SymbolFinder: Brainstorming diverse symbols using local semantic networks

ABSTRACT

Visual symbols are the building blocks for visual communication. Symbols communicate information in a way that is quickly intelligible, memorable, and attention-grabbing. Most importantly, visual symbols can convey abstract concepts like *endangered* and *melt* with concrete objects like a panda bear or ice cream. In a formative study, we show that when novices search for symbols, they struggle to brainstorm diverse symbols because they (1) have to recall associations instead of recognizing them and (2) fixate on these few associations instead of exploring many different related contexts. In this paper we present SymbolFinder, an interactive tool for finding visual symbols for abstract concepts. SymbolFinder molds symbol finding into a recognition rather than recall task by enabling users to explore concrete words associated with an abstract concept. To encourage users to explore a diverse set of symbols, we cluster the set of concrete associations through local semantic networks, enabling breadth-first brainstorming. We evaluate SymbolFinder with two comparative user studies: the first validating the clustering design choice and the second demonstrating that SymbolFinder helps novices find more symbols for abstract concepts with significantly less effort than a popular image database.

INTRODUCTION

Visual symbols play a vital role in daily communication. They are placed on signs to help us locate services in transport hubs and public buildings, like the "lost and found" sign in Figure 1. They are used in human-computer interfaces to represent actions a user can take, such as the search icon symbol. They represent organizations and companies like the World Wildlife Fund's logo. Symbols are also combined in images to represent more complicated messages like "Earth is melting" in the public service announcement in Figure 1. Visual symbols are an important part of how we convey information.

While visual symbols come up in a wide range of different contexts, each symbol in Figure 1 share a key similarity: each represents an abstract idea with a concrete object. In the "lost and found" sign, the idea of *items commonly misplaced* is represented by an umbrella and a glove. In the look-up icon, *search* is represented by a magnifying glass. In the WWF logo, *endangered* is represented by a panda bear. Finally, in the public service announcement, *global warming* is represented by the Earth and an ice cream cone. These types of symbols are known as "representational icons," which consist of simple images of familiar objects that represent abstract ideas [28].

UIST'20, October 20–23, 2020, Minneapolis, MN, USA

ISBN 978-1-4503-6708-0/20/04. . . $15.00

DOI: https://doi.org/10.1145/3313831.XXXXXXX

While there has been a great deal of work in the graphic arts and in icon design on how to create representational icons given a representative object [22] [14], the problem of how to find these representative objects in the first place has been relatively overlooked. Visual language is constantly evolving. New symbols are constantly being created to represent new experiences, organizations, and interactions on interfaces [20]. Novices with little to no experience in graphic design are also creating symbols and icons for logos, websites, slide decks, mobile apps and games. Novices have difficulty not only designing icons from concrete objects, but also finding concrete objects to represent the concepts they want to symbolize in the first place.

In order to understand the challenges and workflow of novice designers, we conducted a formative study, where novice participants used Google Images, a popular image search engine, to find symbols for abstract concepts. We observed that novices relied almost exclusively on recalling their own associations about the concept to search for related images. They often had difficulty brainstorming many and different related words, and ended up fixating on a narrow set of associations, which represented a limited aspect of the concept being symbolized. In other words, novices needed help to explore diverse ideas, which is crucial to finding an effective and creative solution [31] [39]. Finally, novices struggled to convert abstract associations into concrete objects and actions that could visually represent the concept.

Inspired by these observations, we created SymbolFinder to help novices to find compelling visual symbols. SymbolFinder helps users brainstorm associations by providing related words from an expansive word association data set. By clustering the related words into groups each of which represents a related but distinct aspect of the concept, SymbolFinder also encourages users to explore a broad range of related contexts, rather than fixating on a narrow set of associations. To create these clusters, SymbolFinder constructs a semantic network of word associations and detects highly connected communities of words. Finally, SymbolFinder helps users find imageable objects and actions by organizing words related to each cluster by word-concreteness.

This paper presents the following contributions:

- SymbolFinder: an interactive interface for finding concrete images to represent abstract concepts

Figure 1. Four visual symbols, from four domains: transportation hubs, human-computer interfaces, logos, public service announcements.

- A technique for applying local semantic networks to help brainstorm distinct, concrete associations for abstract concepts.
- A preliminary study demonstrating that by organizing word-associations into clusters, users had to search less for diverse symbols and felt they were better able to brainstorm.
- A user study demonstrating that SymbolFinder enables users to find significantly more unique symbols with less perceived effort than with Google Images, a popular tool for visualizing concepts.

RELATED WORK

Visual Symbols

Symbols are fundamental in visual communication and are used in a variety of contexts. They accompany headlines in news articles [20], represent actions in computer interfaces [36] [22], guide people in transportation hubs [32], represent corporations in logos [28], and form associations with products in advertisements [24]. There are many advantages in communicating ideas with symbols. Symbols often require less space to encapsulate an idea than using the word itself, saving space in interfaces, maps, and signs [19]. People can more quickly and easily recognize symbols than words because of our innate visual processes [23] [36]. Symbols are more universally understood than words across cultures, which is why they are used and designed for international transportation hubs [32] [27]. Finally, depicting ideas pictorially aids their memorability and recognition [5] [4]. For these reasons, we built SymbolFinder, to help convey more abstract ideas visually.

Brainstorming and Exploration Tools

Many tools have been created to help people brainstorm and explore related ideas. These systems are often designed to present a small set of related words or images to inspire new ideas. To present related textual ideas, *InspirationWall* [1] presents a few related topics from a knowledge graph, *V8Storming* [25] uses word embeddings to find similar words to suggest, and *CrowdBoard* [25] utilizes a real-time crowd to suggest more personalized ideas. Other tools like *Idea expander* [38] and *IdeaWall* [35] present a few related images based on the current spoken ideas of its users. Displaying closely related words and images is very helpful for finding symbols, but to find diverse symbols, it is also necessary that these words encapsulate different relationships with the concept. SymbolFinder organizes a large set of related words into unique ideas, so that users can explore diverse contexts and images for an abstract concept.

Clustering is a popular method used to help users understand and explore large datasets. *Scatter/Gather* enables user to interactively choose clusters to find and explore specific documents in a large collection [10]. *Exploratory Lab Assistant* presents clusters of documents to users as a preliminary step to help them label groups of documents themselves [15]. *Recipescape* clusters recipes for a dish based on the structure of its preparation, enabling users to find recipes with similar or different steps [8]. SymbolFinder clusters word associations to present users diverse ideas related to the concept being symbolized. Word association data sets are often analyzed as networks, where words are nodes and edges represent associations between them [13] [12]. In this format, they are referred to as semantic networks. We construct a "local" semantic network, consisting of words near the concept being symbolized, and cluster it using a popular network clustering algorithm [3].

Interactive image search

There are many interactive tools that help users find specific images. Some tools have users explicitly define their own rules or specify certain image characteristics like color to make image results more relevant [17] [16] [40]. Other systems, like *MayAI*, incorporate a model that updates and presents images according to user feedback [26]. SymbolFinder does not use rules specified by users or feedback so as not to limit a user's brainstorm, but instead presents clusters of unique associations that users can explore to broaden their idea of the concept. Other systems also use knowledge graphs to present diverse visualizations of concepts [21] [37]. *CIDER* uses a knowledge graph to expand queries to return images that capture the possible meanings of an ambiguous search term. For example, "jaguar" would return images of both the animal and the car. Instead of visualizing specific terms, SymbolFinder presents broader clusters of associations to help users find diverse visualizations of abstract search terms, like "control".

BACKGROUND: WHAT MAKES A GOOD SYMBOL?

According to the theory of symbols, there are three basic types of symbols: abstract, directly representational, and indirectly representational [29] (Figure 3). A symbol is abstract when an abstract pattern represents the idea, like the radioactive symbol. A symbol is directly representational when its content is an exact representation of its idea, like the telephone symbol in Figure 3. A symbol is indirectly representational when the image content is associated with but not an exact representation of the idea, like the coat hanger, which represents a *coat check* (Figure 3). SymbolFinder was built to help people find indirectly representational symbols for abstract terms that have a variety of meanings and contexts associated with them. These types of symbols do not require a new design like the *radioactive* symbol and are difficult to find with current image databases, unlike directly representational symbols, as these databases do not enable an exploration of various ideas related to the concept.

A representational symbol can contain three things: a single object, a few related objects, or an action [22]. For example, the coat hanger is the most essential object related to a *coat*

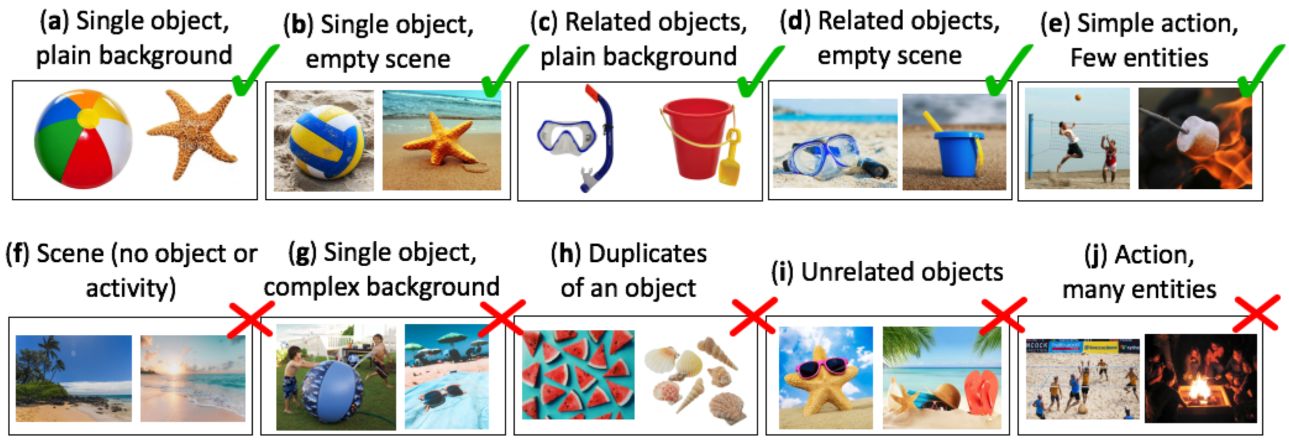

Figure 2. The rules for what makes a good symbol, derived from theory on icons and symbols, and explained with the concept: *summer*. These rules were shown to both raters and participants.

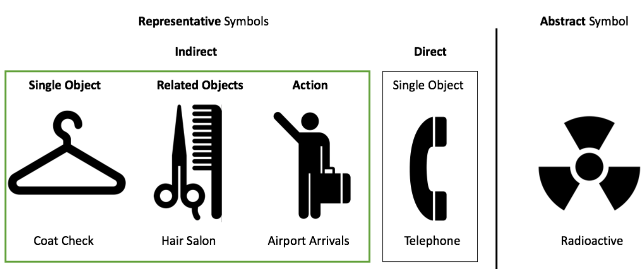

Figure 3. The types of symbols include: representative (indirectly and directly), as well as abstract (radioactive symbol).

check, and thus makes a good single object symbol. Sometimes an extra object makes a symbol more specifically related to the idea it represents. For example, a scissor and a comb together represent a *hair salon* better than either one alone. The two of them together effectively represent the tools a hair stylist uses. Finally, a symbol can also contain an action, like the *airport arrivals* symbol, in which there is a man hailing a taxi. These three categories make up the vast majority of the content displayed in representational symbols.

A good symbol is simple, both in its design and in the content it depicts [14] [22]. The most essential quality of a symbol is that it is recognizable. Its graphic design and content should contain no more than what is necessary to depict the idea. Visual complexity and extra entities only make them slower to interpret and recognize. Therefore, the objects and actions that comprise the content of a symbol should only contain what is necessary to make them recognizable [22]. From this symbol theory, we establish a set of rules to help users of our system find good symbols (Figure 2). A good, indirectly representational symbol can be:

- **A single object**. This object must be able to represent the concept on its own. It can either be in a plain background (Figure 2a) or in an empty, related scene (Figure 2b), but not with extra objects in the background (Figure 2g) as they increase recognition time.
- **Related objects**. These objects should be related and make the symbol more recognizable. They can either be in a plain background (Figure 2c) or empty, related scene (Figure 2d). However, there should not be any duplicate objects, as one is enough to portray the idea (Figure 2h). Also the objects must be related to each other, otherwise they unnecessarily complicate the symbol like the starfish and sunglasses in Figure 2i. These objects could be two separate symbols.
- **An action**. It should be depicted clearly with few entities (Figure 2e). Too many people in the action make it less recognizable, like the full volleyball team in (Figure 2j).
- **No scenes**. Images that are complex containing many people or objects are not good symbols (Figure 2f). There is no plainly displayed object or activity in these images.

FORMATIVE STUDY

To better understand the pitfalls novices face when searching for symbols and how to help them, we conducted a formative study in which we observed participants search for symbols with a popular image database, Google Images. Google Images is often used by professional icon designers to look up visualizations of concepts [41]. Its interface also has a number of features to help users search for particular images, including a list of suggested searches that appear above the image results, each consisting of a related term and a representative image. It also includes the ability to filter images by color and type (clip Art and line drawing), under the "tools" section. Because of its popularity, usage by professional icon designers, and its features, we study how novices use Google Images when searching for symbols.

We recruited 5 participants (3 male, 2 female, average age 24.8). They found symbols for three abstract concepts: *old*, *exciting*, and *innovation*. They were selected randomly from a visual messaging dataset, containing the most common concepts symbolized in online messages [24]. Prior to finding symbols, users were shown a slide-deck detailing the task: to find 20 unique symbols, and shown the good symbol rules (Figure 2). They had 10 minutes to find symbols for each concept. While searching they were asked to think aloud to convey their thought process. After each concept, users were asked to explain the benefits and drawbacks of Google, what search terms helped their brainstorming, and their general strategy.

Participants were frustrated by the lack of conceptual diversity in the images presented when searching the 'concept' as is on Google Images. P1 and P2 both mentioned that the results for *old* predominantly contained images of old people. Similarly, upon seeing the image results for *excited*, P1 states, "These are all images of the word 'excited'. Or just people looking excited." While there was generally a couple representations of the concept in the first set of images produced by Google, users found that they needed to brainstorm on their own to find different symbols.

The most common strategy to find different images was to search terms related to the concept and scan the image results for new visualizations. For example, P1 searched *ancient*, which he recalled on his own, and met many images of the Parthenon, the Colosseum, and pyramids. This turned out to be a fruitful context, from which he was able to collect an additional three symbols for *old*. Similarly, when seeing only images of excited people in the results for *exciting*, P2 subsequently searched *fun* and *adventure*. In doing so, he found other contexts related to *exciting* like extreme sports. Users had to recall these associations on their own. Therefore, our first design goal for SymbolFinder was to **help users brainstorm related words**, in order to enable recognition over recall.

Users however also struggled to find related words that presented different images and concrete contexts related to the concept. For example, when searching for symbols of *exciting*, P2 searched for images of *adventure* and *explore* and was met with similar images of hiking and camping. While he was able to collect a number of symbols from these searches, it was difficult for him to think of another related word that encapsulated a different flavor of *exciting*. Eventually, he searched the word *suspenseful* and found images of horror movies and theatre which inspired more symbols. From this issue we formed our second design goal: when helping users brainstorm associations, we should ensure that we present diverse ideas in order to help them collect **diverse symbols**.

Once users found a fruitful context, their strategy shifted to searching concrete objects and actions that they would select as their symbols. For example, while searching for symbols of *innovation* P2 started searching for advanced technology like virtual reality goggles and hovercrafts. Similarly for *old*, P1 searched for objects old people use like canes and wheelchairs. While more abstract searches like *elder* and *technology* served as inspiration, these highly concrete searches contained the images that would end up being their symbols. When exploring related contexts, users should be able to explore concrete words within these contexts to find representative objects and actions. Thus, our third design goal was to help users **concretize abstract concepts**.

Design Goals. In summary, from the formative study we formed three design goals for the SymbolFinder:

D1: Help brainstorm related words to encourage recognition over recall. Users often recalled related terms to see new visualizations of the concept. By relying on their own memory, they miss obvious symbols and contexts associated with the concept.

D2: Symbol diversity. When helping users brainstorm related terms we should present them a variety of diverse associations so that they can collect diverse symbols from these associations.

D3: Concretize abstract concepts. As well as enabling users to explore diverse associations, they should also be able to explore related concrete terms for each association. This way, users can better find objects and actions to represent the concept.

SYMBOLFINDER

To address these design goals, we present SymbolFinder, an interactive tool that enables novices to find symbols for abstract concepts, by facilitating an exploration of diverse contexts associated with a concept. The SymbolFinder interface consists of two phases. Phase 1 is a breadth-first exploration of clusters of associations related to the concept; users select clusters they would like to explore further (Figure 5). In phase 2, users select symbols while exploring words and images associated with the clusters they selected in phase 1 (Figure 4).

Phase 1: Breadth-first concept exploration

Phase 1 addresses D1 (help brainstorm related words) and D2 (symbol diversity). To help users brainstorm associations, we enable users to explore many words related to the concept. To encourage symbol diversity, similar related words are clustered together into unique associations, where users can select diverse symbols from these different contexts. Each cluster is represented by its three most important words and encapsulates a unique association of the concept: two clusters for *control* are "rule, government, governance" and "handle, lever, knob" (Figure 5). Users scroll through the clusters and select clusters to explore further. For each cluster, the user is posed the following question: "Could symbols of [word 1], [word 2], [word 3] represent [concept]?". The user is instructed to answer this question based on the words. If they think there are potential symbols for these words that could represent the concept, then they press "yes". Below this question are also 5 images related to these words (Figure 5b). Users have the option to select an image if they think it is a good symbol. These images come from three Google Image searches, one for each of the words, where each query is formulated as follows: "[concept] [word]". This is done to keep the results relevant to the concept. The queries for top cluster in figure 5 were: "control rule", "control government", and "control governance". By having users explore each cluster briefly, we quickly expose them to a number of different associations, preventing fixation on any single association.

Phase 2: Image selection within clusters

Phase 2 further supports D1 (help brainstorm related words) as well as D3 (concretize abstract concepts). In phase 2, users further explore the clusters they selected from phase 1 (Figure

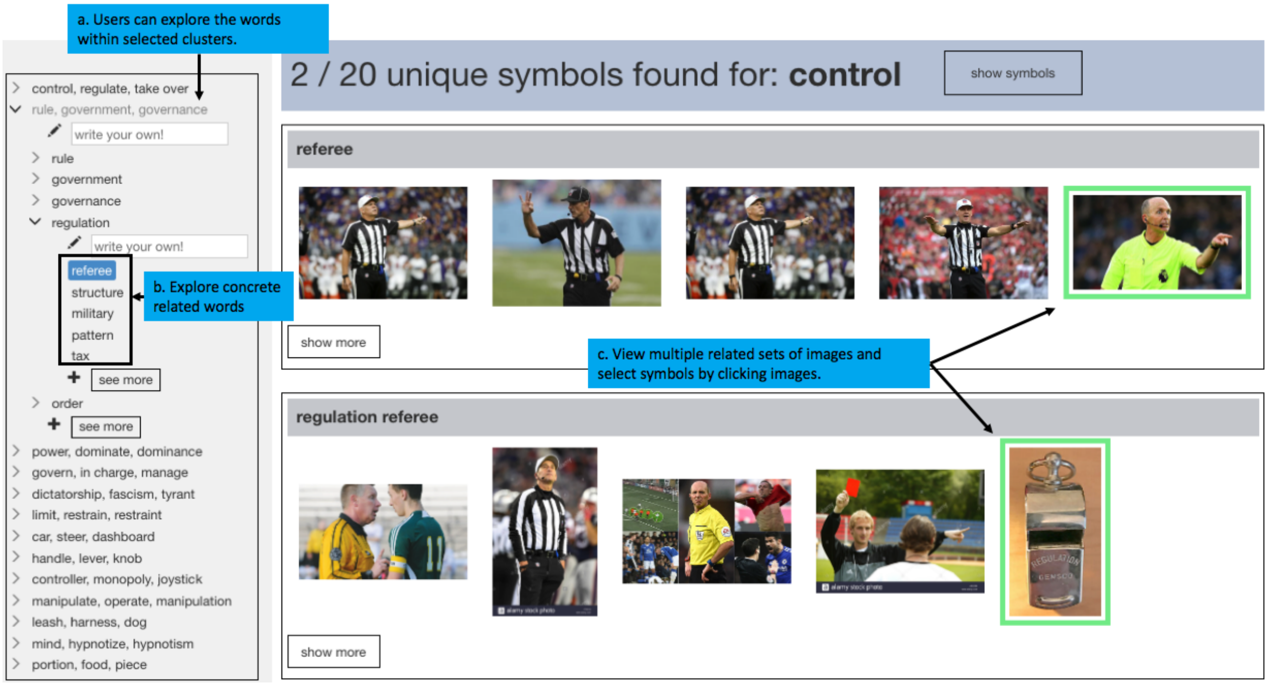

Figure 4. Phase 2: Users explore the clusters they chose from the first phase. On the left sidebar are the clusters, where users can explore related words. On the right are a few Google Image searches for the selected word, which here is "referee". The user selected two images, indicated by the green boxes.

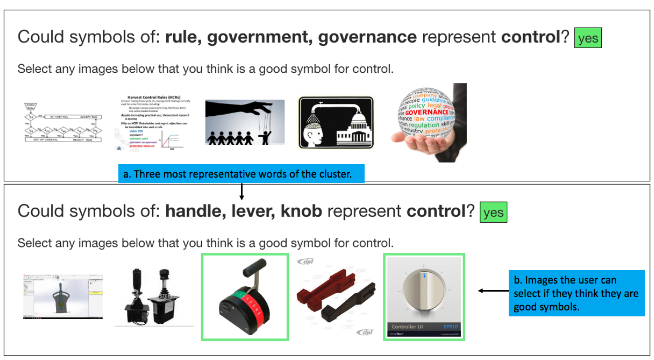

Figure 5. An example of phase 1 for the concept: *control*. Users select relevant clusters they would like to explore further in phase 2. Each cluster conveys a different association related to *control*, like the government (top cluster) or physical tools we use to control machines. (bottom cluster).

4a) and select symbols (Figure 4c). The key part of this interface is the sidebar on the left which is where users explore the clusters (Figure 4a) and recursively explore concrete words related to them (Figure 4b). To support D3, when users select the top level cluster words, they are shown related words sorted by concreteness (Figure 4b). In Figure 4, the user selected the "rule, government, governance" cluster. They then expanded *regulation*, one of the cluster words, and selected *referee*, a related concrete term. As well as exploring the clusters, users can also type associations they think of themselves in the "write your own" text boxes and view images related to their entry. In this way, the sidebar enables users to recognize good symbols as well as use their own thought processes.

The second key part of this interface is the set of Google Image search results that populate the screen when a word is selected (Figure 4c). Four queries are made per word, and they include the word on its own [referee], the word and its parent [regulation referee], the concept and the word [control referee], and finally the word and "icon" appended to the search [referee icon]. We include the parent and concept queries as they help keep images on topic. We include the icon query as they often provide simple images of the action or item we are looking for. Together, the sidebar of clusters and the multiple image searches effectively help users to find concrete symbols.

IMPLEMENTATION

D1: Helping users brainstorm related words

In the formative study, participants searched words related to the concept in order to find new images of symbols. This was a good strategy, but novices had difficulty recalling related words on their own, and missed many useful words. Therefore our first design goal was to help users brainstorm related words by making it a recognition rather than a recall task. First, in phase 1, SymbolFinder presents clusters of word associations for users to browse and choose from. Then, in phase 2, users can explore more related words associated with each cluster. We explored two different options for creating word associations: (1) Glove word embedding, trained on Common Crawl [34] and (2) Small World of Words (SWOW), a crowd-sourced word association database [11]. Word embeddings are commonly used for comparing the similarity between words [30] and have been used in a number of brainstorming tools to compare the similarity of ideas [35] [7]. SWOW is a large English word association dataset. The dataset was created by having thousands of participants complete a free word association task, in which each participant records the first three words they think of when seeing a cue word.

In initial testing, we found that SWOW produced words that were more relevant and concrete than those by Glove. For example, for the abstract word *help*, the most related words that Glove produces include words like: *helping* and *need*, which are related, but are not specific or concrete. Meanwhile SWOW produces terms like: *donation*, *red cross*, and *tutor*, giving specific actions, organizations, and people that help. These words are actually helpful for visualizing *help*. At the same time, word embeddings tend to also return antonyms when querying similar words, which was less of a problem with SWOW [33]. For these reasons, we chose SWOW to be our dataset for providing related words.

D2: Diversity using Local Semantic Networks

In the formative study, users struggled to think of related words that represented new associations with the concept they were symbolizing. They explored redundant terms and fixated on only a few related contexts. To support D2 and help users find diverse symbols, we cluster a large set of word associations and present a diverse set of contexts from which users can find symbols.

Word association datasets are often analyzed as networks, where each word in the dataset is a node and each association is an edge. In this format, they are referred to as semantic networks. Researchers analyze these networks to identify important words in languages via network centrality measures as well as highly connected clusters of words [13] [12]. We leverage this technique to find clusters of words close to the concept we are symbolizing. We create a "local" semantic network, consisting of words in the concept's neighborhood. We then run a popular network clustering algorithm on this network to identify sets of highly connected words that represent distinct associations of the concept.

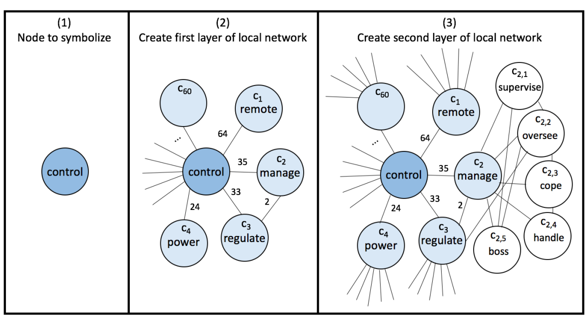

Figure 6. The construction of the local semantic network.

Constructing a local semantic network

Our high-level goal when constructing the local semantic network was to create a network consisting of a variety of words to generate a diverse set of clusters, without making the clusters too large so that they became indecipherable. To ensure a variety of words, we construct a network with two levels of nodes. This network consists of the root concept, its aggregate list of related words (level 1), and the related words of nodes in level 1 (level 2). This is shown in Figure 6, where the root is *control*, the words in level 1 are in light blue, and the words in level 2 are in white. The words of level 2 are still relevant to the root concept but also representative of a more specific association to the root. For example, in Figure 6, from *control* we get *manage* which yields *boss*. While *boss* is more directly associated with *manage*, it is still very related to *control*. Each word in SWOW has a set of associated words, each with a count indicating the number of unique participants who made that association. For every pair of nodes in the network, we create an undirected edge between them if they are associated, where the weight of that edge is the associated count. We do not construct a third level of words from the second as we found this includes many irrelevant nodes in the network, making the clusters less interpretable. By incorporating two levels of words, we ensure the words in the network are related to the concept and diverse.

To ensure that the clusters do not get too large or indecipherable we set two constraints. The first constraint is that we take up to the first c related words for the root concept to include in level 1. We empirically determined 60 to be a suitable value for c; this leads to clusters containing on average 21 words. The size of the clusters will not vary much for small changes in c, but generally when set to 90 or higher, the clusters tend to get large, containing on average 35 words, becoming less interpretable. That being said, the average number of related words per concept is 20.32, with 6.1% of the dataset containing more than 60, so we generally take all of the root concept's related words. The second constraint pertains to the nodes we add to level 2. For each of the c nodes, we take their first g most related words not present in the network. We set g to be 5. We keep g relatively low compared to c because some level 1 nodes, like *remote* in figure 6, introduce words unrelated to the root concept, *control*. Also by setting c to 60 and g to 5, we ensure a maximum number of nodes that does not exceed 360, which keeps the cluster sizes in the next step manageable. While simple, these settings produce a network that is reasonably sized and a set of communities that are coherent.

Clustering the network

Our goal is to present users a list of diverse and distinct ideas related to the concept. The local semantic network we create can contain up to a few hundred words. While containing many different associations, this is too large a space for users to explore without guidance. To condense this information into a manageable set of diverse associations, we cluster the network. Our goal is to create a set of clusters that contain highly related words that capture a distinct association of the concept. To cluster the network we considered two algorithms: the Clauset-Newman-Moore [9] and Louvain [3] network clustering algorithms. Both are very popular hierarchical clustering algorithms. From initial experimentation we determined that the Louvain algorithm produced more interpretable clusters, as the Clauset-Newman-Moore algorithm tended to produce fewer clusters with a greater number of words, often mixing together clusters that the Louvain algorithm separated.

The Louvain algorithm optimizes modularity, a measure which compares the edge density of the nodes in a cluster to the edge density of the same nodes in a randomly generated network. The higher the modularity, the better the set of clusters describe the relationships between the nodes in the network. More formally, consider a network G, which consists of n nodes and m edges. The edges of G are represented by the n-by-n adjacency matrix A. The entry $A_{i,j} = 0$ if no edge exists between nodes i and j. If an edge exists, then $A_{i,j} = w_{i,j}$, the edge's weight. In our case, the weight of an edge is the count pertaining to that association. Also, since our network is undirected, $A_{i,j} = A_{j,i}$. Modularity (Q) is defined as follows:

$$Q = \frac{1}{2m}\sum_{i,j}\left[A_{i,j} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j)$$

where $k_i = \sum_j A_{i,j}$, which is the sum of all the edge weights connected to node i, c_i is the cluster assigned to node i, and $\delta(c_i, c_j) = 1$ if $c_i = c_j$ or 0 otherwise. The value $\frac{k_i k_j}{2m}$ is the expected number of edges connecting nodes i and j in a randomly configured network. A higher value for $A_{i,j} - \frac{k_i k_j}{2m}$ indicates a stronger connection between nodes in a cluster, and thus a higher value for Q indicates a better set of clusters that describes the relationships between the nodes in the graph.

The Louvain employs a greedy strategy to optimize modularity over a series of passes. Each pass consists of two steps. The first is to iterate through all the nodes in the network and move nodes to the clusters of other nodes which produce the maximum gain in modularity. The second step is to generate a new network by condensing the clusters into single nodes. Step 1 is then run on this new network, starting a new pass. Once completed the algorithm returns a hierarchy of clusters, where the last set contains the fewest. We use the final set of clusters produced by the Louvain Algorithm. We generally found that when run on the semantic networks we generated, the final pass of the algorithm generates about 12 to 20 clusters, which is small enough for a user to explore and large enough to contain a variety of unique ideas related to the concept.

D3: Concretize abstract concepts

In the formative study, even when users found relevant associations, they still had trouble identifying concrete objects and actions that could visually represent the concept. Therefore, our third design goal was to enable users to explore concrete words related to contexts associated with the concept. To support D3, users can explore related concrete words for each of the words within the clusters. For example, in Figure 4c, the user expanded *regulation*, one of the cluster words, to view its related concrete words. The most concrete word related to *regulation* is *referee*, which led to two good symbols: a soccer referee and a whistle. Thus, by exploring these related concrete terms, users can find objects and actions related to abstract clusters like "rule, government, governance".

Concreteness is a measure of how physically perceptible something is. For example, words like *seashell* are more concrete than words like *fun*. To incorporate concreteness, we use a crowd-sourced dataset that includes concreteness ratings for 40,000 English words and phrases [6]. Crowd-workers rated words on a scale from 1 (abstract) to 5 (concrete). When resorting the words, we incorporate both concreteness and strength of association, as sorting by concreteness alone leads to random words at the top. To include concreteness, we converted the ratings to a 0 (abstract) to 1 (concrete) scale. Similarly, we convert the association strength count for each related word into a value ranging from 0 to 1; we divide each count by the maximum count in the list. Finally, we multiply these two values and sort by this product. Consider the examples shown in Table 1. Instead of abstract related terms like *fun*, the concrete lists provide imageable words like *motorcycle* that can symbolize the abstract concept: *exciting*.

Exciting		Future	
Related	Concrete	Related	Concrete
Fun	Motorcycle	Past	Crystal Ball
Thrill	Race car	Tomorrow	Hovercraft
Happy	Roller coaster	Crystal Ball	Robot
Interesting	Package	Someday	Car
New	Firework	Prediction	Spacecraft

Table 1. Associations sorted by strength of association (related) and by concreteness (concrete) for two abstract concepts: *Exciting* and *Future*

PRELIMINARY CLUSTER STUDY

To address D2 (symbol diversity), SymbolFinder presents clusters of unique associations in order to enable users to easily explore a variety of symbols. In this study we evaluate whether the clusters represent coherent ideas related to the concept and whether they help users find different symbols more efficiently than a simple list of related words. We conducted a comparative user study where participants used two different versions of the SymbolFinder: (1) cluster interface, the tool as described in the SymbolFinder section, and (2) list interface. The list interface is exactly like the cluster interface except for two key differences: (1) there is no phase 1 exploration of clusters, (2) the sidebar consists of the concept's related words, sorted by strength of association. In the list-interface, users can still explore related, concrete terms for each of the concept's related words. After using both versions of the interface, participants completed a semi-structured interview which gauged their preference of the interfaces and their opinions of the clusters' usefulness.

Methodology

We recruited 6 participants (2 male, 4 female, average age of 24.7). Each participant searched for symbols of two different concepts of similar concreteness, randomly chosen from the same visual message data set used in the formative study: *Halloween* (concreteness = 0.64) and *energy* (0.62). Half of the participants found symbols for *energy* with the cluster interface and *Halloween* with the list interface, while half did the opposite. Order of interface usage and concept assignments were randomized. They were given ten minutes to find 20 different symbols per concept. Prior to finding symbols for the two test concepts, they practiced finding symbols for *summer* with each system for up to 10 minutes. Afterwards, they completed a semi-structured interview.

Figure 7. Users collected more unique symbols for each concept with SymbolFinder. Above are results for *control*, where the SymbolFinder user collected 15 unique symbols and the Google Image user collected 8.

Results

Participants found that most of the clusters were coherent and related to the concept. When using the cluster interface, participants kept on average 16.3 ($\sigma = 1.2$) of the 19 *energy* clusters and 12.3 ($\sigma = 0.9$) of the 16 *Halloween* clusters, indicating that the majority of clusters were deemed relevant. Generally, for both concepts, there were a couple clusters that users felt were unrelated, like the "harness, horse, dog" cluster for *energy*. But in the interview afterwards, all confirmed it was generally easy for them to rule out these clusters in phase 1.

Participants had to search less to find symbols with the cluster interface than with the list interface. Users expanded on average 42.5($\sigma = 5.7$) words with the list-interface and 25.47($\sigma = 4.9$) words with the cluster-interface. This difference is predominantly due to phase 1 (Figure 5) of the cluster-interface, where participants collected on average 9.5($\sigma = 3.4$) of the 20 symbols they were asked to find. Subsequently, in phase 2 (Figure 4), users were able to narrow their search and explore fewer words. All users preferred the cluster-interface over the list-interface because they appreciated searching less with the cluster-interface. P1 noted that exploring so many disparate words with the list-interface was "more frustrating. Everything was disorganized. I had to jump from one idea to another". Users had to think more when using the list-interface.

Clusters helped users brainstorm more associations. In the interviews, participants noted that clusters provided greater context to the related words, enabling them to think of more associations. For instance, for P2, the clusters were "a jumping off point. I thought of different types of fuel and vehicles when I saw the 'fuel gas car' cluster." Meanwhile, words in the list-interface suffered from a lack of context. For example, when seeing *green*, a word associated with *energy*, P5 stated that she immediately thought of the color, instead of *renewable energy*, its actual association. Clusters formed more coherent ideas with clearer connections to the concept than single words.

To summarize: participants found the clusters to be coherent and related to the concept, searched less for symbols with the cluster-interface than with the list-interface, and more easily formed their own connections with the clusters than with the list-interface.

EVALUATION

In this study we evaluate whether users can find more unique symbols with SymbolFinder than with Google Images for both very abstract and less abstract concepts. Google Images is a good baseline as both professional icon designers and novice designers alike use it to browse visualizations of ideas [41]. We also compare the perceived difficulty of finding symbols with SymbolFinder to finding symbols with Google Images.

Methodology

We recruited 10 graduate students: 2 female and 8 male, with an average age of 26.6. Each participant found symbols for six concepts and after each concept, completed a NASA-TLX questionnaire to measure perceived effort [18]. Prior to finding symbols, each participant was shown a slide-deck explaining the task: find 20 different symbols for each term. They are also shown the symbol rules in Figure 2; we emphasize that the symbols should be unique, displaying different objects and activities.

We compare SymbolFinder with Google for concepts of varying levels of concreteness. The six concepts were randomly selected from the same visual messaging dataset [24] from three levels of concreteness: most concrete (*fast*, concreteness=0.66, *art*, 0.83), less concrete (*dangerous*, 0.46, *rugged*, 0.55), least concrete (*control*, 0.38, *simple*, 0.32). Users found symbols for the concepts in the following order: *fast*, *dangerous*, *control*, *art*, *rugged*, *simple*. To counterbalance the experiment, we had half of the participants use SymbolFinder for the first three concepts and half use it for the last three. For each tool, users find symbols for concepts from three different levels of concreteness. We hypothesize that **users will find more unique symbols across all concepts with SymbolFinder**. We also hypothesize that **SymbolFinder will help more with the most abstract concepts**, as concepts like *simple* and *control* apply to multiple diverse contexts, and SymbolFinder presents a diverse set of unique associations (D2).

Prior to finding symbols, users practiced with each interface for ten minutes to collect symbols for the concept *summer*.

	SymbolFinder	Google	P-value
Total Symbols	26.1 (9.6)	16.2 (4.96)	**<0.001**
Good Symbols	22.7 (8.26)	14 (4.49)	**<0.001**
Unique Symbols	14.8 (5.5)	9.92 (2.89)	**<0.001**

Table 2. The total number of symbols, good symbols, and unique good symbols found by participants in the second user study. Bold P-values indicate statistical significance. Participants found significantly more unique symbols with SymbolFinder than with Google.

	SymbolFinder	Google	P-value
Mental Demand	5.13 (1.41)	6.8 (1.97)	**<0.001**
Physical Demand	1.97 (1.43)	3.97 (2.58)	**<0.001**
Temporal Demand	5.13 (2.55)	6.17 (2.44)	0.10
Performance	6.77 (2.03)	5.9 (1.8)	0.061
Effort	4.87 (1.67)	7.43 (1.52)	**<0.001**
Frustration	3.0 (1.93)	4.33 (1.92)	**0.034**

Table 3. NASA-TLX Questionnaire results. Bold P-values indicate statistical significance. SymbolFinder required significantly less mental demand, effort, physical demand and frustration.

During this practice, users were guided through the features of each tool. In the Google condition, participants used the standard Google Image web-page. They were shown the related search terms that appear above the images as well as the filter tools. While practicing, users could ask any questions about the goodness of their symbols. When finding symbols for the six concepts, they had a "cheat sheet", containing the rules of figure 2, but were no longer helped by the experimenter. Participants had 10 minutes to find 20 or more symbols for each concept.

After finding symbols for each concept, participants completed a NASA Task Load Index (NASA TLX) survey, to understand their perceived effort for each word-tool combination [18]. The NASA TLX is a standard measure for subjective workload when using a tool, measuring: mental demand, physical demand, temporal demand, performance, effort, and frustration level. We hypothesize that **users will find SymbolFinder easier to use** as it helps users brainstorm associations (D1) and find concrete objects (D3), requiring less mental demand and effort.

We recruited two graduate students in design to annotate the collected images for unique, good symbols. They annotated good symbols based on the rules of Figure 2, which are rooted in symbol theory. Along with these rules, annotators were instructed to reject images that seemed unrelated to the concept. As well as symbol goodness, the annotators also labeled duplicate symbols. They were instructed to label two images as duplicates if they conveyed the same object or activity, regardless of style, background or color. Because of the natural subjectivity of this task, we had the annotators label two practice sets of images for good and unique symbols together. They then annotated the images collected during the evaluation separately.

Results

We report the percent agreement between the two raters for determining good and unique symbols (and the Cohen's Kappa correlation coefficients) to be 94%(0.74) and 96%(0.75) respectively. To calculate the results in Table 2, we averaged the number of good and unique symbols determined by the raters for both conditions.

Participants found more unique symbols with SymbolFinder. We conducted paired t-tests and found with SymbolFinder, participants collected significantly more unique, good symbols than with Google Images, finding an average of 14.8 unique symbols across the six concepts with SymbolFinder and 9.92 unique symbols with Google ($t = 4.39$, $p < 0.001$) (Table 2). Figure 7 shows results for *control*; the SymbolFinder user almost doubles the amount of unique symbols found by the Google Image user. SymbolFinder users collected significantly more symbols, many of them good. Participants on average collected 26.1 symbols per concept with SymbolFinder compared to 16.2 with Google ($t = 4.5$, $p < 0.001$) , with an average of 22.7 good symbols per concept, compared to 14 with Google ($t = 4.75$, $p < 0.001$) . In both conditions, users had a tendency to select multiple symbols of the same object or activity, leading to a drop off from good symbols to unique symbols.

SymbolFinder helped users find more symbols for both very abstract and less abstract concepts. Participants were able to find more unique symbols for each of the six concepts with SymbolFinder than with Google, regardless of the concept's concreteness (Table 4). We conducted unpaired t-tests and found the difference in unique symbols was statistically significant for two concepts: *fast* a less abstract term and *simple*, a more abstract term. Participants collected an average of 18.4 unique symbols for *fast* with SymbolFinder and 10.7 with Google ($t = 6.4$, $p < 0.001$). For *simple*, participants collected 14.3 unique symbols with SymbolFinder and 7.3 with Google ($t = 2.4$, $p = 0.04$). For both these terms, despite their difference in abstractness, SymbolFinder presented a number of associations Google users did not consider. For example, when finding symbols for *simple* with Google, participants fixated on a single meaning of *simple*, "easily understood" and collected symbols of shapes and simple mathematics. Meanwhile, SymbolFinder users found symbols for this context and many others, including "primitive" and "pure", collecting symbols like the caveman wheel and a water droplet.

SymbolFinder was easier to use than Google. A summary of the NASA-TLX results are shown in Table 3. Participants found using SymbolFinder to be significantly less mentally demanding than Google Images, reporting an average mental demand of 5.13 with SymbolFinder and 6.8 with Google Images ($t = -3.98$, $p < 0.001$). Similarly, frustration was also significantly lower with SymbolFinder (3) than with Google (4.33), ($t = -2.23$, $p = 0.034$). Users often hit dead-ends of redundant symbols with Google, increasing frustration and mental demand as they brainstormed unassisted for new ideas. Physical demand was significantly lower for SymbolFinder, with an average of 1.97, compared to 3.97 with Google ($t = -4.85$, $p < 0.001$). This is likely due to users having to copy and paste

Concept	Art		Fast		Rugged		Dangerous		Control		Simple	
Concreteness	0.83		0.66		0.55		0.43		0.39		0.32	
Condition	SF	Google	SF	Google	SF	Google	SF	Google	SF	Google	SF	Google
Unique Symbols	13.4	10.1	**18.4**	10.7	11.1	9	18.4	12	13.4	10.2	**14.3**	7.3

Table 4. With SymbolFinder (SF), participants found more unique symbols for all concepts, across all levels of concreteness than with Google. For *fast* (a less abstract term) and *simple* (a very abstract term) this difference is statistically significant.

images from Google into a slide-deck. For the same reasons as mental and physical demand, effort was also significantly lower for SymbolFinder (4.87) than Google (7.43), ($t = -6.41$, $p < 0.001$). Temporal demand was lower with SymbolFinder (5.13) than Google (6.17), but not significantly so ($t = 1.7$, $p = 0.1$). When users ran out of ideas with Google, they felt a greater time pressure to find 20 symbols. Some users felt more time pressure with SymbolFinder as they wanted to explore all the related words and images. Finally, participants felt they had performed better with SymbolFinder (6.77) than with Google (5.9), but this difference was not significant ($t = 1.95$, $p = 0.06$).

DISCUSSION

With SymbolFinder, users found significantly more unique symbols with less perceived difficulty for both abstract and concrete concepts. In the following section we discuss subjective influence in finding symbols, limitations and future work to improve the system, as well as applications of SymbolFinder to other visual media.

Subjective influences in finding symbols

Even when introduced to a number of associations related to the abstract concept, some participants would still fixate on a single association. For example, P10 predominantly found images of rough textures for *rugged*, such as rocky paths. Even though SymbolFinder presented many other associations, like durable tools, he did not select any images from these categories, as his own definition of *rugged* was much more narrow. While some users interpreted *rugged* in a very narrow way, others interpreted *simple* very broadly. P1 collected images that were visually simple depictions of objects and ideas, even if these ideas were not related to *simple*. For instance, on of the symbols she collected was an icon with a red slash over a camera, which she thought was a simple depiction of "no camera". Finding representative objects for abstract concepts is a subjective task and depends on the user's understanding of the concept.

Limitations and future work

When using SymbolFinder users would sometimes struggle to find simple images of activities and objects. SymbolFinder returns 10 images per search to prevent scanning images for too long. However, a simple image of an object might not appear in these images, resulting in users giving up on finding one. Some users were able to find a good image by being more specific in their search. For example, image results of *lightning* would often bring up a number of lightning bolts in the sky. One user was able to get an image of a single bolt by searching *one lightning bolt*. In the future, we could apply automatic techniques to select simple images from results, using object detection to gauge complexity by understanding how many items or people are in the image. We could also include various editing features for users to easily crop parts of images that could be a good symbol. Finally, we could also incorporate these Google Image search techniques, such as "[one] + object", to assist users to find simple images.

Sometimes the clusters contain words that are not related to the concept. When the local semantic network is constructed, all the nodes in the first level are used to build the second level, introducing irrelevant words. For example, in Figure 6, the first-level node: *remote*, introduces words like *far* which is not related to the concept: *control*. To reduce the number of unrelated nodes, we could use different measures of node importance to prune words that are less connected in the network. Intuitively, in the local network for *control*, words like far are most likely connected to very few, or even one word, indicating that they are less related to the central concept. Another route is to measure the semantic similarity of each word in level 2 to the root concept using word embeddings and select words that reach a threshold of similarity. In the future, we plan to explore these directions.

Applying SymbolFinder to other visual media databases

Though built with Google Images as its image database, SymbolFinder can be applied to any other image database, depending on the users' needs. SymbolFinder's core feature is the clustered local semantic network, which enables users to explore a number of related and unique contexts associated with a concept. This idea can be applied to help explore other image databases, such as the Noun Project, a popular icon database containing a wide variety of black-and-white icons. At the same time, we could apply local semantic networks to any visual media database to help users explore related but diverse Gifs, videos, and memes for an abstract search term. Gifs are often used to concisely and quickly convey an abstract emotion or idea in conversation [2]. We could help users find visualizations that convey their idea more easily by letting them explore a number of gifs conveying different flavors of the concept.

CONCLUSION

Novice graphic designers have trouble finding diverse symbols for abstract concepts. We introduce SymbolFinder, an interactive tool that enables users to find diverse symbols for abstract concepts. SymbolFinder generates and clusters a local semantic network to present a unique set of diverse associations for users to brainstorm symbols from. In the future, local semantic networks can be applied to other visual media databases, to help users explore diverse, semantically-related results.

REFERENCES

[1] Salvatore Andolina, Khalil Klouche, Diogo Cabral, Tuukka Ruotsalo, and Giulio Jacucci. 2015. InspirationWall: Supporting Idea Generation Through Automatic Information Exploration. In *Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition (C&C '15)*. Association for Computing Machinery, New York, NY, USA, 103–106. DOI: http://dx.doi.org/10.1145/2757226.2757252

[2] Saeideh Bakhshi, David A. Shamma, Lyndon Kennedy, Yale Song, Paloma de Juan, and Joseph "Jofish" Kaye. 2016. Fast, Cheap, and Good: Why Animated GIFs Engage Us. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 575–586. DOI: http://dx.doi.org/10.1145/2858036.2858532

[3] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, 10 (2008), P10008.

[4] M. A. Borkin, Z. Bylinskii, N. W. Kim, C. M. Bainbridge, C. S. Yeh, D. Borkin, H. Pfister, and A. Oliva. 2016. Beyond Memorability: Visualization Recognition and Recall. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 519–528.

[5] M. A. Borkin, A. A. Vo, Z. Bylinskii, P. Isola, S. Sunkavalli, A. Oliva, and H. Pfister. 2013. What Makes a Visualization Memorable? *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2306–2315.

[6] Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior research methods* 46, 3 (2014), 904–911.

[7] Joel Chan, Pao Siangliulue, Denisa Qori McDonald, Ruixue Liu, Reza Moradinezhad, Safa Aman, Erin T. Solovey, Krzysztof Z. Gajos, and Steven P. Dow. 2017. Semantically Far Inspirations Considered Harmful? Accounting for Cognitive States in Collaborative Ideation. In *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition (C&C '17)*. Association for Computing Machinery, New York, NY, USA, 93–105. DOI: http://dx.doi.org/10.1145/3059454.3059455

[8] Minsuk Chang, Leonore V. Guillain, Hyeungshik Jung, Vivian M. Hare, Juho Kim, and Maneesh Agrawala. 2018. RecipeScape: An Interactive Tool for Analyzing Cooking Instructions at Scale. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, Article Paper 451, 12 pages. DOI: http://dx.doi.org/10.1145/3173574.3174025

[9] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. 2004. Finding community structure in very large networks. *Physical review E* 70, 6 (2004), 066111.

[10] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey. 1992. Scatter/Gather: A Cluster-Based Approach to Browsing Large Document Collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '92)*. Association for Computing Machinery, New York, NY, USA, 318–329. DOI: http://dx.doi.org/10.1145/133160.133214

[11] Simon De Deyne, Danielle J Navarro, Amy Perfors, Marc Brysbaert, and Gert Storms. 2019. The "Small World of Words" English word association norms for over 12,000 cue words. *Behavior research methods* 51, 3 (2019), 987–1006.

[12] Simon De Deyne, Daniel J Navarro, and Gert Storms. 2013. Better explanations of lexical and semantic cognition using networks derived from continued rather than single-word associations. *Behavior research methods* 45, 2 (2013), 480–498.

[13] Simon De Deyne and Gert Storms. 2008. Word associations: Network and semantic properties. *Behavior research methods* 40, 1 (2008), 213–231.

[14] Ron S Easterby. 1970. The perception of symbols for machine displays. *Ergonomics* 13, 1 (1970), 149–158.

[15] Cristian Felix, Aritra Dasgupta, and Enrico Bertini. 2018. The Exploratory Labeling Assistant: Mixed-Initiative Label Curation with Large Document Collections. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. Association for Computing Machinery, New York, NY, USA, 153–164. DOI: http://dx.doi.org/10.1145/3242587.3242596

[16] James Fogarty, Jodi Forlizzi, and Scott E. Hudson. 2001. Aesthetic Information Collages: Generating Decorative Displays That Contain Information. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST '01)*. Association for Computing Machinery, New York, NY, USA, 141–150. DOI:http://dx.doi.org/10.1145/502348.502369

[17] James Fogarty, Desney Tan, Ashish Kapoor, and Simon Winder. 2008. CueFlik: Interactive Concept Learning in Image Search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. Association for Computing Machinery, New York, NY, USA, 29–38. DOI: http://dx.doi.org/10.1145/1357054.1357061

[18] Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Advances in psychology*. Vol. 52. Elsevier, 139–183.

[19] Kathleen Hemenway. 1982. Psychological Issues in the Use of Icons in Command Menus. In *Proceedings of the 1982 Conference on Human Factors in Computing Systems (CHI '82)*. Association for Computing Machinery, New York, NY, USA, 20–23. DOI: http://dx.doi.org/10.1145/800049.801748

[20] Nigel Holmes and Rose De Neve. 1985. *Designing pictorial symbols*. Rac Books.

[21] Enamul Hoque, Orland Hoeber, and Minglun Gong. 2013. CIDER: Concept-based image diversification, exploration, and retrieval. *Information processing & management* 49, 5 (2013), 1122–1138.

[22] William K Horton. 1994. *The icon book: Visual symbols for computer systems and documentation*. John Wiley & Sons, Inc.

[23] WH Huggins and Doris R Entwisle. 1974. *Iconic Communication: An Annoted Bibliography*. Johns Hopkins University Press.

[24] Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. 2017. Automatic understanding of image and video advertisements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1705–1715.

[25] Yui Kita and Jun Rekimoto. 2018. V8 Storming: How Far Should Two Ideas Be?. In *Proceedings of the 9th Augmented Human International Conference (AH '18)*. Association for Computing Machinery, New York, NY, USA, Article Article 14, 8 pages. DOI: http://dx.doi.org/10.1145/3174910.3174937

[26] Janin Koch, Andrés Lucero, Lena Hegemann, and Antti Oulasvirta. 2019. May AI? Design Ideation with Cooperative Contextual Bandits. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article Paper 633, 12 pages. DOI:http://dx.doi.org/10.1145/3290605.3300863

[27] Paul A Kolers. 1969. Some formal characteristics of pictograms. *American scientist* 57, 3 (1969), 348–363.

[28] Aaron Marcus. 1984. Corporate identity for iconic interface design: The graphic design perspective. *Interfaces in Computing* 2, 4 (1984), 365–378.

[29] Siné JP Mcdougall, Martin B Curry, and Oscar de Bruijn. 1999. Measuring symbol and icon characteristics: Norms for concreteness, complexity, meaningfulness, familiarity, and semantic distance for 239 symbols. *Behavior Research Methods, Instruments, & Computers* 31, 3 (1999), 487–519.

[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[31] Murli Nagasundaram and Alan R Dennis. 1993. When a group is not a group: The cognitive foundation of group idea generation. *Small Group Research* 24, 4 (1993), 463–489.

[32] American Institute of Graphic Arts. 1974. *Symbol Signs*. National Technical Information Service.

[33] Masataka Ono, Makoto Miwa, and Yutaka Sasaki. 2015. Word embedding-based antonym detection using thesauri and distributional information. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 984–989.

[34] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[35] Yang Shi, Yang Wang, Ye Qi, John Chen, Xiaoyao Xu, and Kwan-Liu Ma. 2017. IdeaWall: Improving Creative Collaboration through Combinatorial Visual Stimuli. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. Association for Computing Machinery, New York, NY, USA, 594–603. DOI: http://dx.doi.org/10.1145/2998181.2998208

[36] Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Niklas Elmqvist, and Nicholas Diakopoulos. 2016. *Designing the user interface: strategies for effective human-computer interaction*. Pearson.

[37] Grant Strong, Enamul Hoque, Minglun Gong, and Orland Hoeber. 2010. Organizing and browsing image search results based on conceptual and visual similarities. In *International Symposium on Visual Computing*. Springer, 481–490.

[38] Hao-Chuan Wang, Dan Cosley, and Susan R. Fussell. 2010. Idea Expander: Supporting Group Brainstorming with Conversationally Triggered Visual Thinking Stimuli. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*. Association for Computing Machinery, New York, NY, USA, 103–106. DOI: http://dx.doi.org/10.1145/1718918.1718938

[39] Hao-Chuan Wang, Susan R. Fussell, and Dan Cosley. 2011. From Diversity to Creativity: Stimulating Group Brainstorming with Cultural Differences and Conversationally-Retrieved Pictures. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work (CSCW '11)*. Association for Computing Machinery, New York, NY, USA, 265–274. DOI:http://dx.doi.org/10.1145/1958824.1958864

[40] Jingdong Wang and Xian-Sheng Hua. 2011. Interactive Image Search by Color Map. *ACM Trans. Intell. Syst. Technol.* 3, 1, Article Article 12 (Oct. 2011), 23 pages. DOI:http://dx.doi.org/10.1145/2036264.2036276

[41] Nanxuan Zhao, Nam Wook Kim, Laura Mariah Herman, Hanspeter Pfister, Rynson W.H. Lau, Jose Echevarria, and Zoya Bylinskii. 2020. ICONATE: Automatic Compound Icon Generation and Ideation. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. DOI:http://dx.doi.org/10.1145/3313831.3376618