Automatic Generation of Semantic Icon Encodings for Visualizations

Vidya Setlur
Tableau Software
Menlo Park, CA 94025
vidya.setlur@gmail.com

Jock D. Mackinlay
Tableau Software
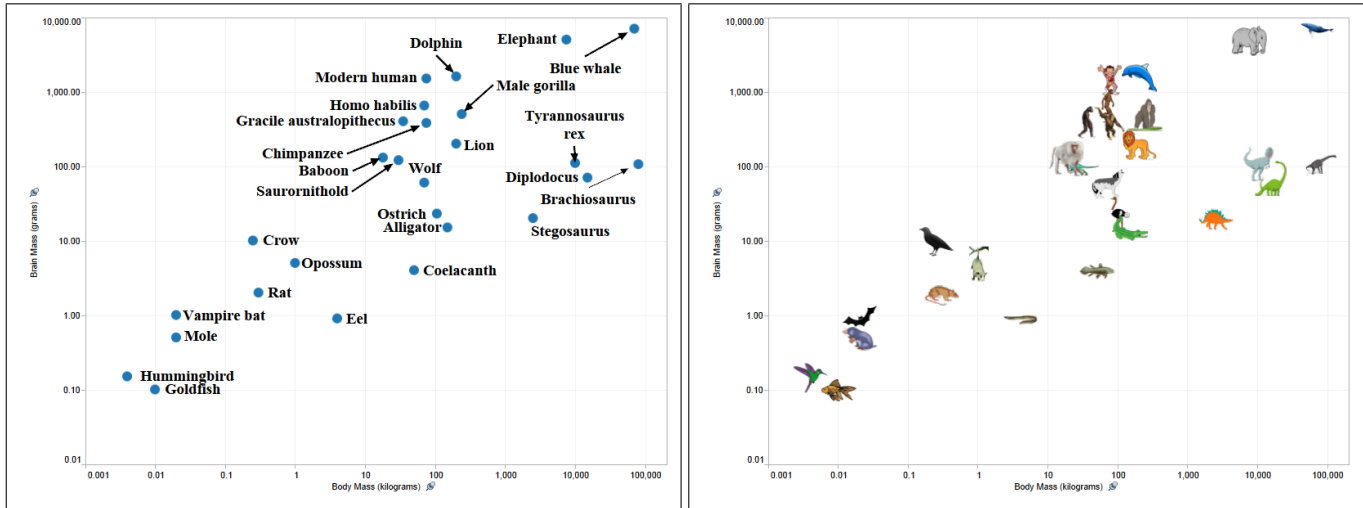Seattle, WA 98103
jmackinlay@acm.org

Figure 1. Left: Carl Sagan's scatter plot shows the ratio of brain to body mass on a log-log scale. This ratio is a good way to measure intelligence. The Sagan view shows that labels can make it difficult to see the position of the marks. Right: Icons are a good way to indicate the semantics of marks, as the human visual system is very effective at seeing icons. Here, these icons were automatically generated by our system.

ABSTRACT

Authors use icon encodings to indicate the semantics of categorical information in visualizations. The default icon libraries found in visualization tools often do not match the semantics of the data. Users often manually search for or create icons that are more semantically meaningful. This process can hinder the flow of visual analysis, especially when the amount of data is large, leading to a suboptimal user experience. We propose a technique for automatically generating semantically relevant icon encodings for categorical dimensions of data points. The algorithm employs natural language processing in order to find relevant imagery from the Internet. We evaluate our approach on Mechanical Turk by generating large libraries of icons using Tableau Public workbooks that represent real analytical effort by people out in the world. Our results show that the automatic algorithm does nearly as well as the manually created icons, and particularly has higher user satisfaction for larger cardinalities of data.

Author Keywords

icon encodings; natural language processing (NLP); visualization; image retrieval.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

Icon encodings are graphical elements that are often used as a visual syntax to represent the semantic meaning of marks representing categorical data. These mappings of information to display elements help the user to perceive and interpret the visualization. These encodings can be effective in enabling visual analysis because they are often rapidly and efficiently processed by the pre-attentive visual system rather than attentive effort [8]. The Gestalt principles reflect strategies of the human visual system to recognize and spatially categorize icons in order to create a meaning understanding of the visualization [4]. In Figure 1, the Sagan view on the left shows that labels can make it difficult to see the position of the marks. However, the figure on the right provides enough semantic information about the data, allowing the visual system to apply visual and spatial categorization in order to understand the scene.
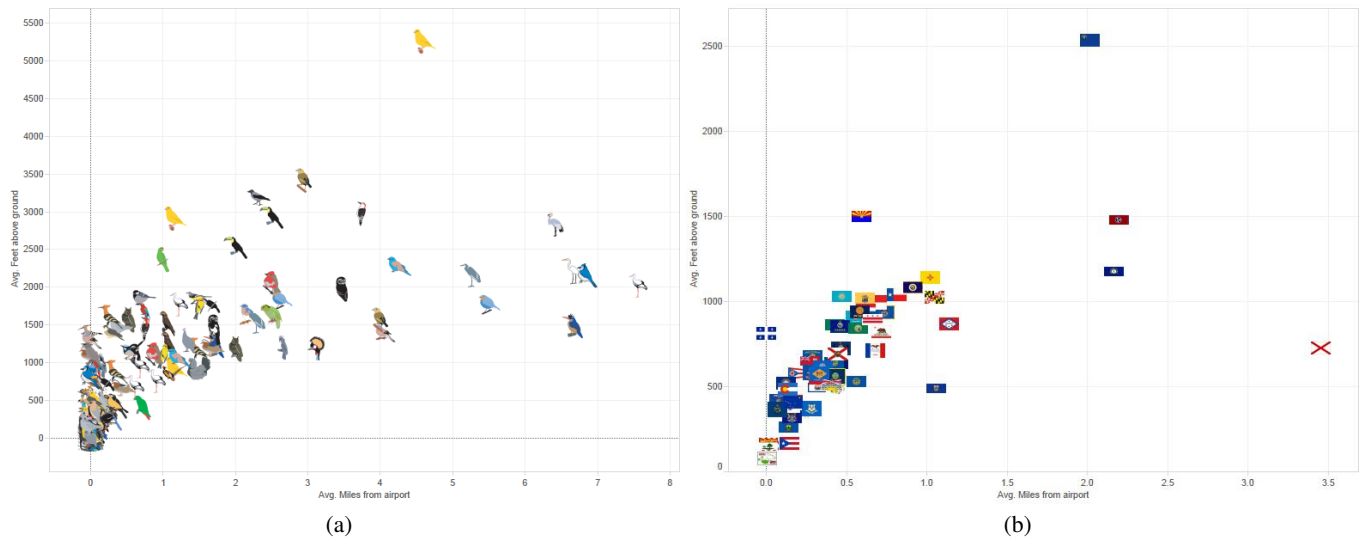
Figure 2. Illustrating the importance of icon encodings during the visual flow of analysis. Left: A scatterplot showing the correlation of bird strikes with average miles from airport and average feet from found. Here, the birds are encoded with custom icons generated by our algorithm. Right: A scatterplot showing the correlation of bird strikes in various states. Here, the states are encoded with custom state flag symbols.

Icon encodings play an important role in the flow of visual analysis. Consider Figure 2a, where a user is looking at a scatterplot to observe how the number of bird strikes with aircraft, is correlated with average miles from airport and average feet from the ground. Here, she is using icons for the wildlife data. While she is looking at this view, she would like to explore the same correlation with respect to state (Figure 2b). When she moves from the first view to the next, it would be useful if the visualization tool could provide a fast and seamless way to assign state icons to all the states in the view. This would keep her engaged in the flow of analysis, rather than manually trying to find the right icons. Quite often, visualization tools contain icon libraries, but are rather small and do not contain semantically meaningful icons. For example, Tableau (`http://www.tableausoftware.com/products/desktop`) provides a default 10 palette of circles, squares, triangles and crosses. For a large dataset like what is illustrated in this example, we need an automatic mechanism to find the right icons.

In particular, our focus is on authors who want to use icons in visualizations and would benefit from an algorithm for getting appropriate icons sets. We have evidence that icons are a natural alternative to color, particularly when the cardinality of categorical data is greater than 20. Considering the Tableau Public dataset, out of $28,502$ unique workbooks containing categorical data, 8700 (30.52%) of them used icons. In particular, a majority of these visualizations were scatterplots (47.82 %), followed by text tables (20.60 %), maps (18.63 %), bar charts (8.33 %) and the remaining are a negligible number of visualizations including line charts and pie charts. We hypothesize that people may be motivated to use icons due to their inherent qualities of rapid recognition and conveying the semantics of the data. Further, icons tend to render well in compact spaces owing to their often square aspect ratios, compared to equivalent text label counterparts. While

the scientific studies of basic shapes can likely be extended to well-designed icon sets, we make no claims about that. Further perceptual and cognitive studies are required to fully understand when icons best replace standard marks for data points. Key to our contribution is that authors clearly feel a need for encoding categorical information in point marks even if it requires significant work. After all, we are visual creatures.

This paper presents an automatic algorithm that help ease that interruption in the visual flow of analysis, by automatically generating large icon libraries for categorical data. By employing natural language processing (NLP) techniques, the algorithm takes into account the inherent design and functional characteristics of icon encodings to provide more meaningful images to the user.

RELATED WORK

In this section, we first summarize previous work on graphical perception of visual encodings and then review some of the most related work from the iconography literature.

Graphical Perception of Visual Encodings

There have been several bodies of research that rank the effectiveness of various visual encodings in the context of visual search and identification for nominal, ordinal and quantitative data. These perceptual channels are not created equal; rather some tend to have more representational power than others because of the constraints of the human perceptual system and the characteristics of the data [3, 13]. Symbols have been determined to be inappropriate for encoding quantitative data because they do not conform to a natural continuum [3, 10]. However, symbols can be effective for categorical data, particularly nominal values, and tend to perform better than color encodings for larger cardinalities [15]. In addition, basic geometric shapes have a long history in print-based data graphics when color was too expensive. Smith and Thomas

compared shapes, geometric forms and symbols in a counting task, with symbols having the most accuracy and faster completion times [21]. In addition, they determined that while color codes produce better results than shape codes at a smaller cardinality (five elements), humans can distinguish far more shapes than colors. Thus, when data require more than the small number of distinguishable colors, shape encodings may yield better results. Our work builds upon these perception principles to extend shape encodings to be more meaningful, and generating icons that tie to the semantics of the data.

Iconography
Well-designed icons enjoy advantages such as easy recognizability [20]. Studies have found that the visual and cognitive features of icons significantly influence an icons effectiveness [2, 5, 17]. The effective depiction of an icon often depends on how semantically representative the image is to the information it represents [7].

Previous work has used a number of graphics techniques for various types of content search tasks. Woodruff *et al.* present designs for enhanced thumbnails that can be used to search the web [23]. Several visual file icon representations exist to help make the task of finding a target file easier [9, 11, 19]. In the mobile domain, there have been attempts to adapt content for mobile web browsing [1, 22].

In the spirit of generating meaningful icons, our work presents an algorithm for generating semantically relevant icons to enhance the effectiveness and aesthetics of the visualization. Particularly, the novelty of the work lies in the retrieval and scoring of images based on perceptual and design heuristics for icon encodings.

CONTRIBUTION
This paper describes a novel method for helping users automatically retrieve icon sets, and easily incorporate the icon encodings in their visualizations as they go about their flow of analysis. By developing an automatic algorithm that is compatible or better than human-generated ones, the value the paper brings to the user experience is saving precious time while authoring these visualizations. Our algorithm automatically finds each icon in 1.04 seconds (mean). This number is important for any cardinality, but especially valuable when dealing with larger sets of data.

In particular, the main contributions of this paper are:

- creating an automatic technique that leverages natural language processing (NLP) for retrieving relevant imagery from the Internet.

- a scoring metric that incorporates query and image relevance, and global image features to provide icons for the underlying data, that are relevant, consistent, yet discriminative.

- providing design implications and directions for how icons can be useful.

APPROACH
We extracted 8700 Tableau Public workbooks that contained user-generated icon sets to develop and test our algorithm. Tableau Public (`http://www.tableausoftware.com/public/community`) is a free service that lets anyone publish interactive visualizations to the public web and embed them in blogs and articles. Public users are journalists, bloggers, students, hobbyists, government employees, and concerned citizens. Tableau Public represents one of the largest samples of data visualizations with over $260,000$ workbooks authored by $38,000$ distinct owners.

System Implementation
The algorithm is implemented in Python 3.3.2, using its Natural Language Toolkit (`http://nltk.org/`). While any image repository could be used for retrieving images, we chose Google's Image API (`https://developers.google.com/appengine/docs/python/images/`) because the service provides comprehensive set of highly relevant images with the flexibility of customizing the search using input parameters. Our algorithm automatically finds each icon in 1.04 seconds (mean).

Context Establishment
A meaningful icon employs symbolic imagery that is semantically connected to the data. To find imagery that makes this connection, we need to establish a context for the categorical data. We do so by leveraging the underlying XML format of Tableau workbooks (TWB files). TWB files are much like Microsoft Excel workbooks. They contain one or more worksheets or dashboards and contain all the information about the various visualizations. We employ XQuery and XPath as the XML search system for querying tags and their text to identify semantic correlations to that of the categorical set. While XQuery provides the means to extract and manipulate data from XML documents, containing a set of XPath expression syntax to address specific parts of an XML document.

In order to identify the workbooks that specifically contain user-generated icon sets, we wrote a rule in XPath that searched for 'map' and 'bucket' entries under the `<encoding attr='shape'>` tag as shown in the XML syntax below. While the 'map' entry contains the icon, the 'bucket's entry contains the corresponding categorical name.

Listing 1. XML snippet of a Tableau Workbook

```xml
<?xml version='1.0' encoding='utf-8' ?>
<workbook>
 <style>
  <style-rule element='mark'>
   <encoding attr='shape' field='[Team]' >
    <map to='asterik>
     <bucket>&quot;Arizona&quot;</bucket>
    </map>
    <map to='circle' >
     <bucket>&quot;Atlanta&quot;</bucket>
    </map>
    <map to='diamond' >
     <bucket>&quot;Baltimore&quot;</bucket>
    </map>
    ...
   </encoding>
  </style>
```

```
   ...
   <worksheet name='Football Team Recruits'>
        ...
     <mark class='Shape' />
      <encodings>
       <shape column='[Team]' />
      </encodings>
      ...
   </worksheet>
  ...
 </workbook>
```

The context establishment process first involves parsing the set of categorical terms for which the icons need to be retrieved. These terms are identified by the tag called 'encoding' where the attribute value is 'shape' (Listing 1). We then proceed to gather some more context around these categorical terms by parsing the 'field' attribute that contains the value for the categorical dataset. We then search for worksheet tags containing the categorical data as a icon encoding, and the name attribute is retrieved. Here in Listing 1, the value of the 'field attribute is 'Team', and the value of the 'name' attribute in the worksheet, is 'Football Team' Recruits).

We then compute semantic relatedness between each of the textual terms retrieved from the 'worksheet' XML tag, to the 'field' attribute value of the categorical data. In order to compute this heuristic, we leverage WordNet [6], a broad coverage lexical network of English words. Here, nouns, verbs, adjectives, and adverbs are each organized into networks of synonym sets (synsets) that each represent one underlying lexical concept and are interlinked with a variety of relations. A simple way to compute semantic relatedness in a taxonomy such as WordNet is to view it as a graph and identify relatedness with path length between the concepts, where the shorter the path from one node to another, the more similar they are [16].

Input term	Top co-occurring words	Score
Country	flag	0.9231
	government	0.5170
	constititution	0.1176
Company	logo	0.9844
	office	0.4153
	employees	0.123
Stock	ticker	0.9738
	market	0.3181

Table 1. Table showing individual relatedness scores to 'symbol'

For each pair of terms t_1 and t_2, we compute semantic relatedness as follows:

$$rel(t_1, t_2) = max_{c_1 \in s(w_1), c_2 \in s(w_2)}[wup(c_1, c_2)]$$

where, c_1 and c_2 are synsets of t_1 and t_2 respectively; $wup(s_1, s_2)$ is the Wu-Palmer similarity function [24] that returns a score denoting how similar two word senses are, based on the depth of the two senses in the Wordnet taxonomy and that of their most specific ancestor node called Least Common Subsumer (LCS), with scores returned in the range $\in [0, 1]$; $s(wi)$ is the set of concepts in the taxonomy that are

senses of word w_i *i.e.*, the relatedness of two words is equal to that of the most-related pair of concepts that they denote.

While several similarity functions exist in the NLP literature, we chose the Wu-Palmer function as it is simple, and has good performance with the data we are analyzing. We then select the related terms that have the top-scoring relatedness to the categorical terms. In practice, we have determined that a threshold score of 0.65 and above, tend to lead to an optimal set of semantically related terms. These highest scored terms subsequently serve as additional metadata for querying images for each of the categorical terms. In the example in Listing 1, 'Football' and 'Recruits' have respective scores of 0.834 and 0.427 with the term 'Team.' We hence use the metadata 'Football' in conjunction with all the categorical team names as the seed query to the Query Expansion set described in the following section.

Query Expansion
While the meta data and the individual categorical terms can be used as input queries to an image search engine, the query words may be quite different to the ones used in the metadata describing the semantics of the imagery. That means a gap exists between the query space and document representation space, resulting in lower precisions and recalls of queries. We thus use automatic query expansion to augment terms to each of the queries for the categorical items to expand the probability of obtaining a larger corpus of images that match the query. In our work, we use the WordNet ontology by applying a query expansion method, based on the synonymy, hypernymy and hyponymy relationships, to both the metadata and individual categorical terms. However, these three relations have different semantic relevances to the query word. Hypernyms occur at the upper layers of the Wordnet hierarchy and have more general semantics to help broaden the search in the case where not enough results are obtained (*e.g.* 'vehicle' is a hypernym of 'car'). Hyponyms occur at lower layers or at the same layer of the Wordnet hierarchy, and have more concrete semantics and tend to retrieve fewer and more precise results, narrowing the search (*e.g.* 'convertible' is a hyponym (a kind) of 'car') [12]. In other words, with those expanded words together, the system could raise both the query precision and recall.
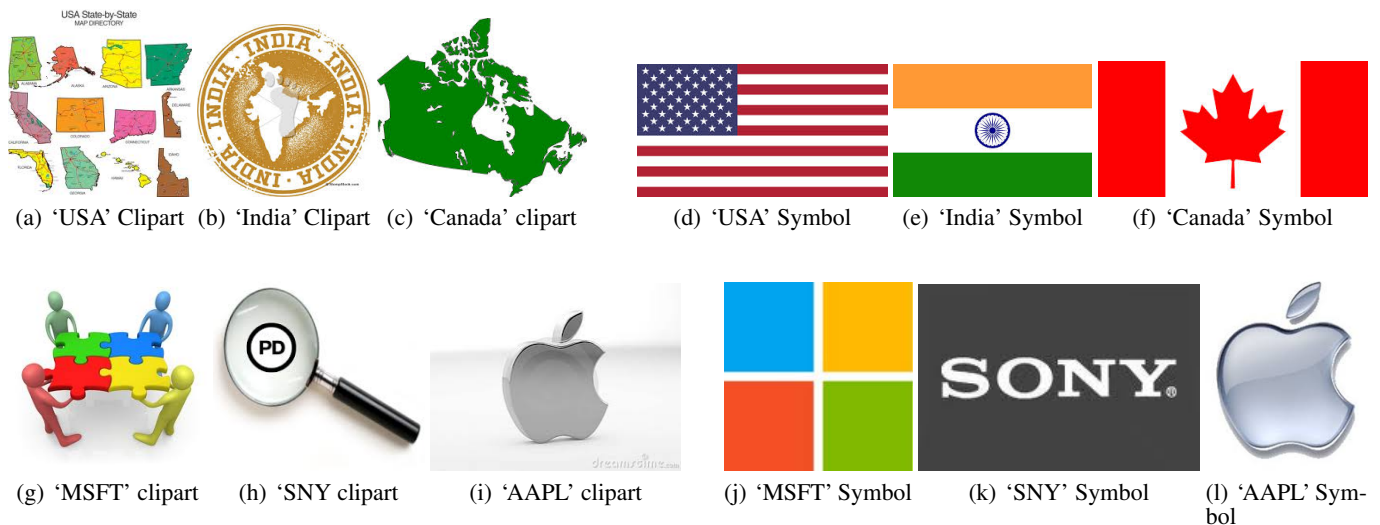
To determine the number of query expansions along each direction, the average precision is used as the objective function for computing the optimal factor for each direction. The objective function is computed using two metrics, namely confidence and support for scoring term associations. So, term association between two terms t_i and t_j is determined as:

$$I(t_i, t_j) = I(t_i) \cap I(t_j)$$

where $I(t_i)$ and $I(t_j)$ denote images that contain metadata including term t_i and t_j respectively. Then, $I(t_i) \cap I(t_j)$ is the set of images that include both t_i and t_j.

We the define confidence C as:

$$C(t_i, t_j) = \frac{\sum I(t_i, t_j)}{\sum I(t_i)}$$

| (a) 'USA' Clipart | (b) 'India' Clipart | (c) 'Canada' clipart | (d) 'USA' Symbol | (e) 'India' Symbol | (f) 'Canada' Symbol |

| (g) 'MSFT' clipart | (h) 'SNY clipart | (i) 'AAPL' clipart | (j) 'MSFT' Symbol | (k) 'SNY' Symbol | (l) 'AAPL' Symbol |

Figure 3. Illustrating the need for the query parameters to return images that are more symbolic in nature. Top: Fig. (a)- (c). The most relevant clipart images retrieved for the countries 'USA', 'India' and 'Canada.' Fig. (d) - (f): By supplying a term 'flag' that is closely connected to the term'symbol', we get flag symbols for each of the countries. Bottom: Fig. (g)- (i). The most relevant clipart images retrieved for the stock symbols 'Microsoft, 'Sony' and 'Apple'. Fig. (j) - (l): By supplying a term 'ticker' that is closely connected to the term'symbol', we get ticker symbols for each of the companies.

where $\sum I(t_i, t_j)$ denotes the total number of images that have metadata including both terms t_i and t_j. $\sum I(t_i)$ are the total number of images with metadata including a term t_i.

Support S is defined as:

$$S_{(t_i, t_j)} = \frac{\sum I(t_i, t_j)}{\sum I}$$

where I denotes the total number of images in the database.

Term Filtering
In the next step, we use C and S to eliminate noise generated in the query expansion that may lead to a decrease in precision. These terms are often low-frequency and unusual words. We employ the association rules to remove the expansion words that have lower support and confidence to the original word. In our system, we use the expanded words which have minimum confidence over 0.6 and support over 0.04 with the original query keyword into our query expansion. For example, for the keyword 'boat', we filter out the words such as 'vas', and 'watercraft.'

Additional Visual Parameters
We employ Google's Image Search API (`http://images.google.com`) to retrieve images based on the query. The goal of the algorithm is to retrieve icons or visual symbols for meaningfully encoding categorical data. We therefore restrict the images returned to being cartoonish as opposed to photographs, by providing 'clipart' as an additional parameter to the query. This tends to work well for many cases like the icons for the animals and birds in Figures 1 and 2. However, we ran into an issue where merely restricting the results to be clipart, did not often provide expected results (Figure 3). We needed a way to retrieve images that were not only clipart images, but were more symbolic in nature [14].

We initially added the term 'symbol' to the query, but that was not always optimal as it was not semantically tied to the term. We then attempted to leverage the Wordnet ontology to see if there were any synset relationships that were semantically related to 'symbol'. However that proved rather futile. So, in order to get other terms associated with the query that may be semantically related to 'symbol,' we use n-grams, to obtain word co-occurrence statistics for term clustering (`http://en.wikipedia.org/wiki/N-gram`). An n-gram is a string of n adjacent words occurring within the thematically related NLTK corpora (`http://nltk.org/api/nltk.corpus.html`). We use a 2-word n-gram called a bi-gram. For instance, if the input query term is 'tea,' some possible bi-grams are 'tea cup', 'hot tea', and 'tea leaves.' For each bi-gram pair of candidate phrase, we compute the semantic relatedness of the co-occurring word to the term 'symbol.' The highest scoring word is then augmented to the existing query in lieu of the Google Image parameter 'clipart' to retrieve symbolic imagery. If there is no co-occurring word that is semantically related to 'symbol', we augment the term 'symbol' to the query and use the default 'clipart' parameter in the query. Table 1 shows examples of top-scoring symbolic terms for a given query term. In the case of people's names, if Wordnet cannot find it in its dictionary corpora, or if its identified to be a proper noun, the 'clipart' parameter is not used with the search.

Global Feature Clustering
The goal in this step, is to retrieve images that are relevant to the given query, satisfying the design constraints of icons. These constraints include being relevant to the content the icons represent, visual consistence, yet discriminative from one another. While we rely on Google's Image API to retrieve relevant images based on the input query, in order to ensure some form of visual consistency across the images, we found that images originating from the same web source

and have similar aspect ratios (*e.g.* rectangular vs. square images) tend to provide more favorable results. We hence, apply a clustering algorithm on the hyperlink and aspect ratio metadata associated with each retrieved images.

So, for every categorical term, we first retrieve the top M (Google's default result set size is 8 images per query) most relevant images. The images are then clustered based on these features using a commonly used clustering algorithm called, group average agglomerative clusterer (GAAC) (`http://nltk.org/api/nltk.cluster.html`). The GAAC clusterer starts with each of the N vectors as singleton clusters. It then iteratively merges pairs of clusters which have the closest centroids. In practice, we found that $N = 4$, provides enough number of images per categorical term and for cluster convergence.

We then iteratively proceed in decreasing order of cluster dominance (*i.e.* starting from the largest cluster) to determine if an image for that categorical item is available. In order to associate subject-centric images for each categorical item, we check if the image has a monochromatic background. For this, we employ OpenCV's GrabCut algorithm [18] to create a mask around the fringes of each image. If the mask is monochromatic, it is deemed to have a segmentable background that can be modified to be transparent. If the mask is deemed otherwise, the background is not modified. A monochromatic background image in the most dominant cluster, is chosen for the corresponding categorical item. If there are no images with monochromatic backgrounds, then the image in the largest cluster is just chosen.

EVALUATION

Since the goal of our algorithm is to reduce the time taken for users to go and find icons for their categorical data, how well would the automatic algorithm fare when compared to the user-generated custom icons? One hypothesis concerns the effect of cardinality on the performance of the automatic vs. manual techniques - The automatic method would perform better than the user-generated method for categorical data with high cardinality given the amount of user effort it takes for generating a large icon set.

Category	Number of workbooks
Sports	122
Places	75
Entertainment	75
People	69
Consumer products	60
Companies	59
Health and Science	32

Table 2. Categorical breakdown of the 492 **Tableau workbooks used in the study**

Design

To answer this question, we conducted a study on Mechanical Turk asking participants to rate the user-generated and auto-generated icon sets from a subset of the same 8700 unique Tableau workbooks that was described in the Approach Section. We randomly sampled a subset of 492 workbooks that
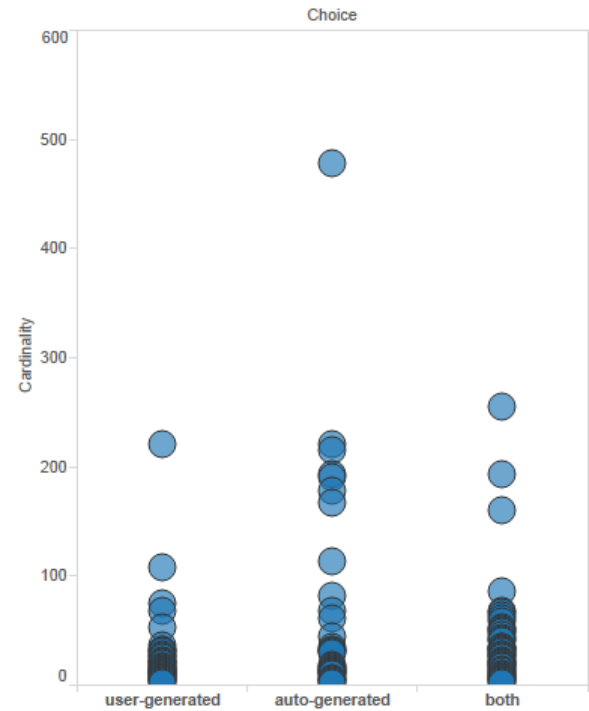

Figure 4. Circle graph to show the effect of cardinality on preference of icon sets. Here, each translucent circle represents a workbook, and the darkness in color correlates to the density of workbooks at a given cardinality. The graph shows that cardinality is a significant factor for describing the user preference ($p - value = 0.003$)

all contain user-generated custom icons. By manually categorizing each of the 492 workbooks, we identified the following categories: 'Sports' (*e.g.* football, basketball teams), 'Companies' (*e.g.* Microsoft, Google, Amazon), 'Places' (*e.g.* country and state names), 'People' (*e.g.* names of political candidates, actors), 'Consumer products' (*e.g.* coffee and chocolate brands), 'Health and science' (*e.g.* health hazards, pollutants, animal species), and 'Entertainment' (*e.g.* movie titles, music albums). The categorical distribution of the 492 workbooks is indicated in Table 2. In general, this random subset of workbooks tends to be representative of the distribution prevalent on Tableau Public. The set of workbooks had a large degree of variance in cardinality ($\mu = 32.986$, $\sigma = 88.357$). A task consisted of showing two sets of icons at a time - a set of manually assigned icons, and an automatically generated set from our algorithm. Participants were asked to select an icon set based on how well it represents the labels by choosing one of three choices - The left set, right set, or both sets. The left-right order of the manual and automatic icons was randomized. Subjects were paid $0.05 per task and the average time for completing each task was 20.6 seconds. In total, we recruited 89 different MTurk workers. 3 different MTurk workers performed each of the tasks. Each subject could only perform a task once, but nothing stopped the same subject from performing all the tasks of an experiment. In order to avoid biasing due to the visualization itself, we just showed a list of icons and their associated labels.

Results and Discussion

For each of the trials, we chose the selection that got the majority of the 3 votes. Any trial that had a tie of one vote for each of the three selections, was not considered in the analysis. 463 out of the 492 trials were identified to have a majority vote in selection. The excluded 29 trials were 7 'People' and 22 'Entertainment' icon sets. Out of these trials, 95 (20.52%) of them had a preference for user-generated icons, 140 (30.24%) were for auto-generated icons, and 228 (49.24%) had both the sets as a preferred choice (Table 3).We ran a repeated measures ANOVA on the user's selection with the method (choosing user generated or auto-generated) as the fixed effect and participant and icon sets as random effects. There was a signcant effect of the methods on the selection ($p-value = 0.03$). It should be noted that 79.48% of the workbooks had a combined preference of auto-generated icons or both.

Analyzing the results by category, we observe that the MTurk study participants preferred either the auto-generated icon set or both sets for a majority of the 'Sports' (96.72%), 'Places' (96%), 'Companies' (88.14%), 'Consumer products' (85%), and 'People' (59.68%) workbooks. On the contrary, participants preferred the user-generated icon sets for the 'Health and science' (56.25%) and and 'Entertainment' (54.72%) of the 463 workbooks. Table 4 shows the study participants' preferences by category.

User-generated	Auto-generated	Both
20.52%	30.24%	49.24%

Table 3. MTurk worker's preference for icons

Category	User-generated	Auto-generated	Both
Sports	4	52	66
Companies	7	3	49
Places	3	24	48
People	25	17	20
Consumer products	9	26	25
Health and science	18	8	6
Entertainment	29	10	14

Table 4. MTurk worker's preference by category. Here, the numbers reflect actual count.

Figure 4 shows the distribution of user preferences based on the cardinality of the icons. An overall ANOVA indicates that the cardinality is a significant factor for describing the user preference ($p-value = 0.003$). Further, running individual ANOVAs at cardinality increments of 50, showed very significant factors ($p-value < 0.001$) for cardinality in the range of 75 and 250, thus supporting our hypothesis.

While exploring the findings of the study, we made several interesting observations:

- MTurkers preferred icons with a white or translucent background. For example, in Table 5b (last page) , the user-generated logos for *Anaheim Ducks* and the *Boston Bruins*
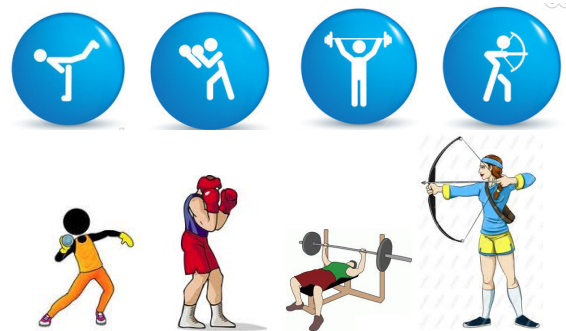

Figure 5. MTurkers preferred user-generated icon sets as opposed to individual icons, even though they had a similar look-and-feel. Top: An example of an icon set that was user-generated in one of the Tableau workbooks with a cardinality of 40. Bottom: The corresponding automatically generated icons.

are not on a white or translucent background. The rest of the icons look similar to that of the automatic algorithm. This behavior was prevalent in the study results.

- MTurkers preferred icons that have a handcrafted look to them once they know that they are meaningful. For example, in Table 5f, even though the user-generated and auto-generated flag symbols were similar, the former was more appealing. A future direction is to look at some best practice design techniques used for generating icons such as adding stylized filters and masks,

- We particularly looked at several of the workbooks with large cardinalities of data (≥ 50), where user-generated icons were preferred. Out of 143 workbooks with ≥ 50 icons, participants preferred user-generated icon sets for 32 of them. A majority of these icons were part of an icon set as shown in Table 5. We would like to explore ways in which our algorithm could retrieve semantically relevant icon sets from the Internet, that match the number of items they need to represent.

LIMITATIONS AND FUTURE WORK

We found that the algorithm performs well for popular sports teams, company logos, states and country flags, and prominent people. While we found it promising that users found 79.48% of the automatic icon assignments to be at least as good as the human selected icons, there are limitations in the algorithm.

Domain-specific content: As we are leveraging Google Image Search for the imagery, our technique is limited when the categorical data is very domain specific. For example, under the 'Science' category, there was a workbook showing tooth decay data over time. Each of the teeth (*e.g.* bottom-left 1st molar, top-left canine) has a corresponding icon that was found to be much superior than the computer generated icon, even though the user-generated icons were all not distinctive from one another (Figure 6). We speculate that the user-generated icons looked more uniform and professional that the auto-generated ones, and were preferred. Allowing the user to indicate domain specific image databases in the tool, could help improve the quality of images retrieved.

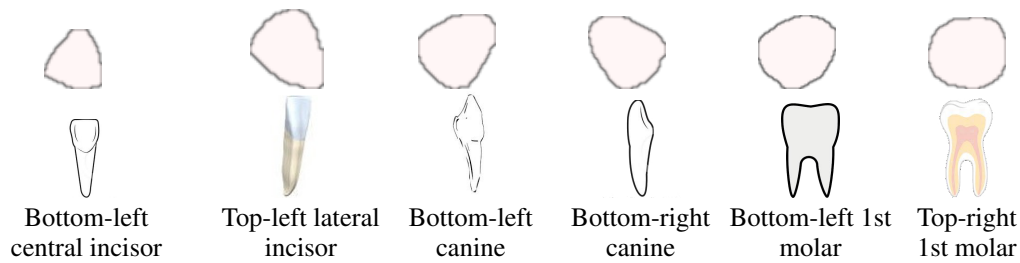| Bottom-left central incisor | Top-left lateral incisor | Bottom-left canine | Bottom-right canine | Bottom-left 1st molar | Top-right 1st molar |

Figure 6. Top row: User-generated icon set for different teeth types. Bottom row: Automatically generated icon set for different teeth types. The automatic algorithm does not perform as well as the user-generated icon sets for domain-specific categorical information. While, the automatic algorithm attempts to retrieve icons that are visually distinctive from one another (e.g. bottom-left and bottom-right canines), this is not always preferable.

Image discriminability: The algorithm relies on the metadata associated with the images to retrieve relevant results, and cluster similar, yet visually distinct icons. The hyperlink sources and the textual content on the webpage associated with the images, are reasonable enough for ensuring discriminability with light-weight computation. However, this information could be somewhat limiting as the algorithm does not take into account the actual visual features in the images. There are cases where images from different sources, could fetch similar icons. Figure 7 shows icons for various mobile phone models that appear similar. One possible improvement to these would be to use generic phone icons that are distinctive, and employ a visual language that visually classifies families of icons so that the Samsung device icons appear to be part of the same class of devices, yet different from the Motorola device icons for example.

| Samsung Galaxy Tablet | Motorola Droid X | HTC ADR6300 | Amazon Kindle Fire |

Figure 7. Image discriminability can be an issue for certain data sets. Here, the icons for each of the mobile devices look very similar even though the images are from different sources.

Lack of sufficient context: Tableau workbooks using acronyms and abbreviations do not provide enough information about the categorical fields, leading to erroneous images. For example, a workbook containing abbreviated football terms such as 'Away Loss (AL)', 'Away Win (AW)', 'Extra Time (ET)', and 'Goalkeeper (GK)' had a more compelling user generated icon set. Providing a user affordance in the tool for allowing the user to expand upon these terms could help improve the performance of the algorithm.

False positives: Given that we used Tableau Public workbooks that often have repeated datasets for which users have icons, we do have information about icons repeatedly wrongly identified. For example, flags for 'Republic of Congo and 'Democratic Republic of Congo are often switched due to artifacts of Google Search. This issue can be addressed in the user experience, by allowing the user to correct the erroneous result in the visualization tool, and employing machine learning techniques to re-train the algorithm

for correcting the mistake. Other possibilities include crowd-sourced approaches for Tableau Public users to upload and correct icon sets.

CONCLUSION

This paper describes an algorithm that automatically maps categorical data to icons that can be used in visualizations to show the semantics of marks. Icons can be particularly effective in visualization when the domains are larger than 20 items, which is often the case. A key advantage of an automatic algorithm is that users can stay in the flow of their visual analysis rather than being interrupted to assign icons to the items in a domain. The algorithm is based a set of powerful natural language processing (NLP) techniques. The algorithm also takes into account the inherent design and functional characterizes of icon encodings to provide meaningful images to users. We described how to expand queries based on the items in domains to also include terms that were symbolic in nature. We evaluated the algorithm using 492 Tableau Public workbooks where authors had created public icon encodings. 79.48% of the evaluations found that the automatic icon assignment was at least as good as the human selected icon. We believe that there are many views in Tableau Public that would be improved by the use of icon encodings. Our hunch is that the authors did not use icons because it is hard work to manually select icons for a domain. An automatic algorithm like the one described in this paper has the potential to significantly improve the views that people are using to see and understand their data, which will help to make the world a better place.

ACKNOWLEDGEMENTS

REFERENCES

1. Björk, S., Holmquist, L. E., Redström, J., Bretan, I., Danielsson, R., Karlgren, J., and Franzén, K. West: a web browser for small terminals. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology (UIST)*, ACM (New York, NY, USA, 1999), 187–196.

2. Blattner, M. M., Sumikawa, D. A., and Greenberg, R. M. Earcons and icons: Their structure and common

design principles (abstract only). *SIGCHI Bulletin 21*, 1 (Aug. 1989), 123–124.

3. Cleveland, W. S., and McGill, R. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association 79*, 387 (1984), pp. 531–554.

4. Coren, S., and Girgus, J. Principles of perceptual organization and spatial distortion: The gestalt illusions. In *Journal of Experimental Psychology: Human Perception and Performance*, vol. 6, Morgan Kaufmann (1980), 404–412.

5. Familant, M. E., and Detweiler, M. C. Iconic reference: Evolving perspectives and an organizing framework. vol. 39, Academic Press Ltd. (London, UK, UK, Nov. 1993), 705–728.

6. Fellbaum, C., Ed. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

7. Gittins, D. Icon-based human-computer interaction. vol. 24, Academic Press Ltd. (London, UK, UK, June 1986), 519–543.

8. Healey, C. G., Booth, K. S., and Enns, J. T. Visualizing real-time multivariate data using preattentive processing. *ACM Transactions on Modeling and Computer Simulation 5*, 3 (July 1995), 190–221.

9. Henry, T. R., and Hudson, S. E. Multidimensional icons. vol. 9, ACM (New York, NY, USA, 1990), 133–137.

10. Kosslyn, S. M. Graphics and human information processing: A review of five books. vol. 80, Journal of the American Statistical Association (1985), 499–512.

11. Lewis, J. P., Rosenholtz, R., Fong, N., and Neumann, U. Visualids: Automatic distinctive icons for desktop interfaces. In *ACM Special Interest Group on Graphics and Interactive Techniques (SIGGRAPH)*, ACM (New York, NY, USA, 2004), 416–423.

12. Li, Y., Bandar, Z. A., and McLean, D. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering 15*, 4 (July 2003), 871–882.

13. Mackinlay, J. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics (TOG) 5*, 2 (Apr. 1986), 110–141.

14. Marcus, A. Icons, symbols, and signs: Visible languages to facilitate communication. *Interactions 10*, 3 (May 2003), 37–43.

15. Nowell, L. T. Graphical encoding for information visualization: Using icon color, shape, and size to convey nominal and quantitative data. Virginia Polytechnic Institute and State University (1997).

16. Resnik, P. Using information content to evaluate semantic similarity in a taxonomy. In *In Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95*, Morgan Kaufmann (1995), 448–453.

17. Rogers, Y. Icons at the interface: Their usefulness. vol. 1, Elsevier Science Inc. (New York, NY, USA, Apr. 1989), 105–117.

18. Rother, C., and Kolmogorov, V.and Blake, A. "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (TOG) 23*, 3 (Aug. 2004), 309–314.

19. Setlur, V., Albrecht-Buehler, C., Gooch, A., Rossoff, S., and Gooch, B. Semanticons: Visual metaphors as file icons. vol. 24 (2005), 647–656.

20. Shneiderman, B. *Designing the User Interface: Strategies for Effective Human-computer Interaction.* Addison-Wesley Longman Publishing Company, Inc., Boston, MA, USA, 1986.

21. Smith, L., and Thomas, D. Color versus shape coding in information displays. vol. 48, Journal of Applied Psychology (1964), 137– 146.

22. Wobbrock, J. O., Forlizzi, J., Hudson, S. E., and Myers, B. A. Webthumb: Interaction techniques for small-screen browsers. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology (UIST)*, ACM (New York, NY, USA, 2002), 205–208.

23. Woodruff, A., Faulring, A., Rosenholtz, R., Morrsion, J., and Pirolli, P. Using thumbnails to search the web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, ACM (New York, NY, USA, 2001), 198–205.

24. Wu, Z., and Palmer, M. Verbs semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, ACL '94, Association for Computational Linguistics (Stroudsburg, PA, USA, 1994), 133–138.

User generated icons	Auto generated icons

a) **Soccer clubs**: Arsenal, Barcelona, Liverpool, Reading and Southampton. **User Preference**: both. Cardinality =21.

b) **NHL Teams**: Anaheim Ducks, Boston Bruins, Buffalo Sabres, Calgary Flames, and Carolina Hurricanes. **User Preference**: auto generated. Cardinality = 30.

c) **Halloween candy**: 3 Musketeers, Butterfinger, Candy corn, KitKat and Nerds. **User Preference**: auto generated. Cardinality = 76.

d) **Companies**: Angie's List, Groupon, LinkedIn, Pandora, and Zillow. **User Preference**: both. Cardinality = 103.

e) **Pokemon characters**: Camerupt, Darmanitan, Golurk, Jigglypuff and Lilligant. **User Preference**: both. Cardinality = 254.

f) **Country flags**: Australia, Belgium, CZech Republic, Denmark and Germany. **User Preference**: user generated. Cardinality =10.

Table 5. *A comparison between user-generated icons and ones generated by our method. Left set: a sub-set of user-generated icons. Right set: a sub-set of auto-generated icons.*

550