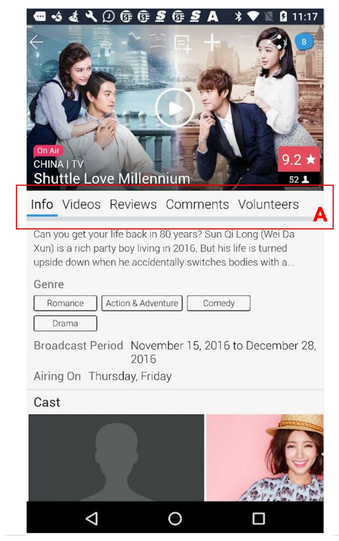
Towards Semantically-Aware UI Design Tools: Design, Implementation, and Evaluation of Semantic Grouping Guidelines

Peitong Duan (peitongd@berkeley.edu), Bjoern Hartmann, Karina Nguyen, Yang Li, Marti Hearst, Meredith Morris

Motivation

A coherent semantic structure, where semantically-related elements are appropriately grouped, is critical for proper understanding of an user interface (UI). To date, computational UI design tools can automate evaluation and optimization of visual aspects, such as spatial layout and aesthetics. Ideally, these design tools should also help designers establish coherent semantic grouping.



Example of an UI with poor semantic grouping from the RICO dataset. Section A contains the tab “Volunteers", which is unrelated to the other tabs about the TV series.

Approach

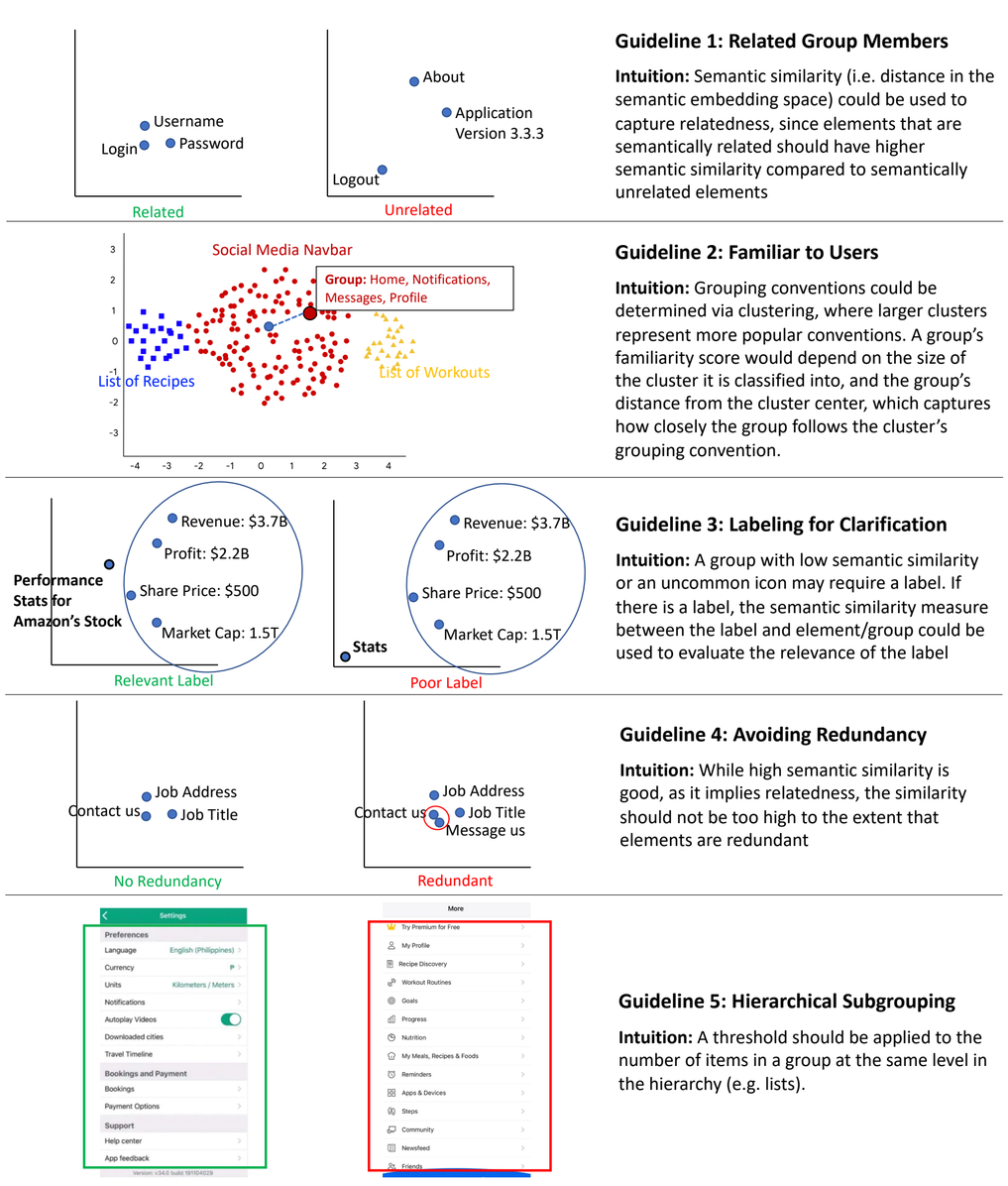
To work towards this, we contribute five semantic grouping guidelines that capture how human designers think about semantic grouping and are amenable to implementation in design tools. We then demonstrate the guidelines’ use for building systems by implementing a set of novel computational metrics that could evaluate the semantic coherence of UIs.

Guideline Generation and Evaluation

The process we followed to obtain the 5 semantic grouping guidelines. The concise version of the guidelines can be found under ”Guidelines (Concise Version)”.

Metrics

To demonstrate the guidelines’ implementability, we built computational metrics that check for application or violation of each guideline. These metrics are based on similarity comparisons and clustering of BERT-based embeddings of semantic data.



Guideline 1: Related Group Members

Intuition: Semantic similarity (i.e. distance in the semantic embedding space) could be used to capture relatedness, since elements that are semantically related should have higher semantic similarity compared to semantically unrelated elements

Guideline 2: Familiar to Users

Intuition: Grouping conventions could be determined via clustering, where larger clusters represent more popular conventions. A group’s familiarity score would depend on the size of the cluster it is classified into, and the group’s distance from the cluster center, which captures how closely the group follows the cluster’s grouping convention.

Guideline 3: Labeling for Clarification

Intuition: A group with low semantic similarity or an uncommon icon may require a label. If there is a label, the semantic similarity measure between the label and element/group could be used to evaluate the relevance of the label

Guideline 4: Avoiding Redundancy

Intuition: While high semantic similarity is good, as it implies relatedness, the similarity should not be too high to the extent that elements are redundant

Guideline 5: Hierarchical Subgrouping

Intuition: A threshold should be applied to the number of items in a group at the same level in the hierarchy (e.g. lists).

An explanation and visualization for the intuition behind each guideline metric.

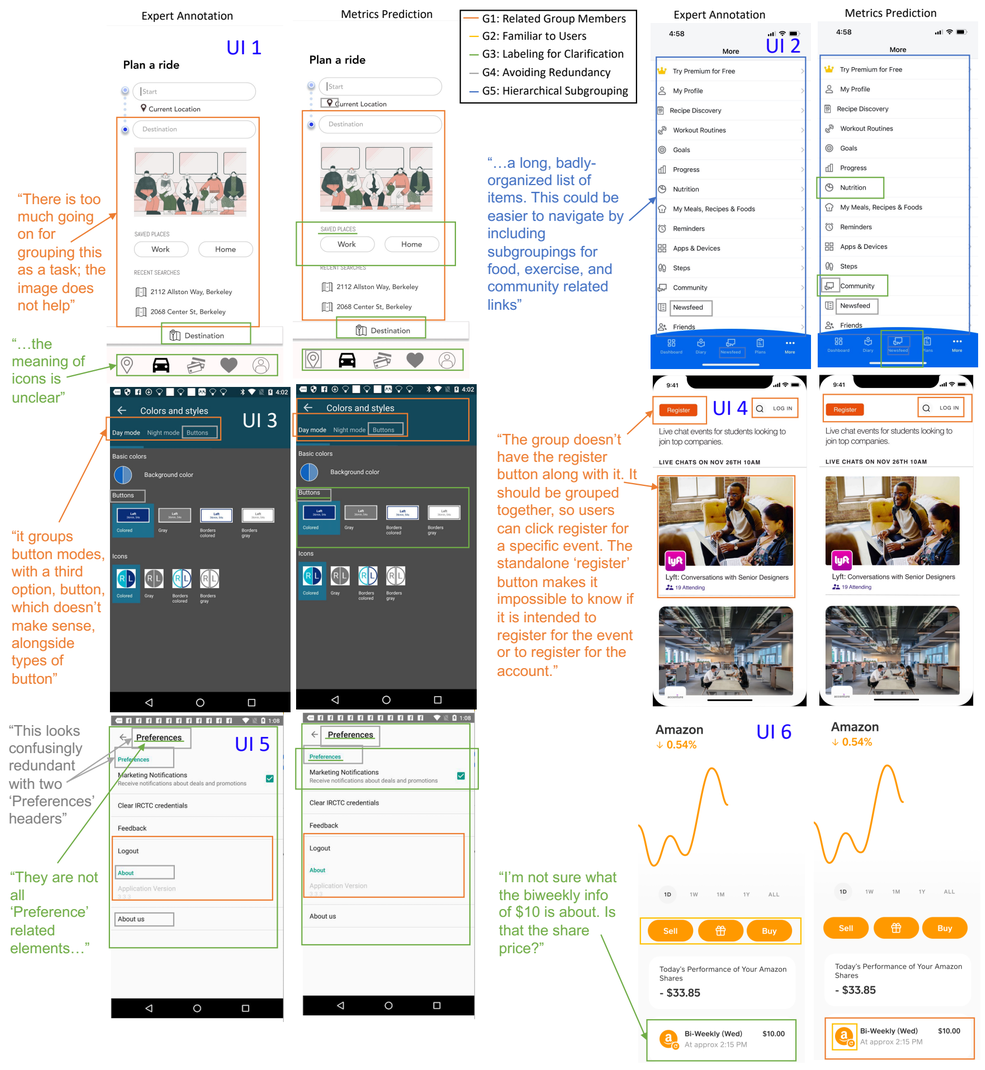
Guidelines (Concise Version)

Guideline Title	Guideline Text (Concise Version)
1. Related Group Members	There should be a clear relationship amongst members in a group. The members could be related by a task, direct effect, category, object, or time (with overlaps allowed). Examples for each option are as follows: a. *Task:* all the input fields and widgets for booking a flight b. *Direct effect:* tabs that separate and control the display of the page content should be grouped with the content c. *Category:* different types of rooms in a home monitoring UI d. *Object:* a profile page with details about a person e. *Time:* this month’s transactions This guideline aims to help users understand the purpose of the group.
2. Familiar to Users	The grouping should be familiar to users. This can be achieved by following established design conventions.
3. Labeling for Clarification	Labels can be used to explain the meaning of an element (a), the meaning of a group of elements (b), and/or the meaning of the UI grouping organization (c). This is especially useful when the purpose of the grouping is not clear and for helping users make sense of apps from less common categories.
4. Avoiding Redundancy	A group should not contain members with redundant functionalities. The purpose is to reduce user uncertainty about each redundant member’s purpose.
5. Hierarchical Subgrouping for Large Groups	A large group (containing many members) should be subgrouped, and there should be a clear hierarchy that shows the subgrouping organization. This makes it easier for users to comprehend these large groups.

Concise versions of the 5 semantic grouping guidelines we obtained from the process shown in “Guideline Generation and Evaluation”.

Validation of Metrics

We compared the computational metrics against a set of violations manually identified by designers, who were given our guidelines. We achieved a F1 score of **0.882** when validated on this ground truth dataset annotated by experts.



Comparison of the guideline violations found by experts during the heuristic evaluation (“Expert Annotation") with those identified our computational metrics (“Metrics Prediction"). A quote from an expert explaining the semantic grouping issue is provided for some violations.

Large Scale Analysis of Real World UIs

We ran our metrics on a large dataset of over 9.5k real-world UIs (taken from RICO) to determine the frequency of groups that would be flagged as having poor semantic grouping by our guidelines. Results showed that about **21.4 percent** of UIs had at least one guideline violation.

Limitations

1. Metrics rely heavily on the accuracy of text-based semantic descriptions of UI elements and the grouping structure
2. Validation of metrics (by experts) was conducted on a small set of 6 UIs
3. Metrics are applied to the grouping structure that is implemented by designers (e.g. Android View Hierarchy) and cannot detect issues when the implemented grouping does not match the grouping structure perceived by users.

Future Work

1. Train data-driven models on noisy input data to be robust against missing or incorrect semantic annotations and erroneous grouping structure
2. Apply existing models to extract users’ perceived grouping structure from UI screenshots to address mismatches between implemented and perceived grouping
3. Develop a numerical semantic coherence score, which could be used in optimization algorithms to automate improvements to a grouping structure