22,410 candidate analogies (6 generated samples * 5 prompts per target concept) by prompting InstructGPT in various configurations. We then ran the Analogical Style Scorer on all the candidates to identify likely analogies (Section 5.4.1), which results in a total of 19.4k analogies.

Next, we selected the top-ranked analogies based on the Creative Analogy Ranking score. Overall, a total of 1075 unique analogies (347 from Science, 349 from Machine Learning, and 379 from Cybersecurity), were evaluated by 3 workers each, which is common in previous work on evaluation of automatically generated text [58].

The study had two main questions asking workers to evaluate the generated responses based on the following: 1. Meaningfulness, i.e., whether it is a correct and valid analogy (Likert-scale: 1-4). For simplicity, we asked that non-analogies also be rated low on this dimension. 2. Novelty, i.e., whether similar text can be found online or not. (Likert-scale: 1-4). We used internet searches as a proxy for evaluating novelty because the PLM training corpus was not publicly available but it is a subset of the internet [10]. Additionally, both for worker quality check (i.e., preventing spamming), and for understanding the reasoning behind workers' novelty ratings, workers were asked to submit their search queries that they used to check for novelty, along with the web urls they found (if any).

We conducted initial screening tests to identify high quality workers and sixteen workers were selected to complete the tasks. Please refer to Appendix A.1 for more details of the study design.

5.3.2 *Annotation Data Analysis.* On average, each crowd evaluator submitted 1.4, 2.5 and 2.6 queries in the science, ML, and Cybersecurity domains respectively. Lower number of queries in science domain is expected because it is easier to find science analogies online. Overall, this indicates that the crowd annotators spent reasonable effort on novelty evaluation.

The inter-annotator agreement (Krippendorf's α [35]) are as follows: Meaningfulness - Science = 0.247, ML = 0.136, Cybersecurity = 0.288, Novelty - Science = 0.402, ML = 0.197, Cybersecurity = 0.2. One possible reason for the relatively low inter-annotator agreement may simply be due to the subjective nature of the nature of the task [52], but another possibility could be that the crowd workers were not experts in domains like ML and Cybersecurity. Thus, we conducted a quality check to test whether their ratings could be used reliably for our task. We compared their average ratings [11] in the ML domain against a subject matter "expert" annotated dataset. To create the expert dataset, one author of this paper and another graduate student with research experience in Machine Learning rated the analogies via mutual discussions.

Upon comparing the agreement of crowd ratings against the expert dataset, we find that crowd ratings are better than the baseline strategies mentioned in 5.2. Also, the τ between expert and crowd rating (τ = 0.39 for Meaningfulness and τ = 0.49 for Novelty) indicates moderately strong correlation between crowd and expert ratings[32], suggesting that the quality of the crowd annotations is reasonably good (refer table 6, Appendix A for details).

5.3.3 *Crowd Evaluation Results.* Table 2 shows the overall results of crowd evaluation, in terms of average ratings for Meaningfulness and Novelty, and the counts of mined Novel, Meaningful, and Creative (both Novel and Meaningful) analogies, i.e. analogies with average rating>3.3. We can see that the performance varies by domain. For example, for the Science domain, the least number of novel responses were generated. This is most likely because there are several science analogies online that InstructGPT would have been trained on, and it simply retrieves those. Overall, 13.7%(147/1075) of the selected responses were highly creative analogies. Using a lower threshold (i.e., average rating>2), 56.37% of the analogies were at least somewhat creative. This shows the effectiveness of the unsupervised CAM framework. Next, we investigate the performance of the individual components of the framework.

Table 2: Results of crowd evaluation showing total number of analogies rated, average meaningfulness and novelty ratings, and number of meaningful, novel, and creative analogies.

Domain	Total #	Avg. Mng.	Avg. Nov.	# Mng.	# Nov.	# Ctv.
Science	347	3.18	1.88	228	38	11
Cyber.	379	2.93	2.77	182	132	49
ML	349	3	3.33	156	225	87
Overall	1075	3.03	2.67	566	394	147

5.4 Performance of Scorers

In this section, we investigate the performance of our scorers.

5.4.1 *Analogical Style Scorer.* For this classifier, we rely on automatic evaluation. We observed high performance on the synthetic validation set 4.1.2 ($F1$ = 97%). This is expected because analogies are generally easy to identify due to phrases like "is like", "is similar to" etc. Using the trained classifier, ≈ 88% of all the candidate analogies were classified as being analogy-like, which again confirms that the PLM is reasonably good at this task.

Table 3: Meaningfulness Scorer Performance. SA is the unsupervised Self Agreement scoring function. Sup. Rate is the supervised model fine-tuned to generate absolute ratings. Sup. Rank is the supervised model fine-tuned to generate pairwise rankings.

	Science			Machine Learning			Cybersecurity		
	NDCG@1	NDCG@10	τ	NDCG@1	NDCG@10	τ	NDCG@1	NDCG@10	τ
Majority	.8014 ± .02	.8014 ± .02	-	.7555 ± .00	.7555 ± .00	-	.7387 ± .02	.7387 ± .02	-
Random	.7978 ± .05	.7978 ± .05	.05 ± .05	.7605 ± .01	.7605 ± .01	.05 ± .02	.7508 ± .05	.7508 ± .05	.02 ± .04
SA	.9167 ± .00	.8585 ± .02	.14 ± .03	.75 ± .00	.8269 ± .02	.04 ± .02	.6111 ± .16	.6346 ± .09	− .0 ± .07
Sup. Rate	.871 ± .02	.871 ± .02	.26 ± .07	.8042 ± .01	.804 ± .01	.22 ± .04	.7477 ± .05	.7477 ± .03	.13 ± .04
Sup. Rank	1.0 ± .00	.904 ± .03	.26 ± .07	.8611 ± .14	.8175 ± .04	.17 ± .04	.8055 ± .11	.7926 ± .05	.12 ± .02

5.4.2 *Performance of Meaningfulness Scorers.* Table 3 shows the average performance of our meaningfulness scorers evaluated by humans. Below are our main findings.

- **Unsupervised scoring function:**. The performance of self-agreement varies by domain. Compared to the baselines, it mainly achieves better overall performance in the Science domain alone. We suspect this is because science domain has a lot of available online information (including analogies) on which the underlying GPT-3 model has been trained. In this case, consensus helps filter out the noisy responses that were somehow generated under some inference configurations. On the other hand, the model possibly does not have true knowledge about some rare, technical concepts in the ML and Cybersecurity domain. So, most model configurations would generate similar, incorrect text representing the model's “best guess” about the concept. One example in the ML domain is the following analogy generated for *NaN traps* [12]: *“Nan traps can be thought of as small machines… They are able to do this by trapping particles called nanotubes,…”*

- **Further improvement via supervised learning:** The effectiveness of the unsupervised meaningfulness scoring method, especially in the science domain, means that it can be used to mine creative analogies from the Web at large scale. However, we are also interested in knowing whether we may further improve its effectiveness by leveraging human annotations for supervised learning. Our results show that the supervised models are generally better, overall, than the unsupervised method, confirming that supervision is helpful. Thus, we may use human (crowd) annotations to further optimize CAM for specific applications.
- **Supervised ranking vs. supervised rating:** The supervised ranking model generally has best NDCG@k scores compared to all models including the supervised rating model. Thus, our designed generative formulation of pairwise ranking achieves the intended goal of high performance at top-k ranking. The rating model consistently achieves the highest overall τ correlation, which is generally a moderate correlation [32]. These results suggest that the ranking model could be used to select the top most meaningful analogies, while the rating model could be used when the full ranked-list order is valuable, for example, selecting less meaningful analogies to improve via self-editing.

5.4.3 Performance of Novelty Scorers. Table 4 shows the performance of novelty scorers. The performance of the unsupervised, semantic-distance based scoring function is relatively stable across all domains, and generally better, overall, than the baselines. This suggests it could be used as a general and robust way to assess the novelty of generated responses.

Table 4: Novelty Scorer Performance. Sem-dist is the unsupervised scoring function that computes semantic distance to a background corpus. Sup. Rate is the supervised model fine-tuned to generate absolute ratings. Sup. Rank is the supervised model fine-tuned to generate pairwise rankings.

	Science			Machine Learning			Cybersecurity		
	NDCG@1	NDCG@10	τ	NDCG@1	NDCG@10	τ	NDCG@1	NDCG@1	τ
Majority	.4803 ± .01	.5095 ± .00	-	.8288 ± .00	.8288 ± .00	-	.6961 ± .01	.6961 ± .01	-
Random	.7936 ± .00	.7936 ± .00	− .01 ± .05	.7621 ± .00	.7621 ± .00	.05 ± .00	.7186 ± .02	.7186 ± .02	− .02 ± .01
Sem-dist	.7778 ± .08	.75 ± .01	.27 ± .07	.8889 ± .04	.8492 ± .02	.19 ± .05	.8333 ± .11	.7704 ± .02	.17 ± .06
Sup. Rate	.5227 ± .19	.6327 ± .08	.33 ± .07	.8599 ± .02	.8599 ± .02	.18 ± .07	.8567 ± .07	.8274 ± .03	.33 ± .05
Sup. Rank	.5833 ± .14	.6894 ± .02	.26 ± .07	1.0 ± .00	.8929 ± .03	.13 ± .03	.9444 ± .04	.8096 ± .03	.22 ± .03

Table 5: Sample text generated by the PLM before and after prompt refinement.

	Generated Text
Before	Attack surface is the potential entry points an attacker could use to gain access to a system. It can be thought of as the sum total of all the ways an attacker could get into a system. Attack surface can be increased…
After	Attack surface can be thought of as the doors and windows of a house. The more doors and windows a house has, the easier it is for someone to get in. The same is true for a computer system. The more ways an attacker can get into a system, the easier it is for them to gain access.

The overall trends for supervised methods compared to unsupervised methods, and supervised ranking vs. rating are similar to the meaningfulness scorers. The novelty scoring functions generally achieve moderate correlation with human ratings based on τ.

In terms of NDCG@k scores, the science domain seems to be the most challenging as no method was able to beat the random baseline. This could be because it has very low overall novelty. Thus, the models potentially did not have enough samples with very high novelty (i.e, top-ranked samples) to learn from. In future, we could explore the following: 1) since the sem-dist function has a more stable performance due to the precise background knowledge from the web, it could be used as a feature to improve the supervised novelty scoring function; 2) apply the self-editing strategy to generate more novel analogies in this domain.

5.5 Analysis of self-editing strategy

Results from the previous sections, indicate that there is still room for improving the recall or mining more analogies using PLM. We now investigate whether the second round of prompting, augmented with the response from the first round, could help generate more analogy-like text. We found that majority (*86.8%*) of the candidates generated using the second round of prompts were classified as being analogy-like. This indicates that the additional context provided by the response from the first round of generation is a helpful prompt-refinement for this task that could enable the PLM to self-edit. Table 5 shows an example of an analogy where the refined prompt helped improve the response. The text generated in the first round contains no analogy at all but when leveraged to refine the prompt in the second round, leads to the generation of a meaningful analogy. Thus, self-editing via iterative refinement of prompts with PLMs could be further explored in future for this task. Specifically, it would be interesting to explore how to mine more meaningful and novel analogies via self-editing.

6 LIMITATIONS

Our study has the following main limitations: (1) The inter-annotator agreement between crowd annotators is relatively low, although these scores tend to be open to interpretation [58] and low scores are common in subjective tasks [52], highlighting the difficulty in evaluation. As suggested in [58], we've released our annotated datasets online for further investigation by the community. Moreover, our findings that the supervised scorers (trained on subsets of crowd-ratings and evaluated on held-out test sets) are almost always much better than the baselines indicate that the ratings contain useful training signals and are not completely stochastic. (2) In some cases, the performance of our methods is still low. This highlights the difficulty of our new task of creative analogy mining and directly suggests an interesting new research direction.

7 CONCLUSION

We presented a novel Creative Analogy Mining (CAM) framework for mining creative analogies from the Web and evaluated its effectiveness. The proposed framework leverages large, pre-trained language models as text mining tools and prompts such a model to generate analogies based on the knowledge that the model has learned from large text collections during training. We further propose multiple scoring components, including Analogical Style Scoring, Meaningfulness Scoring, and Novelty Scoring, to enable mining of creative analogies. Experiment results using InstructGPT show that CAM is able to infer creative analogies from the Web.

Our work opens up multiple interesting future research directions. First, considering the generality of CAM, its promising results suggest that we can use it to power a general analogy discovery engine to support many interesting new applications, enabling many users (e.g., learners, designers, and writers) to discover useful analogies from the Web. The higher performance obtained by training supervised scorers leveraging user feedback also suggests that a human-machine collaborative system would be even more effective at mining creative analogies suitable for an application.

Second, we also found that the novelty of the PLM-generated analogies was low with high meaningfulness in some domains (e.g., Science), whereas in other domains (e.g., Machine Learning), more novel analogies were generated although they were not all meaningful. Thus, we see the full spectrum of creative artifacts mined from the Web using PLMs as stated in Ventura's hierarchical model of creativity [61], i.e., from Plagiarized to completely Random, most likely impacted by the kind of training data available per domain. Thus, more research is required to achieve “meaningful novelty” [37] in all domains. For example, how to leverage research on hallucination in language models [27] to reduce randomness while not sacrificing combinational creativity is highly interesting.

Third, our work shows the great promise of using PLMs as a novel text mining tool. From this perspective, a PLM can be regarded as first pre-extracting and encoding knowledge from a large corpus, e.g., the Web, and then using the knowledge for discovery tasks via prompting and iteratively selecting the high-ranked artifacts based on scoring functions. Such a novel strategy is generally quite efficient, adaptive to the new Web content, and can be potentially used to perform many creative text mining tasks in the future.

ACKNOWLEDGMENTS

This material is based upon work supported in part by the National Science Foundation under Grant No. 1801652 and by the IBM-Illinois Discovery Accelerate Institute.

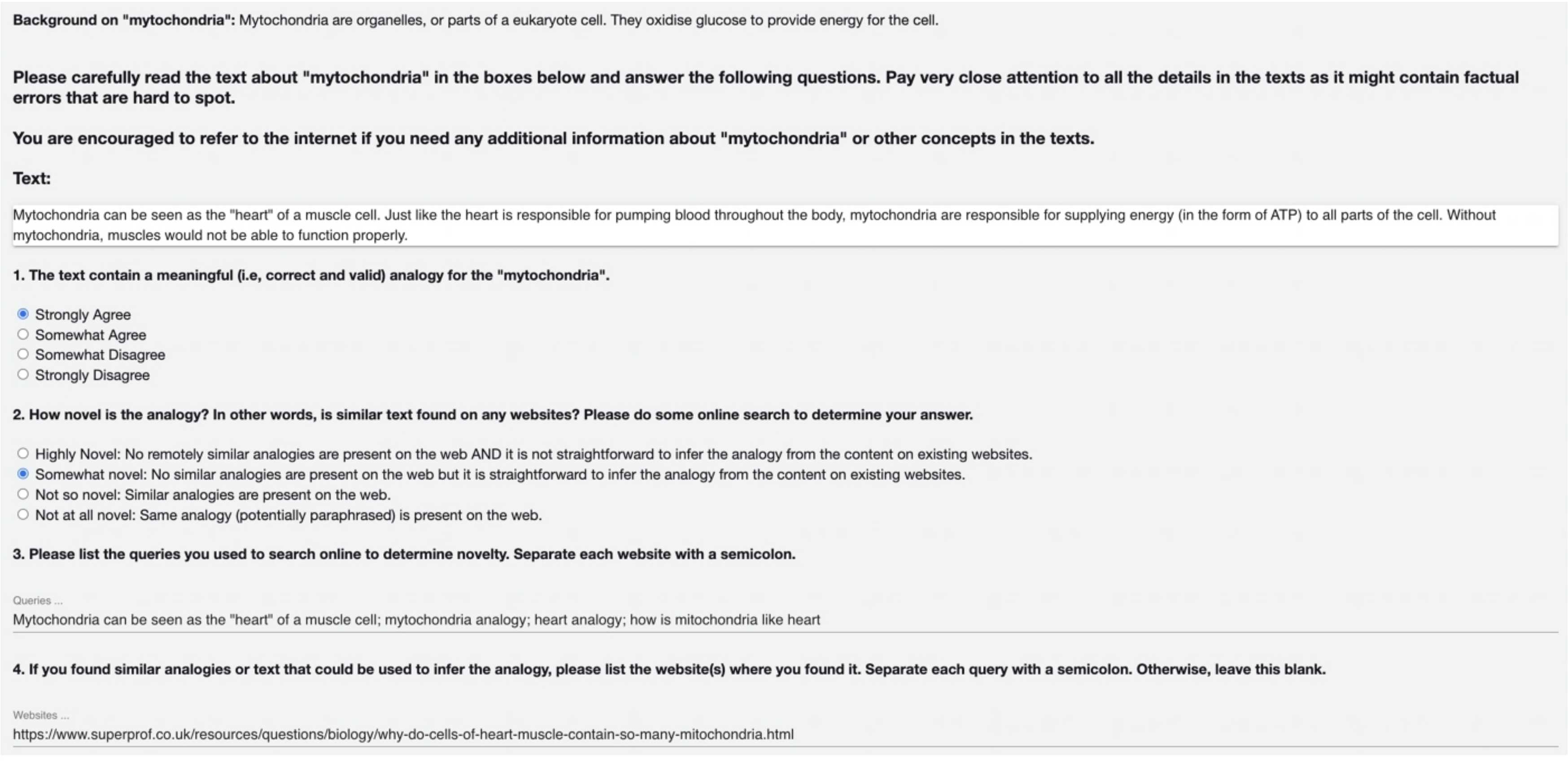
A SUPPLEMENT

Table 6: Crowd performance compared to expert ratings in terms of NDCG@10, Mean Absolute Error, and Kendall's tau.

	Meaningfulness			Novelty		
	NDCG@10	MAE(↓)	τ	NDCG@10	MAE(↓)	τ

Majority	0.59	1.32	-	0.59	2.06	-
Random	0.63	1.28	0.02	0.74	1.2	0.03
Crowd	<u>0.81</u>	<u>0.98</u>	<u>0.49</u>	<u>0.86</u>	<u>0.66</u>	<u>0.39</u>

A.1 Amazon Mechanical Turk



Background on "mytochondria": Mytochondria are organelles, or parts of a eukaryote cell. They oxidise glucose to provide energy for the cell.

Please carefully read the text about "mytochondria" in the boxes below and answer the following questions. Pay very close attention to all the details in the texts as it might contain factual errors that are hard to spot.

You are encouraged to refer to the internet if you need any additional information about "mytochondria" or other concepts in the texts.

Text:

Mytochondria can be seen as the "heart" of a muscle cell. Just like the heart is responsible for pumping blood throughout the body, mytochondria are responsible for supplying energy (in the form of ATP) to all parts of the cell. Without mytochondria, muscles would not be able to function properly.

1. The text contain a meaningful (i.e, correct and valid) analogy for the "mytochondria".

● Strongly Agree
○ Somewhat Agree
○ Somewhat Disagree
○ Strongly Disagree

2. How novel is the analogy? In other words, is similar text found on any websites? Please do some online search to determine your answer.

○ Highly Novel: No remotely similar analogies are present on the web AND it is not straightforward to infer the analogy from the content on existing websites.
● Somewhat novel: No similar analogies are present on the web but it is straightforward to infer the analogy from the content on existing websites.
○ Not so novel: Similar analogies are present on the web.
○ Not at all novel: Same analogy (potentially paraphrased) is present on the web.

3. Please list the queries you used to search online to determine novelty. Separate each website with a semicolon.

Queries ...
Mytochondria can be seen as the "heart" of a muscle cell; mytochondria analogy; heart analogy; how is mitochondria like heart

4. If you found similar analogies or text that could be used to infer the analogy, please list the website(s) where you found it. Separate each query with a semicolon. Otherwise, leave this blank.

Websites ...
https://www.superprof.co.uk/resources/questions/biology/why-do-cells-of-heart-muscle-contain-so-many-mitochondria.html

Figure 2: Screenshot of Amazon Mechanical Turk evaluation interface.

A.1.1 Annotators. In addition to using a screening test consisting of a single question asking to identify an analogy, we set the following MTurk qualifications :workers should have completed at least 5k tasks with >98% approval rate and be located in the US since the task requires proficiency in english (since Mturk does not provide any good way to identify native english speakers).

A simple definitions of the target from online glossaries was provided to workers as reference and they were encouraged to refer to the internet to learn more about the shown concepts. Several sample annotations were provided as part of the instructions to guide workers. Moreover, we were available to answer clarification questions via a shared chatroom.

Annotators were informed that the data be used for research. We consulted with our university ethics board and found that IRB was not required for this study. Annotators were paid at the rate of $18/hr. The rate was decided based on open discussions with them and is above the minimum wage.

A.1.2 Novelty Rating Criteria. Highly Novel (4) : No remotely similar analogies could be found on the web AND it is not straightforward to infer the analogy from the content on existing websites.

Somewhat novel (3): No similar analogies could be found on the web but it is straightforward to infer the analogy from the content on existing websites.

Not so novel (2): Similar analogies are present on the web

Not at all novel (1): Same analogy (potentially paraphrased) is present on the web.

A.2 Prompt Templates and Queries

Table 7: Prompts for generating analogies.

Id	Prompt
P1	Explain <target> using an analogy.
P2	Create an analogy to explain <target>.
P3	Using an analogy, explain <target>.
P4	What analogy is used to explain <target>?
P5	Use an analogy to explain <target>.

Table 8: Prompt Templates for generating non-analogies for Analogical Style Scorer.

Id	Prompt
P1	Explain *<target>*.
P2	What is *<target>*?
P3	Explain *<target>* in plain language to a second grader.

Table 9: One-shot Prompt Template for Source extraction from the generated analogy. The prompt is inspired by an example prompt provided by the OpenAI API for parsing unstructured text. '''is a stop sequence.

> Table summarizing the following analogies: One way to think of empirical risk minimization is as a process of tuning a machine learning model so that it performs well on the training data. The goal is to find a configuration of the model parameters that leads to the lowest possible error on the training set. This can be thought of as analogous to tuning a car's engine so that it runs as smoothly as possible. | Target | Source |Empirical risk minimization | tuning a car's engine '''*<generated analogy>*| Target | Source | *<target>* |

Table 10: General query templates using Microsoft Bing operators for Web analogy retrieval

Id	Query
Q1	*<domain>* +(*<target>* AND analogy)
Q2	*<domain>* +("*<target>* is like")
Q3	*<domain>* +("*<target>* is similar")
Q4	*<domain>* +("just as *<target>*")
Q5	*<domain>* +("*<target>* can be thought of as")
Q6	*<domain>* +("*<target>* can be compared to")

Table 11: Source-specific query templates using Microsoft Bing operators for Web analogy retrieval

Id	Query
Q1	+(*<target>*) is like +(*<src>*)
Q2	+(*<target>*) is similar to +(*<src>*)
Q3	+(*<target>*) can be thought of as +(*<src>*)
Q4	+(*<target>*) can be compared to +(*<src>*)

A.2.1 Prompts for supervised scoring functions. For meaingfulness and novelty rating, the prompt template used is *<anlgy>\ n<label_type>:*, where label_type is *Meaningfulness* and *Novelty*, respectively. The output (completion) is the rating on a scale of 1-4.

For meaingfulness and novelty ranking, the prompt template is *Option 1:<anlgy1>\ nOption 2:<anlgy2>\ nMore <label_type>:*, where label_type is *Meaningful* and *Novel*, respectively. The output is either 1 or 2, depending on the order of the higher quality analogy in the prompt.

A.3 Implementation details

A.3.1 Analogical Style Scorer. The model was fine-tuned for one epoch with weight_decay=0.01, batch size=4, and default values for all other hyperparameters.

The training dataset had a total of 1.9k generated non-analogies for concepts from one domain (Science) and we randomly sampled the same number of analogies generated using prompts. We randomly split the dataest into 80%-20% for training and validation respectively and report accuracy on validation set from 3 different runs. Finally, we fine-tuned BERT classification model on the full training dataset for 500 steps only (to prevent overfitting).

A.3.2 Source Extraction from Generated Analogies. For source extraction from the generated analogies, we used the GPT-3 *text-davinci-001* model with temperature=0 to avoid any random generations from outside the analogy text. Number of maximum tokens is set to 50 as source concept is expected to be a short phrase. If multiple mappings of source and target sub-concepts are generated (e.g., in case of complex analogies with multiple mappings and sub-parts), we only use the first extracted source corresponding to the main target concept. If no source is extracted, we by-pass the source-specific queries and use the entire generated analogy as a query.

A.3.3 Training Data Construction for Supervised Ranking and Rating. To construct pairs for the ranking task, for each generated analogy, we randomly sample two analogies having a large margin (=1) between their ratings. Additionally, to prevent any bias due to the order of analogies in the input, we also create additional data samples with the same input pair in the reversed order and output label reversed. Overall, there are 3.2 k pairs, on average, in the training set for supervised meaningfulness ranking, and 2k for novelty ranking. The unique analogies in the training set used for the ranking models are used as the training set for the rating models to enable comparison between them. To account for the larger number of samples, the ranking models are fine-tuned for only one epoch, and rating models for the default four epochs.

REFERENCES

[1] Charu C. Aggarwal and ChengXiang Zhai (Eds.). 2012. *Mining Text Data*. Springer. Navigate to

[2] Mostafa A Alksher, Azreen Azman, Razali Yaakob, Rabiah Abdul Kadir, Abdulmajid Mohamed, and Eissa M Alshari. 2016. A review of methods for mining idea from text. In *2016 Third International Conference on Information Retrieval and Knowledge Management (CAMP)*. IEEE, 88–93. Navigate to

[3] Workneh Yilma Ayele. 2020. Adapting CRISP-DM for idea mining: a data mining process for generating ideas using a textual dataset. *International Journal of Advanced Computer Sciences and Applications* 11, 6 (2020), 20–32. Navigate to

[4] Workneh Y Ayele and Gustaf Juell-Skielse. 2021. A Systematic Literature Review about Idea Mining: The Use of Machine-Driven Analytics to Generate Ideas. In *Future of Information and Communication Conference*. Springer, 744–762. Navigate to

[5] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. 65–72. Navigate to

[6] Brendan Bena and Jugal Kalita. 2020. Introducing aspects of creativity in automatic poetry generation. *arXiv preprint arXiv:**2002.02511 (https://arXiv.org/abs/2002.02511)*** (2020). Navigate to

[7] Bhavya Bhavya, Jinjun Xiong, and Chengxiang Zhai. 2022. Analogy Generation by Prompting Large Language Models: A Case Study of InstructGPT. arxiv:**2210.04186 (https://arXiv.org/abs/2210.04186)** [cs.CL] Navigate to

[8] MA Boden. 1994. What is creativity?,[w:] MA Boden (red.), Dimensions of creativity. Navigate to

[9] Margaret A Boden. 2009. Computer models of creativity. *AI Magazine* 30, 3 (2009), 23–23. Navigate to

[10] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, *et al.* 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901. Navigate to

[11] Joel Chan, Joseph Chee Chang, Tom Hope, Dafna Shahaf, and Aniket Kittur. 2018. Solvent: A mixed initiative system for finding analogies between research papers. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–21. Navigate to

[12] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:**2110.14168 (https://arXiv.org/abs/2110.14168)*** (2021). Navigate to

[13] Simon Colton, Geraint A Wiggins, *et al.* 2012. Computational creativity: The final frontier?. In *Ecai*, Vol. 12. Montpelier, 21–26. Navigate to

[14] Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric P Xing, and Zhiting Hu. 2022. RLPrompt: Optimizing Discrete Text Prompts With Reinforcement Learning. *arXiv preprint arXiv:**2205.12548 (https://arXiv.org/abs/2205.12548)*** (2022). Navigate to

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:**1810.04805 (https://arXiv.org/abs/1810.04805)*** (2018). Navigate to

[16] Giulia Di Fede, Davide Rocchesso, Steven P Dow, and Salvatore Andolina. 2022. The Idea Machine: LLM-based Expansion, Rewriting, Combination, and Suggestion of Ideas. In *Creativity and Cognition*. 623–627. Navigate to

[17] Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics* 8 (2020), 439–453. Navigate to

[18] Kenneth D Forbus, Ronald W Ferguson, Andrew Lovett, and Dedre Gentner. 2017. Extending SME to handle large-scale cognitive modeling. *Cognitive Science* 41, 5 (2017), 1152–1201. Navigate to

[19] Giorgio Franceschelli and Mirco Musolesi. 2021. Creativity and machine learning: A survey. *arXiv preprint arXiv:**2104.02726 (https://arXiv.org/abs/2104.02726)*** (2021). Navigate to

[20] Dedre Gentner. 2002. Analogy in scientific discovery: The case of Johannes Kepler. *Model-based reasoning: Science, technology, values* (2002), 21–39. Navigate to

[21] Karni Gilon, Joel Chan, Felicia Y Ng, Hila Liifshitz-Assaf, Aniket Kittur, and Dafna Shahaf. 2018. Analogy mining for specific design needs. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–11. Navigate to

[22] Ben Goodrich, Vinay Rao, Peter J Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 166–175. Navigate to

[23] Maureen E Gray and Keith J Holyoak. 2021. Teaching by analogy: From theory to practice. *Mind, Brain, and Education* 15, 3 (2021), 250–263. Navigate to

[24] Douglas R Hofstadter and Melanie Mitchell. 1994. The Copycat project: A model of mental fluidity and analogy-making. (1994). Navigate to

[25] Tom Hope, Joel Chan, Aniket Kittur, and Dafna Shahaf. 2017. Accelerating innovation through analogy mining. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 235–243. Navigate to

[26] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446. Navigate to

[27] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *Comput. Surveys* (2022). Navigate to

[28] Faustina Johnson and Santosh Kumar Gupta. 2012. Web content mining techniques: a survey. *International journal of computer applications* 47, 11 (2012). Navigate to

[29] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, *et al.* 2021. Highly accurate protein structure prediction with AlphaFold. *Nature* 596, 7873 (2021), 583–589. Navigate to

[30] Mahmut Kaya and Hasan Şakir Bilge. 2019. Deep metric learning: A survey. *Symmetry* 11, 9 (2019), 1066. Navigate to

[31] Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika* 30, 1/2 (1938), 81–93. Navigate to

[32] Harry Khamis. 2008. Measures of association: How to choose?*Journal of Diagnostic Medical Sonography* 24, 3 (2008), 155–162. Navigate to

[33] Hyung-Kwon Ko, Gwanmo Park, Hyeon Jeon, Jaemin Jo, Juho Kim, and Jinwook Seo. 2022. Large-scale Text-to-Image Generation Models for Visual Artists' Creative Works. *arXiv preprint arXiv:**2210.08477 (https://arXiv.org/abs/2210.08477)*** (2022). Navigate to

[34] Raymond Kosala and Hendrik Blockeel. 2000. Web mining research: A survey. *ACM Sigkdd Explorations Newsletter* 2, 1 (2000), 1–15. Navigate to

[35] Klaus Krippendorff. 2011. Computing Krippendorff's alpha-reliability. (2011). Navigate to

[36] P. Kruse, A. Schieber, A. Hilbert, and E. Schoop. 2013. Idea mining–text mining supported knowledge management for innovation purposes. In *AMCIS (2013)*. Navigate to

[37] Carolyn Lamb, Daniel G Brown, and Charles LA Clarke. 2018. Evaluating computational creativity: An interdisciplinary tutorial. *ACM Computing Surveys (CSUR)* 51, 2 (2018), 1–34. Navigate to

[38] Won Sang Lee and So Young Sohn. 2019. Discovering emerging business ideas based on crowdfunded software projects. *Decision Support Systems* 116 (2019), 102–113. Navigate to

[39] Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2022. On the Advance of Making Language Models Better Reasoners. *arXiv preprint arXiv:**2206.02336 (https://arXiv.org/abs/2206.02336)*** (2022). Navigate to

[40] Bill Yuchen Lin, Ziyi Wu, Yichi Yang, Dong-Ho Lee, and Xiang Ren. 2021. Riddlesense: Reasoning about riddle questions featuring linguistic creativity and commonsense knowledge. *arXiv preprint arXiv:**2101.00376 (https://arXiv.org/abs/2101.00376)*** (2021). Navigate to

[41] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81. Navigate to

[42] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:**2107.13586 (https://arXiv.org/abs/2107.13586)*** (2021). Navigate to

[43] Melanie Mitchell. 2021. Abstraction and analogy-making in artificial intelligence. *arXiv preprint arXiv:**2102.10717 (https://arXiv.org/abs/2102.10717)*** (2021). Navigate to

[44] Richard G Morris, Scott H Burton, Paul M Bodily, and Dan Ventura. 2012. Soup Over Bean of Pure Joy: Culinary Ruminations of an Artificial Chef.. In *ICCC*. Citeseer, 119–125. Navigate to

[45] Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. *arXiv preprint arXiv:**1806.05599 (https://arXiv.org/abs/1806.05599)*** (2018). Navigate to

[46] Takaya Ogawa and Yuya Kajikawa. 2017. Generating novel research ideas using computational intelligence: A case study involving fuel cells and ammonia synthesis. *Technological Forecasting and Social Change* 120 (2017), 41–47. Navigate to

[47] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, *et al.* 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:**2203.02155 (https://arXiv.org/abs/2203.02155)*** (2022). Navigate to

[48] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, *et al.* 2022. Training language models to follow instructions with human feedback. (2022). Navigate to

[49] 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. https://arxiv.org/abs/1908.10084 **(https://arxiv.org/abs/1908.10084)** Navigate to

[50] Graeme Ritchie. 2001. Assessing creativity. In Proc. of AISB'01 Symposium. Citeseer. Navigate to

[51] René Rohrbeck. 2014. Trend scanning, scouting and foresight techniques. In *Management of the fuzzy front end of innovation*. Springer, 59–73. Navigate to

[52] Ananya B Sai, Akash Kumar Mohankumar, and Mitesh M Khapra. 2022. A survey of evaluation metrics used for NLG systems. *ACM Computing Surveys (CSUR)* 55, 2 (2022), 1–39. Navigate to

[53] Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. BLEURT: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696 (https://arXiv.org/abs/2004.04696)* (2020). Navigate to

[54] Hanieh Shakeri, Carman Neustaedter, and Steve DiPaola. 2021. Saga: Collaborative storytelling with gpt-3. In *Companion Publication of the 2021 Conference on Computer Supported Cooperative Work and Social Computing*. 163–166. Navigate to

[55] Dirk Thorleuchter, Dirk Van den Poel, and Anita Prinzie. 2010. Mining ideas from textual information. *Expert Systems with Applications* 37, 10 (2010), 7182–7188. Navigate to

[56] Hannu Toivonen and Oskar Gross. 2015. Data mining and machine learning in computational creativity. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 5, 6 (2015), 265–275. Navigate to

[57] 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 (https://arXiv.org/abs/2105.04949)* (2021). Navigate to

[58] Chris van der Lee, Albert Gatt, Emiel van Miltenburg, and Emiel Krahmer. 2021. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech & Language* 67 (2021), 101151. Navigate to

[59] Tony Veale. 2013. Once More, With Feeling! Using Creative Affective Metaphors to Express Information Needs.. In *ICCC*. 16–23. Navigate to

[60] Tony Veale and Yanfen Hao. 2007. Comprehending and generating apt metaphors: a web-driven, case-based approach to figurative language. In *AAAI*, Vol. 2007. 1471–1476. Navigate to

[61] Dan Ventura. 2016. Mere generation: Essential barometer or dated concept. In *Proceedings of the Seventh International Conference on Computational Creativity*. Sony CSL, Paris, 17–24. Navigate to

[62] Graham Wallas. 1926. *The art of thought*. Vol. 10. Harcourt, Brace. Navigate to

[63] Hei-Chia Wang, Tzu-Ting Hsu, and Yunita Sari. 2019. Personal research idea recommendation using research trends and a hierarchical topic model. *Scientometrics* 121, 3 (2019), 1385–1406. Navigate to

[64] Kai Wang. 2019. Towards a taxonomy of idea generation techniques. *Foundations of Management* 11, 1 (2019), 65–80. Navigate to

[65] Ruishuang Wang, Zhao Li, Jian Cao, Tong Chen, and Lei Wang. 2019. Convolutional recurrent neural networks for text classification. In *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–6. Navigate to

[66] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171 (https://arXiv.org/abs/2203.11171)* (2022). Navigate to

[67] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems* 22 (2009). Navigate to

[68] Thomas Winters and Pieter Delobelle. 2021. Survival of the wittiest: Evolving satire with language models. In *Proceedings of the Twelfth International Conference on Computational Creativity*. Association for Computational Creativity (ACC), 82–86. Navigate to

[69] Ashima Yadav and Dinesh Kumar Vishwakarma. 2020. Sentiment analysis using deep learning architectures: a review. *Artificial Intelligence Review* 53, 6 (2020), 4335–4385. Navigate to

[70] Kaiyu Yang, Jia Deng, and Danqi Chen. 2022. Generating Natural Language Proofs with Verifier-Guided Search. *arXiv preprint arXiv:2205.12443 (https://arXiv.org/abs/2205.12443)* (2022). Navigate to

[71] Jieli Zhou, Yuntao Zhou, and Yi Xu. 2018. Analogy Search Engine: Finding Analogies in Cross-Domain Research Papers. *arXiv preprint arXiv:1812.06974 (https://arXiv.org/abs/1812.06974)* (2018). Navigate to

FOOTNOTE

[1] All the data and source code will be released here: https://github.com/Bhaavya/CAM **(https://github.com/Bhaavya/CAM)**

[2] In this paper, any text within <> tags (e.g., <target>) indicate a placeholder for the corresponding object.

[3] https://beta.openai.com/docs/models/gpt-3 **(https://beta.openai.com/docs/models/gpt-3)**

[4] https://beta.openai.com/docs/api-reference/completions **(https://beta.openai.com/docs/api-reference/completions)**

[5] https://huggingface.co/docs/transformers/model_doc/bert **(https://huggingface.co/docs/transformers/model_doc/bert)**

[6] https://www.microsoft.com/en-us/bing/apis/bing-web-search-api **(https://www.microsoft.com/en-us/bing/apis/bing-web-search-api)**

[7] https://chegg.com/ **(https://chegg.com/)**

[8] https://ml-cheatsheet.readthedocs.io/en/latest/glossary.html **(https://ml-cheatsheet.readthedocs.io/en/latest/glossary.html)**

[9] https://developers.google.com/machine-learning/glossary **(https://developers.google.com/machine-learning/glossary)**

[10] https://niccs.cisa.gov/about-niccs/cybersecurity-glossary **(https://niccs.cisa.gov/about-niccs/cybersecurity-glossary)**

[11] We also tried another strategy of optimally aggregating the possibly noisy labels from crowd labelers of varying expertise [67] but found that it performed worse than simple averaging.

[12] NaN trap means when one number in the model becomes a NaN (Not a Number) during training and causes many or all other numbers in the model to eventually become a NaN.

WWW '23, April 30–May 04, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9416-1/23/04...$15.00.
DOI: https://doi.org/10.1145/3543507.3587431 **(https://doi.org/10.1145/3543507.3587431)**