Accelerating Innovation Through Analogy Mining

Tom Hope[1], **Joel Chan**[2], **Aniket Kittur**[2] and **Dafna Shahaf**[1]
[1] The Hebrew University of Jerusalem
[2] Carnegie Mellon University
tom.hope@mail.huji.ac.il, joelchan@umd.edu, nkittur@cs.cmu.edu, dshahaf@cs.huji.ac.il

Abstract

The availability of large idea repositories (e.g., patents) could significantly accelerate innovation and discovery by providing people inspiration from solutions to analogous problems. However, finding useful analogies in these large, messy, real-world repositories remains a persistent challenge for both humans and computers. Previous approaches include costly hand-created databases that do not scale, or machine-learning similarity metrics that struggle to account for structural similarity, which is central to analogy. In this paper[1] we explore the viability and value of learning simple structural representations. Our approach combines crowdsourcing and recurrent neural networks to extract purpose and mechanism vector representations from product descriptions. We demonstrate that these learned vectors allow us to find analogies with higher precision and recall than traditional methods. In an ideation experiment, analogies retrieved by our models significantly increased people's likelihood of generating creative ideas.

1 Introduction

The ability to find useful analogies is critical to driving innovation in a variety of domains. Many important discoveries in science were driven by analogies: for example, an analogy between bacteria and slot machines helped Salvador Luria advance the theory of bacterial mutation. An analogy to a bicycle allowed the Wright brothers to design a steerable aircraft. Whether architecture, design, technology, art, or mathematics, the ability to find and apply patterns from other domains is fundamental to human achievement [Hesse, 1966; Markman and Loewenstein, 2010; Dahl and Moreau, 2002].

The explosion of available online data represents an unprecedented opportunity to find new analogies and accelerate human progress across domains. The US Patent database has full text for more than 9 million patents. Millions of scientific papers and legal cases are searchable on the web. Websites like InnoCentive[2], Quirky[3] and OpenIDEO[4] contain millions of ideas, problems and solutions.

We believe these datasets form a treasure trove of analogies that can accelerate problem solving, innovation and discovery. In a striking recent example, a car mechanic invented a simple device to ease difficult childbirths by drawing an analogy to extracting a cork from a wine bottle, which he discovered in a YouTube video. We imagine a future in which people could search through data based on deep analogical similarity rather than simple keywords; lawyers or legal scholars could find legal precedents sharing similar systems of relations to a contemporary case; and product or service designers could mine myriad potential solutions to their problem.

However, sifting through massive data sources to find analogies poses a serious challenge for both humans and machines. In humans, memory retrieval is highly sensitive to surface similarity, favoring near, within-domain analogs that share object attributes over far, structurally similar analogs that share object relations [Gentner *et al.*, 1985; Holyoak and Thagard, 1996; Gentner *et al.*, 1993; Gick and Holyoak, 1983]. Analogical processing also incurs a heavy cognitive load, taxing working memory when even a few relations are required to be processed [Halford *et al.*, 2005].

Finding analogies is challenging for machines as well, as it is based on having an understanding of the deep relational similarity between two entities that may be very different in terms of surface attributes [Gentner, 1983]. Recent advances in data mining and information retrieval use vector representations in order to calculate similarity measures [Mikolov *et al.*, 2013; Deerwester *et al.*, 1990; Blei *et al.*, 2003]. These approaches excel at detecting *surface similarity*, but are often unable to detect similarity between documents whose word distributions are disparate. The problem is especially acute when the source and target domains are different (e.g., bacterial mutation and slot machines).

In this paper, we are interested in **automatically discovering analogies in large, unstructured data sets**. In particular, we focus on a corpus of **product innovations**. There are two insights behind our approach that we believe may make this problem tractable despite its longstanding status as a "holy grail" in both cognitive science and AI. First, rather than trying to solve the problem of fully structured analogical reasoning, we instead explore the idea that we can use weaker structural representations that can be learned and reasoned with at scale (in other words, there is a tradeoff between the ease of extraction of a structure and its expressivity). Specifically, we investigate the weaker structural representation of an idea's

[1] See full version [Hope *et al.*, 2017] [2] innocentive.com [3] quirky.com [4] OpenIDEO.com

purpose and *mechanism* as a way to find useful analogies. The second insight is that advances in crowdsourcing have made it possible to harvest rich signals of analogical structure that can help machine learning models learn in ways that would not be possible with existing datasets alone.

Our approach uses the behavioral traces of crowd workers searching for analogies and identifying the purpose and mechanisms of ideas, then developing machine learning models and similarity metrics suited for analogy mining. We demonstrate that our methods allow us to find analogies with higher precision and recall than traditional information-retrieval methods. In a user study, we show that we can generate far analogies that "inspire" participants to generate more innovative ideas than alternative baselines, increasing the relative proportion of positively-rated ideas by at least 25%.

2 Learning a Representation for Analogies

2.1 Motivation

Much work in computation analogy has focused on fully structured data, often with logic-based representations. For example [Falkenhainer *et al.*, 1989],

```
CAUSE(GREATER-THAN[TEMPERATURE(coffee),
      TEMPERATURE (ice-cube)],
      FLOW(coffee, ice-cube, heat, bar))
```

These representations, while very expressive, are notoriously difficult to obtain. In this section, our goal is to come up with a weaker representation that can be *learned*, while still being expressive enough to allow analogical mining.

Analogies between product ideas are intricately related to their *purpose* and *mechanism*. Informally, a product's purpose is "what it does, what it is used for", and a product's mechanism is "how it does it, how it works". The importance of purpose and mechanism is theoretically rooted in early cognitive psychology work on schema induction, which define the core components of a schema as a goal and proposed solution to it [Gick and Holyoak, 1983]. More recently, the practical value of defining a problem schema as a purpose and mechanism has been demonstrated to have empirical benefits in finding and using analogies to augment idea generation [Yu *et al.*, 2016b; 2016a; 2014b; 2014a].

Separating an idea into purpose and mechanisms enables core analogical innovation processes. Assume (for the moment) that we have for each product i two vectors, $\mathbf{p}_i$ and $\mathbf{m}_i$, representing the product's purpose and mechanism, respectively. Using this representation, we can apply rich queries to our corpus of products, such as:

Same purpose, different mechanism. Given the corpus of all products $\mathcal{P}$, a product i with (normalized) purpose and mechanism vectors $\mathbf{p}_i, \mathbf{m}_i$, and distance metrics $d_p(\cdot,\cdot), d_m(\cdot,\cdot)$ between purpose and mechanism vectors (respectively), solve:

$$\text{argmin}_{\tilde{\mathbf{i}}}\, d_p(\mathbf{p}_i, \mathbf{p}_{\tilde{i}}) \quad s.t.\ d_m(\mathbf{m}_i, \mathbf{m}_{\tilde{i}}) \geq \text{threshold}, \qquad (1)$$

Same Mechanism, different purpose (re-purposing). Solve:

$$\text{argmin}_{\tilde{\mathbf{i}}}\, d_m(\mathbf{m}_i, \mathbf{m}_{\tilde{i}}) \quad s.t.\ d_p(\mathbf{p}_i, \mathbf{p}_{\tilde{i}}) \geq \text{threshold} \qquad (2)$$

Importantly, our dataset of product descriptions contains noisy texts, often written informally by non-professional people. In these texts product descriptions are often lacking detail or are ill-defined. To automatically describe a product in terms of a richer, formal functional model such as in engineering research [Hirtz *et al.*, 2002; Ookubo *et al.*, 2007] would require an inordinate amount of meticulous data annotation and collection by professional engineers.

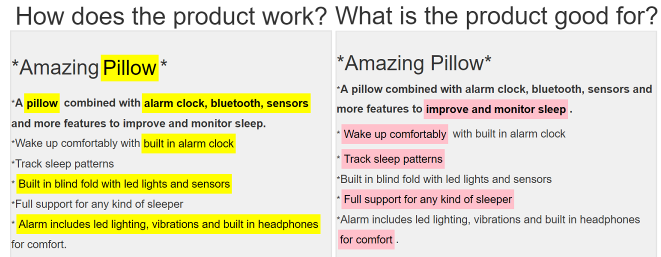

Figure 1: Collecting purpose, mechanism annotations.

2.2 Data

Innovation Corpus. We test our approach with a corpus of product descriptions from Quirky.com, an online crowd-sourced product innovation website. Quirky is representative of the kinds of datasets we are interested in because it is large (at the time of writing, it hosts upwards of 10,000 product ideas, of which our corpus included 8500), unstructured (ideas are described in natural language), and covers a variety of domains which makes cross-domain analogies possible.

Collecting Purpose and Mechanism Data. We needed to collect analogy-specific data to train our model. We aim to develop a lightweight task that avoids complex structured representations, so we can scale up the collection of annotations through the use of crowdsourcing [Kittur *et al.*, 2008]. We show Amazon Mechanical Turk (AMT) crowd workers a product description, asking them to annotate the parts of the text related to the purposes and mechanisms of the product. We frame the problem in simple terms, guiding workers to look for chunks of text talking about "what the product does, what it is good for" (purposes), and "how it works, what are its components" (mechanisms). See Figure 1.

2.3 Method

Extracting Purpose and Mechanism vectors

In this section, we describe our approach to learning to extract purpose and mechanism product representations. We begin with a set of N training product texts $\mathcal{X}_N = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, where each $\mathbf{x}_i$ is a variable-length sequence of tokens $(x_i^1, x_i^2, \ldots, x_i^T)$. For each document $\mathbf{x}_i$, we collect a set of K purpose annotations and K mechanism annotations, where K is the number of workers who annotate each document. We define the *purpose annotation* to be a binary vector $\tilde{\mathbf{p}}_{i_k} = (\tilde{p}_{i_k}^1, \tilde{p}_{i_k}^2, \ldots, \tilde{p}_{i_k}^T)$ of the same length as $\mathbf{x}_i$, with $\tilde{p}_{i_k}^j = 1$ if token x_i^j is annotated as purpose by annotator k, $\tilde{p}_{i_k}^j = 0$ if not. In the same way, we denote the *mechanism annotation* with $\tilde{\mathbf{m}}_{i_k} = (\tilde{m}_{i_k}^1, \tilde{m}_{i_k}^2, \ldots, \tilde{m}_{i_k}^T)$.

We apply a simple and soft aggregation of the K annotations. In simple terms, we look at all words that were annotated, and take a TF-IDF-weighted average of their GloVe [Pennington *et al.*, 2014] word vectors (pre-trained on Common Crawl web data), resulting in weighted-average vectors

$\mathbf{p}_i \in \mathbb{R}^{300}$ and $\mathbf{m}_i \in \mathbb{R}^{300}$ for purpose and mechanism annotations, respectively. We consider $\mathbf{p}_i, \mathbf{m}_i$ to be *target* vectors we aim to predict for unseen texts. By concatenating all annotations and weighting by TF-IDF, we naturally assign higher impact to words considered important by all annotators with respect to purpose/mechanism.

Learning purpose and mechanism

We now have N training product texts $\mathcal{X}_N = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, and N corresponding target tuples $\mathcal{Y}_N = \{(\mathbf{p}_1, \mathbf{m}_1), (\mathbf{p}_2, \mathbf{m}_2), \ldots, (\mathbf{p}_N, \mathbf{m}_N)\}$. We represent each $\mathbf{x}_i$ with its pre-trained GloVe vectors, $\mathbf{w}_i$. Our goal is to learn a function $f(\mathbf{w}_i)$ that predicts $(\mathbf{p}_i, \mathbf{m}_i)$. To this end, we model $f(\cdot)$ with a Recurrent Neural Network as that takes as input the variable-length sequence $\mathbf{w}_i$, processes it with a bidirectional RNN (BiRNN) to form a shared document representation. Following this layer, more layers are added to extract two separate product and mechanism representations $\hat{\mathbf{p}}_i, \hat{\mathbf{m}}_i$, finally predicting the targets $(\mathbf{p}_i, \mathbf{m}_i)$ in a multiple-output regression setting. We qualitatively examine what our representations learned by finding words in our vocabulary that are close to the purpose/mechanism representations (see [Hope *et al.*, 2017] for details). For example, for a yogurt making machine, we obtain "pump, steel, electric" for mechanism and "food, produce, concentrate" for purpose.

3 Evaluation: Analogies

We begin with evaluating the predicted $\hat{\mathbf{p}}_i, \hat{\mathbf{m}}_i$ in the context of their ability to capture distances that reflect analogies, which is the primary focus of this paper. To do so, we first create a dataset of analogies and non-analogies.

3.1 Collecting Analogies via Crowdsourcing

We crowdsourced analogy finding within a set of about 8000 Quirky products. AMT crowd workers used our search interface to collect analogies for about 200 seed documents. Median completion time for each seed was 7 minutes. Pairs that were tagged as **matches** became positive examples in our analogy dataset. Borrowing from information retrieval, we assume that people read the search results sequentially, and treat the **implicitly rejected** documents (i.e., documents that were not matches, despite appearing before matches) as negatives. To further increase the chance that the document has actually been read, we restrict ourselves to the top-5 results.

Results. We rank all pairs in the test data ($N = 2500$, with training done on about 5500 products) based on their distances, according to various metrics including our own, and measure precision and recall @ K results. Across all levels of K – top-%1, %5, %10, %15, %20, %25 – our approach outperformed the baselines. As an example, our approach is able to obtain precision @ top-%1 of 0.74, while standard approaches such as TF-IDF, average of GloVe word vectors with TF-IDF weighting, LDA and LSA and yield only $0.63, 0.61, 0.43, 0.41$, respectively. See [Hope *et al.*, 2017] for full results. Note that a considerable portion of test product pairs were tagged by workers as analogies despite having only surface similarity, likely due to the strong tendency towards surface features in analogical retrieval [Gentner *et al.*, 1993]. This created mislabeled positive examples that favor the surface-based baselines.

4 Evaluation: Ideation by analogy

Since a major application of the enhanced search and retrieval capabilities of analogy is enhanced creativity, we now evaluate the **usefulness** of our algorithms. We examine the degree to which our model's retrieved output improves people's ability to generate creative ideas, compared to other methods. To do so we use a standard ideation task in which participants redesign an existing product [Ullman, 2002], and are given inspirations to help them – either from our approach, a TF-IDF baseline, or a random baseline. See Figure 2 for an example task given to crowdworkers (here, a cell phone charger case). The middle part shows the top 3 inspirations per condition.

Our assumption is that our approach will help users explore more diverse parts of the design space (that are still relevant). We hypothesize that our approach will lead to better results than the TF-IDF baseline (highly relevant but non-diverse) and the random baseline (highly diverse but low relevance).

4.1 Experiment Design

We recruited 38 AMT workers to redesign an existing product, a common creative task in design firms [Ullman, 2002]. To ensure robustness of effects, the experiment included 12 different "seed" products. Participants were paid $1.5. To maximize statistical power, we utilized a within-subjects design with a single manipulated factor, *inspiration_type*:

- ANALOGY: For each seed product we find 12 inspiration products with similar purposes but far mechanism. We use a combination of clustering by the purpose metric $d_p(\cdot,\cdot)$ and diversification by mechanism metric $d_m(\cdot,\cdot)$ with a MAX-MIN diversification approach [Ravi *et al.*, 1994].
- BASELINE: SURFACE: participants receive product inspirations retrieved using TF-IDF, by finding the top 12 products similar to the seed. This baseline is meant to simulate current search engines.
- BASELINE: RANDOM: participants receive 12 product inspirations randomly sampled from our product corpus.

Participants completed the redesign task under each of the 3 inspiration_type conditions. The order of conditions was counterbalanced to prevent order effects. To ensure unbiased permutations, we used the Fisher-Yates shuffle.

Since prior work has shown that people benefit more from analogies if they receive them after ideation has begun [Tseng *et al.*, 2008], the ideation task proceeded in two phases: 1) generating ideas unassisted for one minute, then 2) receiving 12 inspirations and generating more ideas for 6 minutes. The inspirations were laid out in four pages, 3 inspirations per page, and the users could freely browse them.

Figure 2 provides an overview of the experiment and an excerpt from the data. The task was to redesign a cell phone charger case. The SURFACE baseline retrieves products that are very phone-related. In contrast, our algorithm retrieves diverse results such as a human pulley-powered electricity generator suit. The bottom of the figure shows ideas generated by users in each condition. Interestingly, the user exposed to our approach suggested a case that generates power using movement, potentially inspired by the suit.

4.2 Results

Measures. Following [Reinig *et al.*, 2007], we measured creative output as the rate at which a participant generates good

Figure 2: Overview and excerpts of the ideation experiment. Top: Seed product. Workers were asked to solve the same problem in a different way. Middle: Top 3 inspirations for each of the conditions. Note that the TF-IDF baseline returns results from the same domain, while our method returns a broader range of products. Bottom: Ideas generated by users exposed to the different conditions.

ideas. We recruited five graduate students to judge each idea generated by our participants as good or not. Our definition of "good" follows the standard definition of creativity in the literature as a combination of novelty, quality, and feasibility [Runco and Jaeger, 2012]. Each judge was instructed to judge an idea as good if it satisfied all of the following criteria: 1) it uses a different mechanism/technology than the original product (novelty), 2) it proposes a mechanism/technology that would achieve the same purpose as the original product (quality), and 3) could be implemented using existing technology and does not defy physics (feasibility).

Agreement between the judges was substantial (Fleiss kappa 0.51), lending our measure of creativity acceptable inter-rater reliability. The final measure of whether an idea was good or not was computed by thresholding the number of votes, so that good = 1 if at least k judges rated it as good. We report results for both liberal and strict settings $k = 2, 3$.

Evaluation. In summary, across both liberal and strict settings, our approach was able to generate a considerably large relative positive effect leading to better ideas, both in terms of the absolute number of positively-rated ideas and in terms of proportions. For $k = 2$, the proportion of good ideas in our condition was 46% ($N = 105$). Next was the random baseline with 37% (49), and finally the TF-IDF baseline achieved 30% (N=54). These results are significant by a χ^2 proportion test ($p \leq .01$). For $k = 3$ (majority vote), the proportion of good ideas in our condition was 38% ($N = 118$), the random baseline had 22% (68), and the TF-IDF baseline achieved 21% (N=63), with $p < .01$.

In addition, to model confounding factors, we used a generalized linear mixed model with a fixed effect of inspiration condition, and random effects of participant and seed (to model within-participant and within-seed dependencies between ideas). Here too, our method led to a significantly higher probability for good ideas. For $k = 2$, pr(Good) = 0.71, 95% confidence interval = [0.48, 0.87] in our condition. TF-IDF had pr(Good) = 0.28 [0.16, 0.44], and random had pr(Good) = 0.27 [0.16, 0.41]. For $k = 3$, we had pr(Good) = 0.56, [0.36, 0.75]. TF-IDF had pr(Good) = 0.16 [0.08, 0.27], and random had pr(Good) = 0.14 [0.08, 0.24], B = -1.94, $p < .01$ vs. TF-IDF, and B = -2.05, $p < .01$ vs. random.

5 Discussion and Conclusion

In this paper, we sought to develop a scalable approach to finding analogies in large, messy, real-world datasets. We explored the potential of learning and leveraging a weak structural representation (i.e., purpose and mechanism vectors) for product descriptions. We use crowdsourcing to obtain purpose/mechanism annotations, and use an RNN to learn purpose and mechanism vectors for each product. We demonstrate that these learned vectors allow us to find analogies with higher precision than traditional information-retrieval similarity metrics like TF-IDF, LSA, GloVe and LDA.

Our ideation study further illustrates the effectiveness of our approach: participants had a higher likelihood of generating good ideas for the redesign ideation task when they received inspirations from our method, compared to a traditional (TF-IDF) baseline or random sampling approach. From a psychological perspective, the benefits of our inspirations are likely due to our approach's superior ability to sample diverse yet still structurally similar inspirations, since diversity is a known robust booster for creative ability [Chan and Schunn, 2015]. The TF-IDF approach yielded inspirations likely to be relevant but also homogeneous, while the random approach yields diversity but not relevance.

While moving to a "weak" structural representation based on purpose and mechanism significantly increased the feasibility of analogy-finding, extensions may be necessary to generalize to other domains besides product descriptions. For example, our purpose and mechanism vectors did not distinguish between higher and lower level purposes/mechanisms, or core/peripheral purposes/ mechanisms, and also did not encode dependencies between particular purposes/mechanisms. These are potentially fruitful areas for future work.

Acknowledgments

This work was supported by NSF grants CHS-1526665, IIS-1149797, IIS-1217559, Carnegie Mellon's Web2020 initiative, Bosch, Google, ISF grant 1764/15 and Alon grant. Dafna Shahaf is a Harry&Abe Sherman assistant professor.

References

[Blei *et al.*, 2003] David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, pages 993–1022, 2003.

[Chan and Schunn, 2015] Joel Chan and Christian D. Schunn. The importance of iteration in creative conceptual combination. *Cognition*, 145:104–115, 2015.

[Dahl and Moreau, 2002] Darren W Dahl and Page Moreau. The influence and value of analogical thinking during new product ideation. *Journal of Marketing Research*, 39(1), 2002.

[Deerwester *et al.*, 1990] Scott Deerwester, Susan T. Dumais, Geroge W. Furnas, and Thomas K. Landauer. Indexing by Latent Semantic Analysis. *JASIST*, 41(6):1990, 1990.

[Falkenhainer *et al.*, 1989] Brian Falkenhainer, Kenneth D. Forbus, and Dedre Gentner. The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1), 1989.

[Gentner *et al.*, 1985] Dedre Gentner, Russell Landers, et al. Analogical reminding: A good match is hard to find. In *Proceedings of the international conference on systems, man and cybernetics*, pages 607–613, 1985.

[Gentner *et al.*, 1993] Dedre Gentner, Mary Jo Rattermann, and Kenneth D Forbus. The roles of similarity in transfer: Separating retrievability from inferential soundness. *Cognitive psychology*, 25(4):524–575, 1993.

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

[Gick and Holyoak, 1983] M. L. Gick and K. J. Holyoak. Schema induction and analogical transfer. *Cognitive Psychology*, 15(1):1–38, 1983.

[Halford *et al.*, 2005] Graeme S Halford, Rosemary Baker, Julie E McCredden, and John D Bain. How many variables can humans process? *Psychological science*, 16(1), 2005.

[Hesse, 1966] Mary B Hesse. Models and analogies in science. 1966.

[Hirtz *et al.*, 2002] Julie Hirtz, Robert B Stone, Daniel A McAdams, Simon Szykman, and Kristin L Wood. A functional basis for engineering design: reconciling and evolving previous efforts. *Research in engineering Design*, 13(2):65–82, 2002.

[Holyoak and Thagard, 1996] Keith James Holyoak and Paul Thagard. *Mental leaps: Analogy in creative thought*. MIT press, 1996.

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

[Kittur *et al.*, 2008] Aniket Kittur, Ed H. Chi, and Bongwon Suh. Crowdsourcing User Studies with Mechanical Turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 453–456, New York, NY, USA, 2008. ACM.

[Markman and Loewenstein, 2010] Arthur B Markman and Jeffrey Loewenstein. Structural comparison and consumer choice. *Journal of Consumer Psychology*, 20(2):126–137, 2010.

[Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*, January 2013. arXiv: 1301.3781.

[Ookubo *et al.*, 2007] Masanori Ookubo, Yusuke Koji, Munehiko Sasajima, Yoshinobu Kitamura, and Riichiro Mizoguchi. Towards interoperability between functional taxonomies using an ontology-based mapping. In *Proc. of ICED 2007*, 2007.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43, 2014.

[Ravi *et al.*, 1994] Sekharipuram S Ravi, Daniel J Rosenkrantz, and Giri Kumar Tayi. Heuristic and special case algorithms for dispersion problems. *Operations Research*, 42(2):299–310, 1994.

[Reinig *et al.*, 2007] Bruce A. Reinig, Robert O. Briggs, and Jay F. Nunamaker. On the Measurement of Ideation Quality. *Journal of Management Information Systems*, 23(4):143–161, May 2007.

[Runco and Jaeger, 2012] Mark A. Runco and Garrett J. Jaeger. The Standard Definition of Creativity. *Creativity Research Journal*, 24(1):92–96, January 2012.

[Tseng *et al.*, 2008] Ian Tseng, Jarrod Moss, Jonathan Cagan, and Kenneth Kotovsky. The role of timing and analogical similarity in the stimulation of idea generation in design. *Design Studies*, 29(3):203–221, 2008.

[Ullman, 2002] D. Ullman. *The Mechanical Design Process*. New York, NY, 3 edition, 2002.

[Yu *et al.*, 2014a] L Yu, B Kraut, and A Kittur. Distributed analogical idea generation: innovating with crowds. In *CHI'14*, 2014.

[Yu *et al.*, 2014b] Lixiu Yu, Aniket Kittur, and Robert E Kraut. Searching for analogical ideas with crowds. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, 2014.

[Yu *et al.*, 2016a] Lixiu Yu, Aniket Kittur, and Robert E Kraut. Encouraging "outside-the-box" thinking in crowd innovation through identifying domains of expertise. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pages 1214–1222. ACM, 2016.

[Yu *et al.*, 2016b] Lixiu Yu, Robert E Kraut, and Aniket Kittur. Distributed analogical idea generation with multiple constraints. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pages 1236–1245. ACM, 2016.