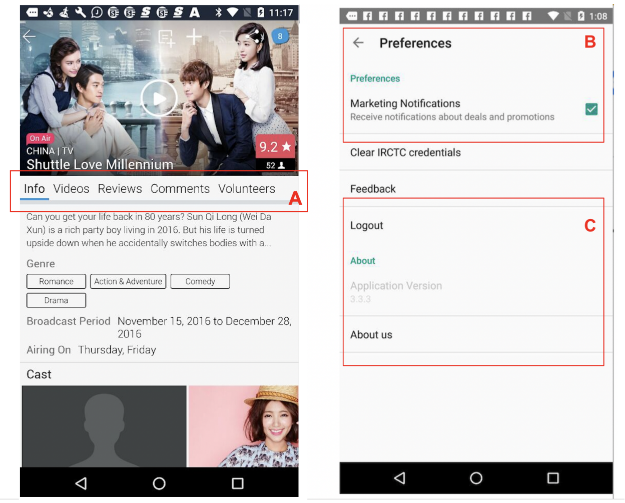
Towards Semantically-Aware UI Design Tools: Design, Implementation and Evaluation of Semantic Grouping Guidelines

Peitong Duan [1] **Bjoern Hartmann** [1] **Karina Nguyen** [1] **Yang Li** [2] **Marti Hearst** [1] **Meredith Morris** [2]

Abstract

A coherent semantic structure, where semantically-related elements are appropriately grouped, is critical for proper understanding of a UI. Ideally, UI design tools should help designers establish coherent semantic grouping. 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. They were obtained from empirical observations on existing UIs, a literature review, and iterative refinement with UI experts' feedback. We validated our guidelines through an expert review and heuristic evaluation; results indicate these guidelines capture valuable information about semantic structure. We demonstrate the guidelines' use for building systems by implementing a set of computational metrics. These metrics detected many of the same severe issues that human design experts marked in a comparative study. Running our metrics on a larger UI dataset suggests many real UIs exhibit grouping violations.



Can you get your life back in 80 years? Sun Qi Long (Wei Da Xun) is a rich party boy living in 2016. But his life is turned upside down when he accidentally switches bodies with a...

Broadcast Period November 15, 2016 to December 28, 2016

Airing On Thursday, Friday

Marketing Notifications ☑

Receive notifications about deals and promotions

Application Version

3.3.3

Figure 1. Examples of UIs with poor semantic grouping from the RICO dataset (Deka et al., 2017). The left screenshot's Section A contains the tab "Volunteers", which is unrelated to the other tabs about the TV series. In the right UI, the "Preferences" label is repeated in Section B, making the purpose of the UI page (titled "Preferences") and the "Preferences" subsection unclear. In Section C, the "Logout" button and the "About" details are grouped, but unrelated. Also, there is a separate "About us" item below this group, which should be grouped with the "About" details instead, as they are more related. Since these UIs come from the Google Play store, their designers were probably unaware of these grouping errors. Our guidelines and metrics address these semantic grouping errors.

1. Introduction

User interface (UI) design tools are transitioning from a paradigm where designers manually specify everything to one where tools increasingly incorporate automation to evaluate, revise, or generate aspects of UIs. To date, computational user interface design tools can automate evaluation and optimization of visual aspects, such as spatial layout (Duan et al., 2020; Todi et al., 2016; Oulasvirta et al., 2018; Quiroz et al., 2007) and aesthetics (Todi et al., 2016; Oulasvirta et al., 2018; Quiroz et al., 2007). Facets of the design that require semantic understanding have been left to the designer to plan and refine (Swearngin et al., 2020). One such characteristic is the grouping of the elements in the interface. For a grouping to make sense to users, the combination of its members' semantics (i.e. functionality, content, or purpose) must be coherent, or logical and understandable. We refer to this way of thinking about grouping, based on the functionality, content, or purpose of its members, as *semantic grouping*.

Highly coherent semantic grouping is important, as studies suggested that the quality of semantic grouping affects users' understanding of the interface, implying that users capitalize on the semantic structure when trying to make sense of the UI (Halverson & Hornof, 2008). Furthermore, studies found that users could complete tasks more efficiently (Card, 1982;

[1] University of California, Berkeley [2] Google. Correspondence to: Peitong Duan <peitongd@berkeley.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

Bailly et al., 2014; Halverson & Hornof, 2008) and gave higher quality ratings (Kotval & Goldberg, 1998) to designs where the grouped items were semantically coherent. However, a study (Chi et al., 2003) found that designers have difficulty constructing an accurate semantic structure for their UI. All this suggests that automated evaluation, optimization, and generation of semantically coherent grouping is a valuable addition to design tools.

Creating these more powerful design tools requires a concrete set of design guidelines that both capture how human experts think about grouping and can be operationalized and implemented in software. While there exist design guidelines that touch on semantic grouping in the literature today, they do so at a high level, without specifying how to implement the advice. For example, Nielsen suggests "making information appear in a natural and logical order" (Nielsen & Molich, 1990), and Brown's principle of focused navigation exhorts designers to "avoid mixing apples and oranges." (Brown, 2010). We bridge this gap by defining five guidelines that capture best practices for semantic grouping and are more specific, which provides a clear path towards operationalization. These guidelines were refined through several rounds of feedback from design experts and then validated with an expert review. To demonstrate that these guidelines are implementable, 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, and are the first metrics to automatically evaluate the semantic grouping quality of UIs. We compared these computational metrics against a set of violations manually generated 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.

Finally, we computationally applied our guidelines on a set of UIs from the RICO dataset via our metrics. The metrics flagged potential violations in 21.4% of these UIs, which implies that automated tools built on these metrics could suggest areas for redesign in a significant number of real-world UIs. To summarize, our contributions include:

- Five guidelines for semantic grouping that were extracted from a literature review, empirical observations, and revised through multiple rounds of expert feedback. These guidelines are specific and operational, providing the basis for developing computational metrics for semantic grouping quality. These guidelines are in Figure 3 and the Appendix (Section A).
- Two expert studies; one confirmed these guidelines capture best practices for semantic grouping, and the other resulted in a set of guideline-based expert annotations in a set of UIs
- Demonstration of these guidelines' implementability by building computational metrics that check for guideline application or violation; this is the first set of metrics that could automatically assess the semantic grouping of UIs. These metrics achieved an F1 score of 0.882 on the expert-annotated ground truth dataset.
- An analysis of how many UIs "in the wild" have potential semantic grouping violations that are detectable by our metrics.

2. Related Work

2.1. UI Design Guidelines

UI design guidelines and heuristics play a direct role in the creation of new interfaces (Höök & Löwgren, 2012) and have been developed and studied extensively. Nielsen's ten heuristics (Nielsen & Molich, 1990) are frequently used for heuristic evaluation, and Van Duyne et. al. researched and extracted customer-focused design patterns (Van Duyne et al., 2007). Similarly, Ribeiro (Ribeiro, 2012) and Hoober et. al. (Hoober & Berkman, 2011) studied and extracted design patterns for mobile UIs, which have a different set of constraints compared to websites (Ribeiro, 2012). Some of these guidelines touch on semantic grouping, but do so at a high level, without specifying how to implement the advice (Nielsen & Molich, 1990; Brown, 2010). Our work aims to turn this high-level advice into detailed guidelines that are automatable for design tools.

Due to the recent rise in AI-based systems, Amershi et al. generated 18 guidelines for human-AI interaction (Amershi et al., 2019). Their process is instructive for us: they generated candidate guidelines and consolidated them using grounded theory coding (Martin & Turner, 1986) and affinity diagramming (Scupin, 1997). They then ran heuristic evaluations with their guidelines and conducted an expert review to finalize their guidelines. We followed a similar iterative process to extract our semantic grouping guidelines.

2.2. Semantic Consistency

The effects of semantic consistency have been studied in real-world scenes (Henderson et al., 1999), vertical menu interfaces (Brumby & Zhuang, 2015; Card, 1982; Bailly et al., 2014), visual layouts of words (Halverson & Hornof, 2008), and widget toolbars (Kotval & Goldberg, 1998). Kotval et. al., tracked users' eye movements to measure the amount of visual processing during search across different widget grouping strategies (Kotval & Goldberg, 1998). They found that grouping widgets by functionality had the highest interface quality score and required the least amount of visual processing. They also found that the interface quality rating decreased and visual processing time increased as the grouping became less semantically related. We built off this prior work by extending the set of guidelines for grouping UI elements beyond just relatedness in functionality. Our study also generalizes to groups with different element types (e.g. a list of articles with images and text), as well as groupings of higher levels (i.e. a group of groups).

2.3. Computational UI Semantics

Prior work on computational semantic understanding of UIs have focused on elements (Liu et al., 2018; Li et al., 2020b; He et al., 2021) or entire UI screens (He et al., 2021; Li et al., 2021; Leiva et al., 2020). Liu et. al. predicted and annotated the UI elements in the RICO dataset(Deka et al., 2017) with their functional semantics (login, retry, etc.) (Liu et al., 2018). Li et. al. trained a model to generate descriptions of widget functionalities by training a model on crowdsourced widget captions(Li et al., 2020b). At the screen level, Jia et. al. generated embeddings of the entire UI screen based on the semantics of their elements, and then applied these screen embeddings for downstream tasks, like finding similar UI screens to an input screen (Li et al., 2021). Furthermore, He et. al. developed a novel way to extract the functionalities of widgets by looking at the interaction flow they are involved in (He et al., 2021). The authors then developed an NLP-inspired method to embed the UI screen by modeling UI elements as tokens and UI screens as sentences. Fu et. al. followed a similar approach to model and embed UI screens as sentences, except they extracted the semantic annotations of elements directly from the screenshot (Fu et al., 2021). However, these black box screen embeddings do not provide insights on how to improve the semantic organization within UI screens.

3. Semantic Grouping

We formally define semantic grouping and explain other relevant terminology in this section. Since grouping in a UI occurs at multiple levels and forms a hierarchy (enforced by the UI layout tree), as shown in Figure 2, we formulate the definition of semantic grouping to be applicable to groups at all levels of the hierarchy, in order to account for all groups in the UI. Semantic grouping is defined as grouping based on the meaning of its members. In the context of UIs, “meaning” refers to functionality, content, or purpose. “Functionality” and “content” apply to low level groups in the hierarchy, which consists of elements only, and “purpose” applies to high level groups that consist of smaller groups. Namely, for low level groups (e.g. Figure 2 Box A), we consider the functionality of their interactive elements (buttons, icons, etc.) and the content of their non-interactive elements (images, text labels, text descriptions) when determining the grouping semantics. For high level groups (e.g. Figure 2 Box B), we consider the purpose of each of their subgroup members.

Figure 2 includes a UI and its corresponding grouping layout tree. Group A in the figure contains only UI elements, so we would consider the functionality or content of its element members, which include ”log in”, ”I’ve forgotten my password”, and ”log in with Facebook”, etc. Group B is a higher level group, with Group A as one of its members. We would consider the purpose of Group A when evaluating the semantics of Group B, which would be something like ”multiple options to log in”, along with the content or functionality of Group B’s element members, such as the text ”To improve your experience, we recommend ...”. Table 2 (Appendix Section A) contains definitions of other important terminology we use in the guidelines and the remainder of this paper.

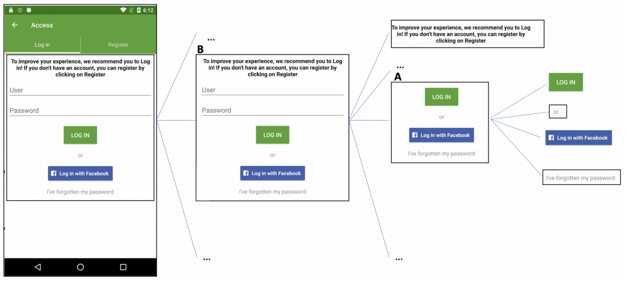
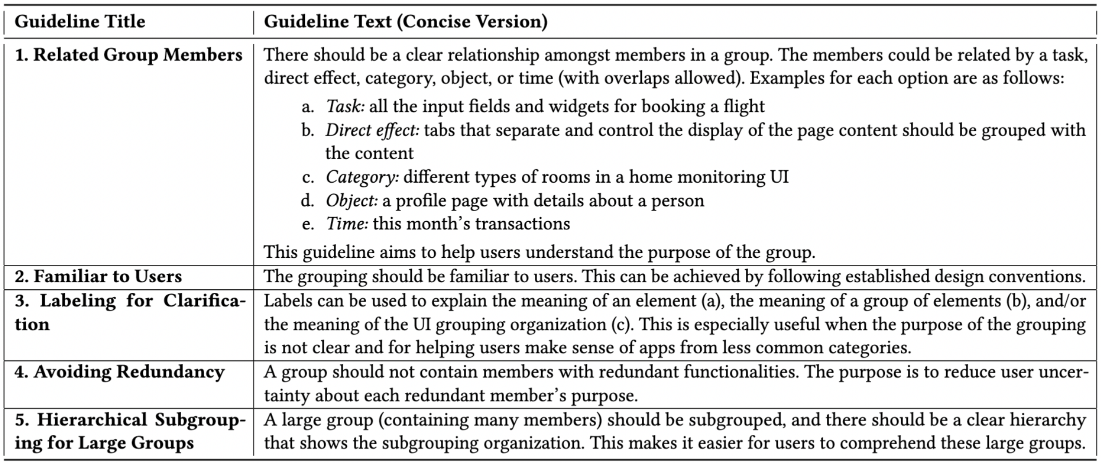

6:12
Access
Log in
Register
To improve your experience, we recommend you to Log in! If you don't have an account, you can register by clicking on Register
User
Password
LOG IN
or
Log in with Facebook
I've forgotten my password
...
B
A
...
...

Figure 2. A mobile UI and its corresponding layout grouping tree. Group A is a low level group containing only UI elements, whereas Group B is a higher level group containing Group A. The entire UI screen is the root node.

4. Guideline Generation, Review, and Evaluation

We first generated an initial list of guidelines from a literature review, supplemented with empirical observations we made on the semantic grouping of a set of example mobile user interfaces. We then revised this initial set of guidelines through several rounds of feedback from design experts, which resulted in the 5 guidelines. Each guideline has a concise and detailed version. Figure 3 contains the concise version and Appendix Section A contains both. Section B (Appendix) contains details of our method and results.

To ensure that our guidelines form an accurate basis for building metrics, we conducted an expert review to 1) evaluate how clear (i.e. understandable) the guidelines are, 2) assess how important each guideline is 1) is to ensure that the guidelines can be easily understood for future development of metrics. 2) is to ensure that the guidelines recommend best practices for semantic grouping that the experts believe are important to follow. We recruited 8 experts from a large technology company [name removed for anonymous review]; they specialize in either design or UX research.



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.

Figure 3. Concise versions of the 5 semantic grouping guidelines; their corresponding detailed versions are in the Appendix (Section A).

Overall, participants found the guidelines clear and highly important to consider when grouping elements in a UI, confirming that our guidelines recommend good practices for semantic grouping and are helpful towards future development of metrics. Section C (Appendix) contains detailed results. After some minor updates based on the experts' feedback, we arrived at the final version of these guidelines shown in Figure 3 and the Appendix (Section A).

We proceeded to use these guidelines in a heuristic evaluation with design experts. The goal was to collect expert-identified guideline violations, which would serve as ground truth for validating our computational metrics and subsequent analyses. We recruited 9 experts (7 designers, 2 UX researchers). The participants used our guidelines to identify violations in 6 distinct UIs with 16 known guideline violations. We decided on 6 UIs, so that the participants could finish within the 90-minute timeframe set aside for this study. Furthermore, these 6 distinct UIs consisted of 3 real-world UIs (from RICO) and 3 author-constructed UIs. We used author-constructed UIs so we could manually insert guideline violations, which would ensure coverage of all guidelines in the set of violations. During the study, participants were asked to explain the semantic grouping issue for each violation they identified, and also provide a usability severity rating (Nielsen, 2006) ranging from from 0 (nonissue) to 4 (usability catastrophe).

Fifteen out of the 16 known guideline violations were found by at least one expert. Figure 5 shows an example UI from the evaluation with all the guideline violations found (under ”Expert Annotation”), along with a representative quote describing the violation. Figure 8 (Appendix) shows the violations found for all UIs. Detailed results are available in Appendix Section D.

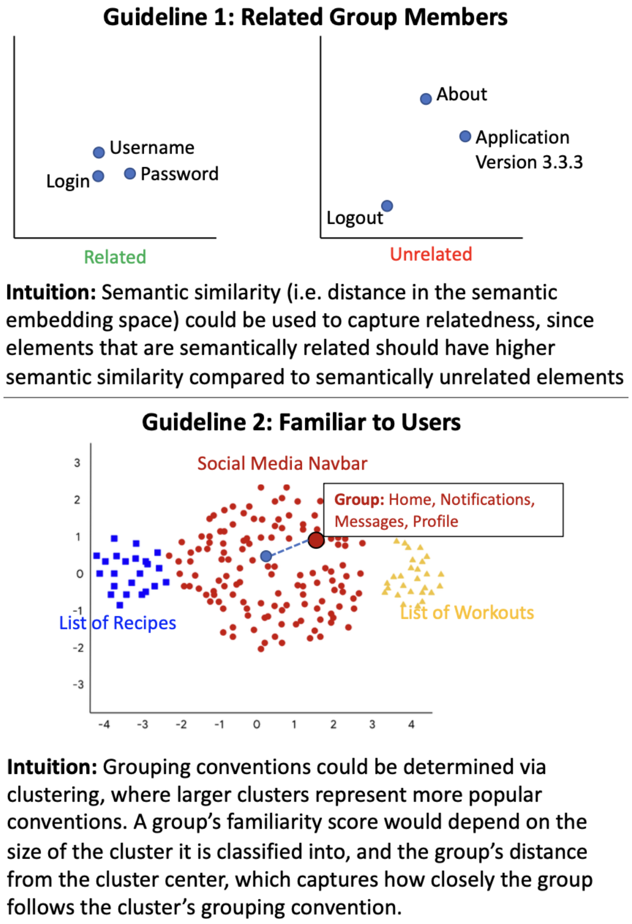
5. Computational Metrics

Given the definitions of the five expert-validated design guidelines above, we implemented computational metrics that check for the compliance or violation of each guideline. We describe the metrics for each guideline in this Section. Figure 4 contains a visualization and explanation for the intuition behind guidelines 1 and 2, and Figure 10 (Appendix) visualizations the intuition for all guidelines.

We do not claim optimality of these metrics, but regard them as a step towards the goal of semantically-aware tools. These metrics require semantic descriptions for each element in the user interface, which could be provided by UI designers at creation time or automatically derived (from the elements' text labels, etc). Semantic information for groups are automatically computed from the semantic data of its members, so only semantic descriptions of individual UI elements are required. For each guideline, its corresponding metric takes as input a group (with text-based semantic descriptions for all its element members) and classifies whether or not the guideline is being followed.

5.1. Guideline 1: Related Group Members

UI elements that are semantically related (by category, task, etc) would have semantic descriptions that are closer in meaning, compared to semantically unrelated elements. As such, we applied SentenceBERT (Reimers & Gurevych, 2019) to encode text-based semantic descriptions of elements, and then computed cosine similarity to measure the distance between embeddings. SentenceBERT embeddings are semantically meaningful in that embeddings that are more similar in meaning are closer in distance(Reimers & Gurevych, 2019). We also fine-tuned the SentenceBERT model with widget captions data from (Li et al., 2020b) to better capture semantic meaning in the UI space. Since


Guideline 1: Related Group Members
Username
Login
Password
Related
About
Application Version 3.3.3
Logout
Unrelated
Guideline 2: Familiar to Users
Social Media Navbar
Group: Home, Notifications, Messages, Profile
List of Recipes
List of Workouts
3
2
1
0
-1
-2
-3
-4
-3
-2
-1
0
1
2
3
4

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

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.

Figure 4. An explanation and visualization for the intuition behind guidelines 1 and 2. Figure 10 (Appendix) visualizations the intuition for all guidelines.

groups are represented by their members, we compute group embeddings by averaging the embeddings of its members, as a way to summarize them. Group embeddings are necessary when metrics are applied to higher level groups that contain smaller groups as members. Also, since cosine similarity is computed between pairs of embeddings, we calculated a group-wide similarity score by computing the cosine similarity between each member and a "summary" (i.e. average) embedding of the others and then averaging this score across all group members. This similarity comparison checks how well each group member relates to the rest of the group.

Finally, we apply a threshold on the group's cosine similarity score to determine whether or not its members are related. We computed the threshold via an ROC curve, which will be discussed in the evaluation section 6.1. We used this ROC curve approach to compute thresholds for all guideline metrics, except for for Guideline 5.

5.2. Guideline 2: Familiar to Users

Our approach to identify whether a group follows a common semantic grouping convention involves several stages. First, we used K-Means clustering to generate clusters, with each assumed to correspond to a specific grouping convention. We computed these clusters from the RICO dataset (Deka et al., 2017), where groups (at all levels) were extracted (from the android view hierarchies), embedded (with our fine-tuned sentenceBERT model), and then clustered. We determined the number clusters with the Elbow method and further tuned it with validation data. We computed the familiarity score for a group using the following equation:

$$\textit{familiarity score} = \frac{\textit{target cluster size}}{\textit{distance from cluster center}^2}$$

target cluster size is size of the cluster the group is mapped to and corresponds to the popularity of its grouping pattern. *distance from cluster center* refers to the distance of the group from the center of its mapped cluster and inversely corresponds to how closely the group follows the cluster's design convention. Combining these two values in the equation above captures how common the group's design pattern is and how closely the group follows this pattern, which are both factors that determine how familiar a grouping is to users. Figure 4 visualizes the intuition behind this metric.

5.3. Guideline 3: Labeling for Clarification

We assume the text labels for the element, group, and hierarchy are annotated in the input group. The text label could either be manually annotated or detected algorithmically in an input UI grouping tree, as we discuss in Section F.0.2 (Appendix). We compute the cosine similarity between the label embedding and the embedding of its corresponding element or group to measure its quality, since high quality labels should be semantically related to their respective element or group.

5.4. Guideline 4: Avoiding Redundancy

We check for redundancy by computing the cosine similarity between every pair of group members (aside from an element and its label). A similarity value close to 1.0 means the two members have nearly the same semantic meaning and are probably redundant.

5.5. Guideline 5: Hierarchical Subgrouping

We designate a group with more than 10 members at the same level in the hierarchy as a violation of this guideline, in line with (Bailly et al., 2014)'s finding that search time was significantly higher for unordered menus of that length (or longer) compared to semantically subgrouped menus of the same length.

6. Computational Metrics Evaluation

We first determined thresholds for binary classification of guideline compliance vs violation. To collect data to tune these thresholds, we hand-labeled a set of groups (with the guidelines they apply or violate) in example UIs retrieved from the Google Play Store and RICO (Deka et al., 2017). We then compared the violations detected by our metrics to violations that human design experts manually flagged during the heuristic evaluation.

6.1. Method

We selected a diverse set 24 mobile app screens and manually recorded groups. For each group, we added semantic descriptions for each of its element members, annotated any group or element labels, and recorded the guidelines it applied or violated. We recorded 300 groups total.

To determine the semantic description for each element, we used the content strings (of text elements), in-element text labels (e.g. “login” for login buttons), and for icons, we used corresponding semantic annotations from existing datasets (Liu et al., 2018; Li et al., 2020b). For image elements, we added our own descriptions of the content. If we felt that the text label for an element inadequately describe its functionality, we provided a more detailed semantic description.

We determined the thresholds for each guideline via an ROC curve. The optimal threshold is the number with the highest value for *True Positive Rate − False Positive Rate*. We used roughly 90 percent of the manually collected dataset for training and 10 percent for validation. The validation set was used for further fine-tuning of the thresholds and determining the number of clusters for Guideline 2’s metric.

There were significantly more examples of guideline applications than violations in the dataset, so we generated synthetic violation examples by randomly selecting a group member and replacing it with a randomly selected element from the entire dataset. However, all the violation examples in the validation set were real samples from the dataset.

6.2. Validation on Violations Found by Experts

We validated our metrics’ performance on a ground truth dataset by comparing their predicted violations with those identified by design experts during the heuristic evaluation. We used the corresponding threshold computed in Section 6.1 to predict each violation. We provided the semantic descriptions for each UI element in this set, following the same procedure described in Section 6.1.

6.2.1. Results

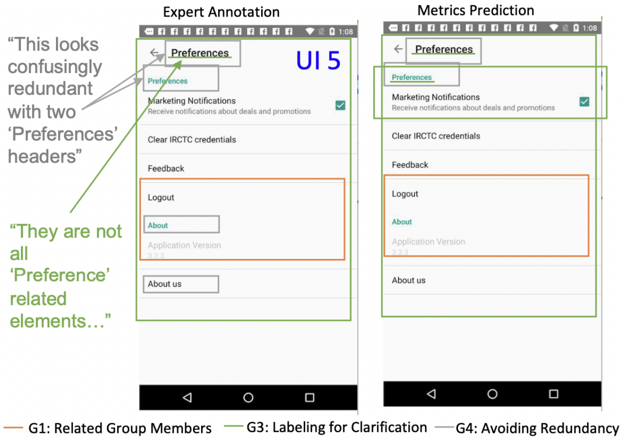

Expert Annotation
Metrics Prediction
“This looks confusingly redundant with two ‘Preferences’ headers”
“They are not all ‘Preference’ related elements…”
UI 5
1:08
Preferences
Preferences
Marketing Notifications
Receive notifications about deals and promotions
Clear IRCTC credentials
Feedback
Logout
About
Application Version
3.3.3
About us
G1: Related Group Members
G3: Labeling for Clarification
G4: Avoiding Redundancy

Figure 5. Comparison of the guideline violations found by experts from the heuristic evaluation (“Expert Annotation”) with those identified our metrics (“Metrics Prediction”) for a single UI. A quote from an expert explaining the semantic grouping issue is provided for some violations. The legend mapping the guideline violated to box color is at the bottom. Results for all 6 UIs can be found in Figure 8 (Appendix)

Figure 5 compares expert-identified UI violations (“Expert Annotation”) with violations predicted by our metrics (“Metrics Prediction”) for a single UI (UI 5), and Figure 8 (Appendix) contains the results for all UIs. While there is considerable overlap, our metrics missed two expert-identified violations: Guideline 4 in UI 5 (“About” and “About us” are redundant) and Guideline 2 in UI 6 (grouping the gift button with “Buy” and “Sell” is an unfamiliar grouping convention). However, the Guideline 4 violation is controversial – “About” and “About us” could be considered redundant; however, another interpretation may be that “About” refers to details regarding the app, whereas “About us” refers to details about the company or creators of the app.

In addition, the metrics detected 10 violations that were not identified by experts. While 2 of them are clearly false positives (“Saved Places” is a poor label for the “Home” and “Work” rideshare destinations in UI 1, and “Buttons” is a bad label for the group of different color options for buttons), the remaining 8 could be considered guideline violations. For example, the model predicted that the teal “Preferences” label is a poor label for the list item titled “Marketing Notifications” in UI 5. Since there is only one item in that list, the “Preferences” label is probably not needed, or could be more specific to market notifications. Furthermore, the model identified redundancy in UI 2, where the Chat icon appears twice; in addition, this icon is grouped with a different text label in each instance (“Community” and “Newsfeed”), so its intended meaning is unclear. These violations are perhaps more nuanced than those detected by experts, though verification from experts would be needed to confirm that these are indeed violations.

Calculating precision, recall, and F1 score (excluding the eight potential valid violations) gives a value of 0.882 for all scores, based on 2 false positives, 2 false negatives, 13 true positives, and 15 total ground truth violations.

6.3. Large-Scale Analysis of Guideline Violations in Real-World UIs

We ran our metrics on a large dataset of real-world UIs (taken from RICO) to determine the frequency of groups that would be flagged as having poor semantic grouping by our guidelines. Each guideline violation identified reflects a potential suggestion or automation for resdesign by a future design tool built on our guidelines and metrics. Hence, this also estimates the amount of redesign that could be facilitated by these design tools in the future.

Not all guideline violations result in poor semantic grouping. For instance, an unrelated group (violating Guideline 1) could be intuitive if it follows a familiar grouping convention (applies Guideline 2), such as a navbar. Hence, we flag a grouping if it violates both Guidelines 1 and 2. Violations of the other guidelines would directly lead to potentially problematic semantic grouping. We describe the characteristics of each guideline violation below:

1. **Guidelines 1&2:** A group without a text label, where the members are unrelated nor follows a familiar grouping convention
2. **Guideline 3:** An element or group with a poor text label or requires a label but is missing one
3. **Guideline 4:** A group containing redundant members
4. **Guideline 5:** A group with over 10 members at the same level in the grouping hierarchy

After applying our metrics to computationally detect these violations and manually verifying each one, we found that 21.4 percent of UIs in a 9.5k RICO subset had at least one such violation. Figure 6 shows examples of each guideline violation, and Section F in the Appendix details our method.

6.3.1. Results

Our metrics detected Guideline 5's violation with nearly perfect accuracy. However, manual verification found many false positives for other guidelines, largely due to incorrect grouping structure present in the RICO dataset and missing or inadequate computationally generated semantic annotations. Table 1 shows percentages of UIs (out of all 9.5k UIs) with each violation, after manual validation. Figure 6 contains an example of each guideline violation, with explanations in the caption.

Error	**Frequency**
Guideline 3b (Element)	6.1
Guideline 3a (Group)	1.5
Guideline 1b (Effect-Focused)	5.3
Guidelines 1 (Other Subparts) and 2	6.9
Guideline 4	0.8
Guideline 5	3.1
Total	**21.4**

Table 1. The percentage of UIs in the subset with each guideline violation. Violations of a few guideline subparts (Effect-focused, Element Labeling, and Group Labeling) are also reported.

About 21.4 percent of UIs had at least one guideline violation. The most common violations were for Guidelines 1&2 (12.2 percent), Guideline 3 (6.1 percent), and Guideline 4 (3.1 percent). Interestingly, around 5.3 percent of UIs violated effect-focused grouping (Guideline 1b, Figure 6 UI E), which was surprisingly common. Furthermore, uncommon icons without labels were a frequent issue (Guideline 3b, Element Labeling), and unorganized long lists were common (Guideline 4, 3.1 percent).

7. Discussion

We highlight interesting insights from our initial implementation of these guidelines, and findings from the expert studies and the analysis on real-world UIs from RICO.

7.1. Support for Automated Design Tools

Our guidelines form a solid foundation for computational methods, as results from the expert review indicate that these guidelines recommend correct ways to carry out semantic grouping. Our initial implementation also achieved high accuracy on a ground truth expert-annotated dataset; they detected all but one major or catastrophic usability concern, as well as more nuanced violations that were missed by experts, indicating their potential as a valuable design aid.

This initial set of metrics is also a starting point for further development of computational methods for semantic grouping. These metrics could be integrated into a semantic grouping linting tool that could detect guideline violations and output results similar to Figure 5. Designers could even input a screenshot of their UI, and an existing tool, such as (Wu et al., 2021), could extract the grouping hierarchy before our metrics are applied. There is also potential for generative design, where designers would just need to specify the UI elements (equivalent to a flat layout tree), and semantically coherent candidate layouts would be created using reinforcement learning (or similar algorithms) to optimize a numerical score of semantic grouping quality.

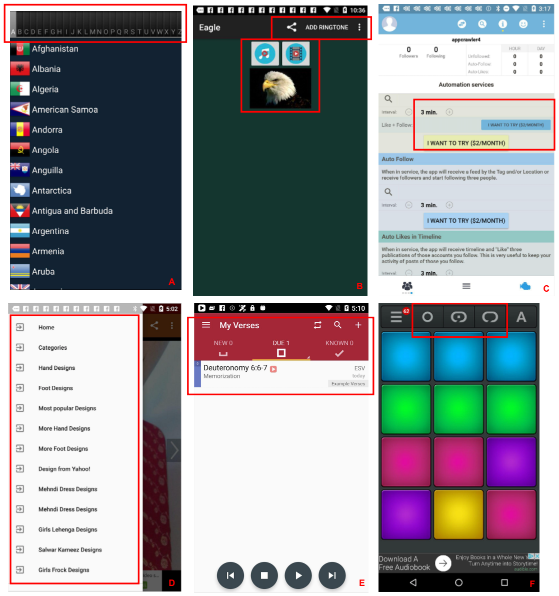


Figure 6. Examples of each type of guideline violation found (in red boxes, with labels in the bottom right corner of the UI). UI A violates Guidelines 2 and 5 as the tabs containing the entire alphabet has too many elements and is an unconventional. UI B violates Guideline 1 (Task Focused) since the "ADD RINGTONE" button is grouped with the unrelated header icons, instead of information about the ringtone. UI C violates Guideline 4 as there are two "I WANT TO TRY ($2/MONTH)" buttons, and it is unclear what each button is for. UI D violates Guidelines 4 and 5, as there are 13 items in the list that could be subgrouped and also has redundant items like "Hand Designs" and "More Hand Designs". UI E violates Guidelines 1 (Effect-focused) and 3 since the tabs are grouped with the unrelated header instead of the section below that they control. Furthermore, the icons are semantically unrelated to their corresponding text labels ("NEW 0", "DUE 1", and "KNOWN 0"). Finally, UI F uses uncommon icons without a text label explaining their purpose, violating Guideline 3.

7.2. Semantic Grouping Guideline Violations in Existing UIs

Our analysis on the RICO subset revealed that a significant number of them had semantic grouping guideline violations. Unfamiliar elements with absent or confusing labels were common, which experts view as severe usability issues, giving these violations an average severity rating of at least 3.5 (between a major and catastrophic usability problem). Furthermore, an expert review participant stated that this could lead users to tap on a mislabelled or unlabelled icon and unintentionally make a serious error, such as starting a livestream. Another common violation was grouping tabs with the header, as opposed to the content the tabs affect (Guideline 1b, see Figure 6 UI E). The fact that this issue is so prevalent suggests it may be common grouping practice amongst apps on the Google Play Store. This could explain why no participant was able identify this violation during the heuristic evaluation, as opposed to Guideline 1b being insignificant. Finally, large groups with no hierarchy (see Figure 6 UI D) were also frequently observed, and a lack of information architecture could slow down task performance (Card, 1982; Bailly et al., 2014; Halverson & Hornof, 2008) and confuse users (Halverson & Hornof, 2008). These frequent semantic grouping violations suggest ample opportunity for redesign facilitated by future tools built on our guidelines and metrics. Studies and expert opinions on these semantic grouping violations imply an improvement in usability following the redesign.

8. Limitations and Future Work

There are several limitations with our computational metrics. They rely heavily on the accuracy of the text-based semantic descriptions of each UI element and the hierarchical grouping structure. This led to the high number of false guideline violations detected during the analysis in Section 6.3, and required manual verification. This manual verification had it own limitations; the guideline violations found in the set of 9.5k real UIs from RICO were decided based on the authors' judgement only, without confirmation from design experts. This differed from the heuristic evaluation described in Section 4, where design experts validated the guideline violations in the set of 6 UIs. Future work is needed to carry out the heuristic evaluation on a larger scale to determine the percentage of flagged guideline violations that match human expert judgement. Another limitation is that these metrics are applied to the grouping hierarchy that is created by the designer (e.g. the android view hierarchy). However, there are cases where this implemented grouping does not match the grouping perceived by users. These metrics would be unable to detect guideline violations in the grouping perceived by users when there is such a mismatch. Finally, these metrics could only be used the assess the coherence of an input group and are unable to detect cases where items that should be grouped are separated in the grouping hierarchy.

These limitations suggest promising opportunities for future work. Data-driven models could be trained on noisy input data to create metrics that are robust against missing or incorrect semantic annotations and erroneous grouping structures. In addition, models that extract the grouping hierarchy from screenshots (e.g. (Wu et al., 2021)) could be applied to attain the grouping structure (as perceived by users) to address cases where there is a mismatch between implemented and perceived grouping. Finally, a numerical semantic coherence score could be developed to evaluate a grouping; it could then be used as the objective or reward function for various optimization algorithms (as done in (Todi et al., 2021)) to automate improvements to a UI's semantic grouping.

References

Allen, M. *The SAGE encyclopedia of communication research methods*. SAGE publications, 2017.

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., and Horvitz, E. *Guidelines for Human-AI Interaction*, pp. 1–13. Association for Computing Machinery, New York, NY, USA, 2019. ISBN 9781450359702. URL https://doi.org/10.1145/3290605.3300233.

Bailly, G., Oulasvirta, A., Brumby, D. P., and Howes, A. Model of visual search and selection time in linear menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pp. 3865–3874, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450324731. doi: 10.1145/2556288.2557093. URL https://doi.org/10.1145/2556288.2557093.

Brown, D. Eight principles of information architecture. *Bulletin of the American Society for Information Science and Technology*, 36(6):30–34, 2010.

Brumby, D. P. and Zhuang, S. Visual grouping in menu interfaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 4203–4206, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450331456. doi: 10.1145/2702123.2702177. URL https://doi.org/10.1145/2702123.2702177.

Card, S. K. User perceptual mechanisms in the search of computer command menus. In *Proceedings of the 1982 Conference on Human Factors in Computing Systems*, CHI '82, pp. 190–196, New York, NY, USA, 1982. Association for Computing Machinery. ISBN 9781450373890. doi: 10.1145/800049.801779. URL https://doi.org/10.1145/800049.801779.

Chi, E. H., Rosien, A., Supattanasiri, G., Williams, A., Royer, C., Chow, C., Robles, E., Dalal, B., Chen, J., and Cousins, S. The bloodhound project: automating discovery of web usability issues using the infoscentπ simulator. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 505–512, 2003.

Deka, B., Huang, Z., Franzen, C., Hibschman, J., Afergan, D., Li, Y., Nichols, J., and Kumar, R. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual Symposium on User Interface Software and Technology*, UIST '17, 2017.

Duan, P., Wierzynski, C., and Nachman, L. *Optimizing User Interface Layouts via Gradient Descent*, pp. 1–12. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450367080. URL https://doi.org/10.1145/3313831.3376589.

Fu, J., Zhang, X., Wang, Y., Zeng, W., Yang, S., and Hilliard, G. Understanding mobile GUI: from pixel-words to screen-sentences. *CoRR*, abs/2105.11941, 2021. URL https://arxiv.org/abs/2105.11941.

Halverson, T. and Hornof, A. J. The effects of semantic grouping on visual search. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '08, pp. 3471–3476, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605580128. doi: 10.1145/1358628.1358876. URL https://doi.org/10.1145/1358628.1358876.

He, Z., Sunkara, S., Zang, X., Xu, Y., Liu, L., Wichers, N., Schubiner, G., Lee, R. B., and Chen, J. Actionbert: Leveraging user actions for semantic understanding of user interfaces. *ArXiv*, abs/2012.12350, 2021.

Henderson, J. M., Weeks Jr, P. A., and Hollingworth, A. The effects of semantic consistency on eye movements during complex scene viewing. *Journal of experimental psychology: Human perception and performance*, 25(1): 210, 1999.

Hoober, S. and Berkman, E. *Designing mobile interfaces: Patterns for interaction design*. " O'Reilly Media, Inc.", 2011.

Höök, K. and Löwgren, J. Strong concepts: Intermediate-level knowledge in interaction design research. *ACM Trans. Comput.-Hum. Interact.*, 19(3), oct 2012. ISSN 1073-0516. doi: 10.1145/2362364.2362371. URL https://doi.org/10.1145/2362364.2362371.

Kotval, X. P. and Goldberg, J. H. Eye movements and interface component grouping: An evaluation method. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 42(5):486–490, 1998. doi: 10.1177/154193129804200509. URL https://doi.org/10.1177/154193129804200509.

Leiva, L. A., Hota, A., and Oulasvirta, A. Enrico: A high-quality dataset for topic modeling of mobile UI designs. In *Proceedings of the 22nd International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, MobileHCI'20, 2020. doi: 10.1145/3406324.3410710.

Li, T. J.-J., Popowski, L., Mitchell, T., and Myers, B. A. Screen2vec: Semantic embedding of gui screens and gui components. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI

'21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380966. doi: 10.1145/3411764.3445049. URL https://doi.org/10.1145/3411764.3445049.

Li, Y., He, J., Zhou, X., Zhang, Y., and Baldridge, J. Mapping natural language instructions to mobile UI action sequences. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8198–8210, Online, July 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.729. URL https://aclanthology.org/2020.acl-main.729.

Li, Y., Li, G., He, L., Zheng, J., Li, H., and Guan, Z. Widget captioning: Generating natural language description for mobile user interface elements. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5495–5510, Online, November 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.443. URL https://aclanthology.org/2020.emnlp-main.443.

Liu, T. F., Craft, M., Situ, J., Yumer, E., Mech, R., and Kumar, R. Learning design semantics for mobile apps. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, pp. 569–579, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450359481. doi: 10.1145/3242587.3242650. URL https://doi.org/10.1145/3242587.3242650.

Martin, P. Y. and Turner, B. A. Grounded theory and organizational research. *The Journal of Applied Behavioral Science*, 22(2):141–157, 1986. doi: 10.1177/002188638602200207. URL https://doi.org/10.1177/002188638602200207.

Nielsen, J. Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pp. 152–158, New York, NY, USA, 1994. Association for Computing Machinery. ISBN 0897916506. doi: 10.1145/191666.191729. URL https://doi.org/10.1145/191666.191729.

Nielsen, J. 113 design guidelines for homepage usability. *Saatavissa: http://www. nngroup. com/articles/113-design-guidelines-homepage-usability/[viitattu 3.4. 2013]*, 2001.

Nielsen, J. Severity ratings for usability problems. 2006.

Nielsen, J. and Molich, R. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 249–256, 1990.

Oulasvirta, A., De Pascale, S., Koch, J., Langerak, T., Jokinen, J., Todi, K., Laine, M., Kristhombuge, M., Zhu, Y., Miniukovich, A., et al. Aalto interface metrics (aim) a service and codebase for computational gui evaluation. In *The 31st Annual ACM Symposium on User Interface Software and Technology Adjunct Proceedings*, pp. 16–19, 2018.

Quiroz, J. C., Louis, S. J., Shankar, A., and Dascalu, S. M. Interactive genetic algorithms for user interface design. In *2007 IEEE Congress on Evolutionary Computation*, pp. 1366–1373, 2007. doi: 10.1109/CEC.2007.4424630.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Reimers, N. and Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL http://arxiv.org/abs/1908.10084.

Ribeiro, J. M. N. Web design patterns for mobile devices. 2012.

Scupin, R. The kj method: A technique for analyzing data derived from japanese ethnology. *Human Organization*, 56, 06 1997. doi: 10.17730/humo.56.2.x335923511444655.

Swearngin, A., Wang, C., Oleson, A., Fogarty, J., and Ko, A. J. *Scout: Rapid Exploration of Interface Layout Alternatives through High-Level Design Constraints*, pp. 1–13. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450367080. URL https://doi.org/10.1145/3313831.3376593.

Todi, K., Weir, D., and Oulasvirta, A. Sketchplore: Sketch and explore with a layout optimiser. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, DIS '16, pp. 543–555, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450340311. doi: 10.1145/2901790.2901817. URL https://doi.org/10.1145/2901790.2901817.

Todi, K., Bailly, G., Leiva, L., and Oulasvirta, A. Adapting user interfaces with model-based reinforcement learning. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2021.

Van Duyne, D. K., Landay, J. A., and Hong, J. I. *The design of sites: Patterns for creating winning web sites*. Prentice Hall Professional, 2007.

Wu, J., Zhang, X., Nichols, J., and Bigham, J. P. Screen parsing: Towards reverse engineering of ui models from screenshots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, UIST '21, pp. 470–483, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450386357. doi: 10.1145/3472749.3474763. URL https://doi.org/10.1145/3472749.3474763.

A. Guidelines

We provide the final version of the Semantic Grouping Guidelines. Each guideline consists of both it's concise and detailed versions. Table 2 contains definitions of important terms found in the guidelines.

A.1. Guideline 1: Related Group Members

A.1.1. Concise Version

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:

- *Task:* all the input fields and widgets for booking a flight
- *Direct effect:* tabs that separate and control the display of the page content should be grouped with the content
- *Category:* different types of rooms in a home monitoring UI
- *Object:* a profile page with details about a person
- *Time:* this month's transactions

This guideline aims to help users understand the purpose of the group.

A.1.2. Detailed Version

Grouped elements or higher level groupings (group of groups) should be related in some way. Different ways of being related include (with overlaps allowed):

1. *Task-focused:*
   (a) For members of a widget-based or task-based group, they are all used to accomplish a task or larger goal (e.g. all the input fields and widgets for booking a flight)
   (b) For members of a content-based group, they all contribute to a goal (e.g. the product's price and other details for its sale)
2. *Effect-focused:*
   Interacting with some group members would affect the other members. At the element level, these elements should be grouped. At the group level, the grouping should occur at the appropriate level in the UI hierarchy. (e.g. tabs that separate and control the display of the page content should be grouped with the content)
3. *Category-focused:*
   (a) For members of a widget-based or task-based group, each of their functionality or purpose falls under the same category (e.g. all widgets for adjusting the user preferences)
   (b) For members of a content-based group, their category should be apparent (if not, see Guideline 3) (e.g. different types of rooms in a home monitoring UI)
      - Allowed actions for each group member should be consistent (e.g. each suggested new contact is grouped with an "add" icon)
4. *Object-focused:*
   (a) For members of a widget-based or task-based group, they all affect a single object (e.g. the widgets to like, download, and shuffle a playlist)
   (b) For members of a content-based group, they all describe a single object or concept (e.g. details about a person or event)
5. *Time-based:*
   The group members all belong in the same timeframe (e.g. this month's transactions)

A.2. Guideline 2: Familiar to Users

A.2.1. Concise Version

The grouping should be familiar to users. This can be achieved by following established design conventions.

A.2.2. Detailed Version

The grouping should be familiar to the app's target user base. It follows established design conventions, so users likely would have seen this grouping before and recall its purpose. These design conventions depend on the type of app (e.g. social media apps typically have icons for home, messages, notifications, and profile in the footer).

- However, this guideline is not meant to discourage innovation when appropriate.

A.3. Guideline 3: Labeling for Clarification

A.3.1. Concise Version

Labels can be used to explain the meaning of an element, the meaning of a group of elements, and/or the meaning of the UI grouping organization. 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.

A.3.2. Detailed Version

Labels may help the user understand the grouping, element, or UI hierarchy. Labels will also improve the accessibility of the UI; for instance they will provide more content for screen readers.

1. *Group Labeling:*
   A label should be used to clarify the group's purpose or category if it's members are related (see Guideline 1), but the relevance is not apparent (e.g. the personalized song recommendations should be labeled with "For You").

2. *Element Labeling:*
   Grouping an element with its corresponding label or caption may help clarify its purpose and make it more accessible to users. For instance, the label for an icon could be used to explain its meaning (e.g. a weather app displaying both an icon illustrating the weather and a text label describing the weather).

   - This guideline is more important for apps from less common categories.

3. *Hierarchy Labeling:*
   Appropriate labeling to groups at multiple levels of the hierarchy can help lead users to understand individual groups and the overall grouping organization in an UI.

A.4. Guideline 4: Avoiding Redundancy

A.4.1. Concise Version

A group should not contain members with redundant functionalities. The purpose is to reduce user uncertainty about each redundant member's purpose.

A.4.2. Detailed Version

A group (at any level in the hierarchy) should not contain members with redundant functionalities or content (in most cases). Redundancy will increase the user's cognitive load by introducing uncertainty regarding each redundant element's purpose and by adding unnecessary visual complexity to the interface.

- The only exception is labeling an element or group for better clarification and/or accessibility, as stated in Guideline 3. In addition, labeling could be used to clarify elements that seem redundant to the user but actually have distinct functionalities.

Term	Definition
element	The basic building block of a user interface (UI). Interactive elements are referred to as widgets (buttons, icons, etc.). Non-interactive elements convey information to the user (e.g. image, text label, text description).
group/grouping	Used to organize the elements in a UI. Grouping is also hierarchical, with low level groups consisting of multiple elements and high level groups consisting of multiple smaller groups. The members of a group are usually related in some way, as described in Guidelines 1 and 2.
widget-based group	A group consisting of mostly interactive widgets that each perform a function.
task-based group	A high level group, where each of its members is a group that supports the completion of a task (e.g. signing up for an appointment time).
content-based group	A group consisting of mostly non-interactive elements or smaller non-interactive groups with the purpose of conveying information to the user.
grouping hierarchy	A hierarchy based on group membership (i.e., the parent group is higher in than hierarchy than the group or element it contains). This is distinct from visual hierarchy, which refers to visually ranking the components in an interface (based on importance) via size, location, and color.

Table 2. Definitions of important terminology used in the guidelines and this paper.

A.5. Guideline 5: Hierarchical Subgrouping for Large Groups

A.5.1. Concise Version

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.

A.5.2. Detailed Version

The UI or any large group (containing many members) should be subgrouped and have a clear hierarchy that shows the subgrouping organization. The subgrouping should follow Guidelines 1-4. This makes it easier for users to parse these large groups.

B. Guideline Generation

This section contains details of our method for creating the initial set of semantic grouping guidelines and the results. We first generated a list of guidelines from existing literature and empirical observations we made on the semantic grouping of example mobile user interfaces. We then revised this initial set of guidelines through several rounds of feedback from design experts, which resulted in the 5 guidelines similar to what is shown in Section A of the Appendix.

B.1. Method

B.1.1. Literature Review

We first consulted existing literature for guidelines on semantic grouping that others have developed in the past. We conducted this literature review by searching the ACM Digital Library for terms such as “User Interface Grouping”, “Interface Semantics”, and “User Interface Design Guidelines”. We extracted guideline candidates by taking relevant points from existing user interface design guides (Nielsen, 1994; 2001; Van Duyne et al., 2007; Ribeiro, 2012; Hoober & Berkman, 2011) and drawing inferences based on experimental setup, results, and discussions of applicable studies (Kotval & Goldberg, 1998; Halverson & Hornof, 2008; Brumby & Zhuang, 2015; Card, 1982; Bailly et al., 2014).

B.1.2. Empirical Observations

To augment the literature review, we also made empirical observations on how designers semantically group UI elements by looking through a diverse set of 27 mobile user interfaces from the RICO dataset (Deka et al., 2017) and web search. The UIs came from various app categories. We also made sure there were examples of both interfaces with good designs and ones with poor designs, so that the guidelines will capture both good practices and mistakes to avoid. Since the observations are based on perceived grouping, we make the assumption that it matches the implemented grouping when recording observations. This is to ensure that the resulting guidelines are applicable to implemented grouping. Two of the authors recorded observations individually and then combined them. The authors then each conducted a round of open coding individually on the set of all observations. Afterwards, the authors collectively organized their codes into higher-level themes via affinity diagramming and axial coding (Allen, 2017).

B.1.3. Expert Feedback

To ensure the extracted guidelines actually recommend methods that result in coherent semantic grouping, we consulted 6 professional UI design experts for their feedback on the importance and clarity of each guideline, as well as ideas for new guidelines to include. We iteratively revised the semantic grouping guidelines based on their feedback.

B.2. Results

The literature yielded three vague candidates that we believed would be directly applicable to mobile UIs: grouped elements should be related in functionality (Kotval & Goldberg, 1998), the grouping should be familiar to users by following design conventions (Nielsen, 1994), and to avoid redundancy (Nielsen, 2001). We obtained some other candidates for semantic grouping that seemed more applicable to web design (large amounts of information should be organized in a hierarchy (Van Duyne et al., 2007)) or to menus (grouped items should fall under the same category (Brumby & Zhuang, 2015; Card, 1982; Bailly et al., 2014; Halverson & Hornof, 2008)). We combined these candidates with results from the empirical observations and eventually obtained guidelines that were more fleshed out versions of these candidates and applicable to mobile UI design.

From the 27 UIs we observed, we recorded 196 observations on semantic grouping. Our observations ranged from very specific comments on widget-based grouping (e.g. “footer contains icons to view calendar, view list of events, see notifications, and view profile”) to more general comments on the overall grouping structure (e.g. “the UI is subgrouped such that the items in each subgroup are highly relevant”), and covered all levels of grouping. Coding and affinity diagramming yielded 6 multi-part guidelines, which fall under two categories: matches user expectations and minimizes confusion for the user. The guidelines under “matches user expectations” (Guidelines 1 and 2) recommend groupings with items that users perceive as being related and groupings that are familiar to users (i.e. users have seen the grouping before and can recall how to use it). The four guidelines under “minimizes confusion” suggest ways to clarify the grouping’s purpose.

Expert feedback led to several changes to the initial set of guidelines: adding Guideline 3c (Hierarchy Labeling) based on a new guideline suggestion, combining “Element Labeling” and “Group Labeling” into a single guideline on labeling (i.e. as subsections to Guideline 3), adding the concise versions of the guidelines (the initial guidelines became the detailed versions), adding visual examples for each guideline, and editing guidelines’ titles to be more descriptive (e.g. “Labeling” became “Labeling for Clarification”). These guidelines can be found in Section A of this Appendix. Figure 7 provides an example of how each guidelines is presented.

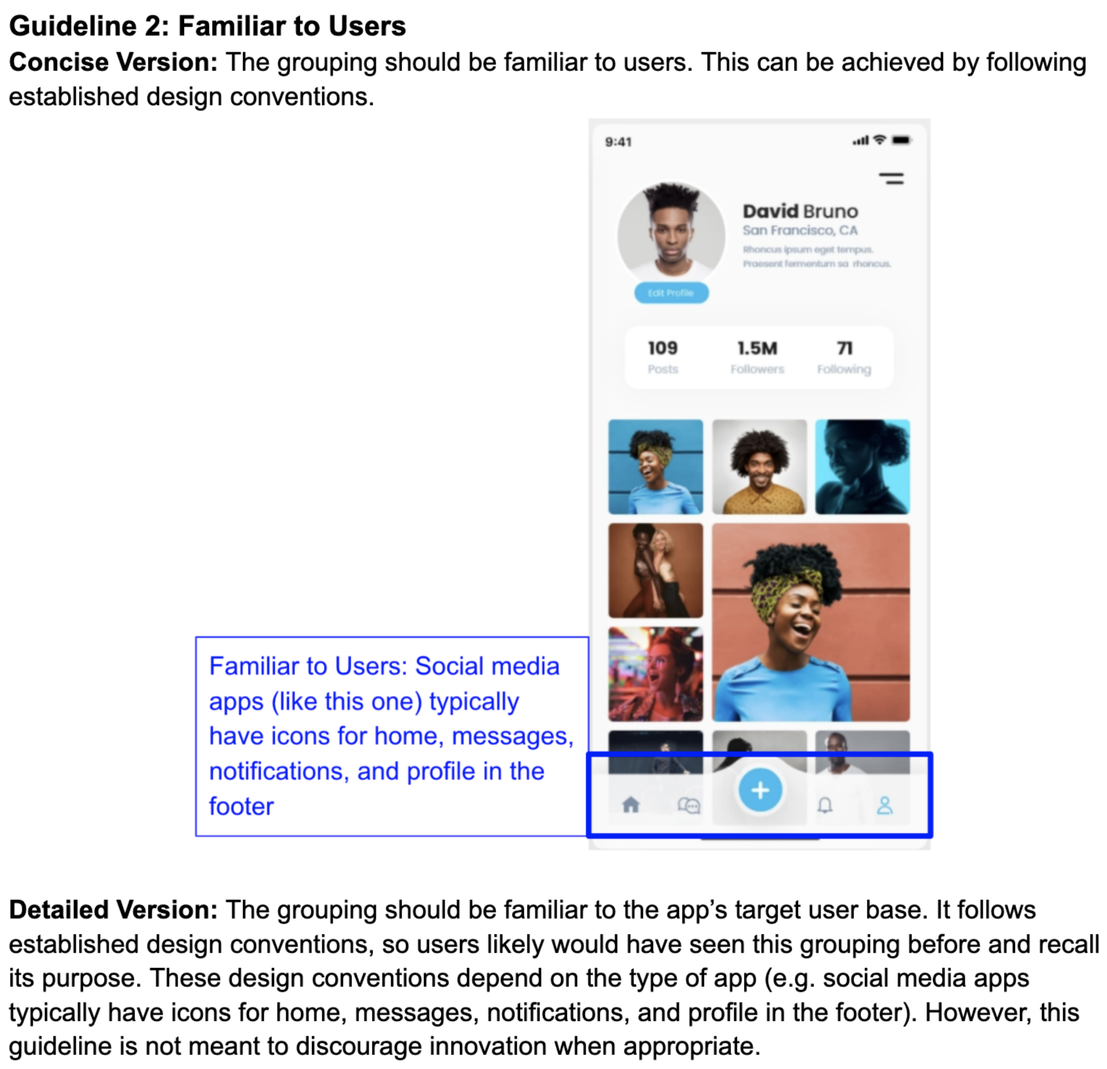
C. Expert Review

This section details our method for the expert review and the results. To ensure that our guidelines form an accurate basis for building metrics, we conducted an expert review of our guidelines to 1) evaluate how clear (i.e. understandable) they are, 2) assess how important each guideline is, and 3) determine how experts have been thinking about semantic grouping in design. 1) is to ensure that the guidelines can be easily understood for future development of metrics. 2) is to ensure that the guidelines recommend best practices for semantic grouping that the experts believe are important to follow, and 3) aims to determine potential use cases for automated semantic grouping tools and these guidelines. We recruited 8 experts from a large technology company [name/location removed for anonymous review]; they specialize in either design or UX research.

C.1. Guideline Clarity

We evaluated guideline clarity by asking participants to read each guideline, explain their interpretation and then rate the clarity on a semantic differential scale (Allen, 2017) with 1 being “Not at all Clear” and 5 being “Very Clear” with justification. Table 3 contains the average clarity rating for each guideline.

Overall, the participants found the guidelines clear, with all but one guideline receiving a clarity rating of 3 or higher.



Guideline 2: Familiar to Users

Concise Version: The grouping should be familiar to users. This can be achieved by following established design conventions.

Familiar to Users: Social media apps (like this one) typically have icons for home, messages, notifications, and profile in the footer

Detailed Version: The grouping should be familiar to the app's target user base. It follows established design conventions, so users likely would have seen this grouping before and recall its purpose. These design conventions depend on the type of app (e.g. social media apps typically have icons for home, messages, notifications, and profile in the footer). However, this guideline is not meant to discourage innovation when appropriate.

Figure 7. An example of how each guideline is presented (Guideline 2 is shown). The concise version summarizes the main idea of the detailed version, and the visual example illustrates an application of the guideline.

Guideline 1 was generally considered the least clear due to the various subcategories (e.g. object-focused), and several participants suggested adding more visual and text-based examples to better illustrate the different subcategories, which we added in the revision following this study.

C.2. Guideline Importance

We assessed guideline importance by first having participants identify applications and violations of each guideline in a set of 6 UIs, which aims to deepen the participants' understanding and for them to see more examples of each guideline's application and violation in UIs. Afterwards, we went through each guideline and had participants rate the importance on a scale of 1 being "Not at all Important" to 5 being "Very Important" with justification.

The participants found all our guidelines to be highly important to consider when grouping elements in a UI, with all guidelines scoring an average of 4.5 or higher. Table 3 contains the average importance ratings as well as a representative quote from the experts on the importance of of each guideline. Participants generally agreed on the importance of each guideline, except Guideline 4 ("Avoiding Redundancy"). Some participants thought avoiding redundancy in an interface was extremely important, stating that redundancy is "confusing and distracting", which "adds to the user's cognitive load". However, others found this guideline less important. They stated that redundancy just "degrades the user experience by

Guideline	Clarity (Concise)	Clarity (Detailed)	Importance	Comments on Importance
1. Related Group Members	3.71 ± 0.92	4.35 ± 0.69	$5. \pm 0.$	“core of any interface design”
2. Familiar to Users	4.28 ± 0.45	4.57 ± 0.73	4.5 ± 0.5	“users may not interact with unfamiliar groupings”
3. Labeling for Clarification	4.71 ± 0.45	4.71 ± 0.45	$5. \pm 0.$	“apps will be unusable if there weren’t labels helping users understand what’s going on”
4. Avoiding Redundancy	4.42 ± 0.73	3.91 ± 0.93	4.5 ± 0.71	“redundancy is confusing and distracting”
5. Hierarchical Subgrouping	4.42 ± 0.49	4.28 ± 0.70	4.88 ± 0.33	“hierarchical subgrouping will make it easier for users to find the information they need”

Table 3. A summary of the experts’ ratings on each guideline’s clarity and importance. The ratings were on a five point semantic differential scale where 1 = “not at all clear” or “not at all important” and 5 = “very clear” or “very important”. The ratings are averaged across all 8 participants and presented with the standard deviation. The fifth column (“Comments on Importance”) includes some representative comments on the guideline’s importance from the experts.

creating confusing but does not lead to critical failure”, or that redundancy may sometimes be helpful (e.g. having duplicate buttons at the top and bottom of the page may reduce user effort). We accounted for this polarizing feedback by revising the detailed version of Guideline 4 to state “A group ... should not contain members with redundant functionalities (**in most cases**)...”

C.3. Semantic Grouping in Practice

To determine use cases for automated semantic grouping tools and these guidelines, we asked participants how they have been thinking about semantic grouping in their design or evaluation work, and how important of a factor it was when designing UIs.

All participants said they think about semantic grouping in their work and that it was highly important, as it “ensures that the interface makes sense to users”. The designers said they mostly think about semantic grouping in the early stages of design, when they are “brainstorming and defining early mock ups”. The UX researchers think about semantic grouping during UI evaluation, such as trying to make sense of user feedback or figuring out why users were confused. Concrete use cases of our guidelines were also suggested, which include visually inspecting a UI to find semantic grouping issues, contextualizing issues users brought up, and serving as an educational resource for UX or design classes.

An interesting finding was that all participants said they think about semantic grouping on an intuitive basis (i.e. not by following rules or guidelines), which could be due to the lack of specific and actionable guidelines. In addition, several participants stated that their intuition agrees with our guidelines and expressed enthusiasm that this process is being formalized with guidelines.

D. Heuristic Evaluation

This section contains details of our method for the heuristic evaluation and the results. The expert review confirmed that our guidelines recommend good practices for semantic grouping, so after some minor updates based on the experts’ feedback, we arrived at the final version of these guidelines. Section A of the Appendix contains these guidelines. We proceeded to use these guidelines in a heuristic evaluation with design experts, and the purpose was to collect expert-identified violations of these guidelines to serve as ground truth for validating our computational metrics and subsequent analyses. We recruited 9 experts consisting of 7 designers and 2 UX researchers. The participants used our guidelines to identify violations in 6 distinct UIs with 16 known guideline violations; the 6 UIs are shown in Figure 8. For each violation identified, they were asked to explain the semantic grouping issue and provide a usability severity rating (Nielsen, 2006) ranging from from 4 (usability catastrophe) to 0 (nonissue).

D.1. Results

Figure 8 shows all the guideline violations found (under "Expert Annotation"), along with a representative quote describing the violation. Fifteen out of the 16 known guideline violations were found by at least one expert. The violation missed was for Guideline 1b; it was an incorrect effect-focused grouping where the tabs are grouped with the header as opposed to the content they control as shown in Figure 8 UI 3.

Figure 9 contains the average severity ratings (across all participants who found the violation) for each guideline violation. Each guideline in the figure is color-coded to match the box around the guideline violation in the UI in Figure 8. The majority of the semantic grouping errors had an average severity rating of at least a 3 (major usability problem) implying that semantic grouping errors could have a significant impact on usability (based on the participants' judgement).

The Guideline 1d violation (Object-focused) was found by 6 (out of 9) participants and was rated a 4 (usability catastrophe) by all 6 experts. It is found in UI 4 (Figure 8), where the "Register" button is not grouped with the card for the event that users are registering for. On the other hand, violations of Guideline 4 (Avoiding Redundancy) had relatively low severity ratings, with most violations having an average rating of 2 (minor usability issue). This aligns with the results from the expert review, where several participants though redundancy was a relatively minor usability issue and may even be beneficial in certain situations.

E. Intuition for Metrics

This section contains a figure (Figure 10) that illustrates the intuition behind each guideline's metrics.

F. Detection of Guideline Violations in RICO

This section describes the method we used to detect guideline violations in a set of real world UIs (taken from RICO).

F.0.1. UI Selection and Semantic Annotations

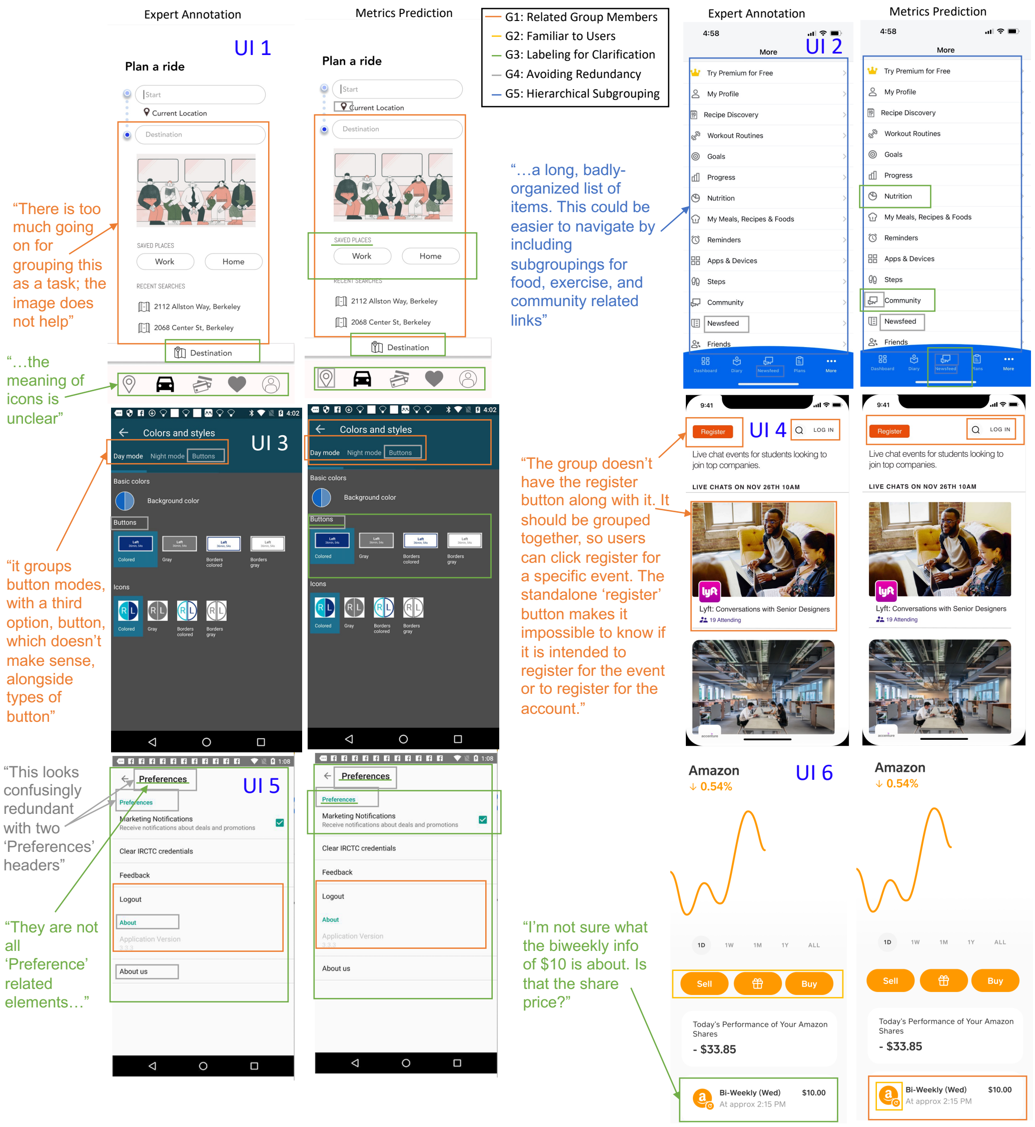
For this analysis, we only included UIs from RICO that were represented in the widget captions dataset (Li et al., 2020b) and had view hierarchies that matched the screenshot, as manually verified by (Li et al., 2020a). This resulted in around 9.5k UIs. Since we are analyzing the android view hierarchies, we are working directly with implemented grouping.

To computationally acquire semantic annotations for UI elements, we used the elements' text labels or widget captions from (Li et al., 2020b). If neither were present, we used the semantic concept annotations from (Liu et al., 2018). None of these sources provide semantic annotations for the content of image elements. Hence, we generated semantic annotations for images by using a pre-trained CLIP model (Radford et al., 2021) to classify the image content to one of 8,000 common nouns, which we then used as the semantic annotation for the image.

F.0.2. Computational Detection of Semantic Grouping Errors

The RICO dataset provides the android view hierarchy for each UI, and we extracted all the groups by taking the intermediate nodes in the hierarchy. To detect Guideline 3 violations (Labeling) we first look for a text label within the group via some heuristics based on its location. If a text label is detected, we apply the corresponding metric for Guideline 3 to assess it's quality. We also return detected non-text elements without labels to manually review them for violations of element labeling.

If a group does not have a text label, we check for violations of both Guidelines 1 and 2 with their corresponding metrics. We check for Guideline 4 by checking all pairs of elements in a group (other than an element and its corresponding text label) and applying Guideline 4's metric. Finally we check for Guideline 5 by computing the number of members in each group at the same level (from both the top and bottom of the grouping hierarchy) and detecting counts that exceed 10.



"There is too much going on for grouping this as a task; the image does not help"

"…the meaning of icons is unclear"

"it groups button modes, with a third option, button, which doesn't make sense, alongside types of button"

"This looks confusingly redundant with two 'Preferences' headers"

"They are not all 'Preference' related elements…"

"…a long, badly-organized list of items. This could be easier to navigate by including subgroupings for food, exercise, and community related links"

"The group doesn't have the register button along with it. It should be grouped together, so users can click register for a specific event. The standalone 'register' button makes it impossible to know if it is intended to register for the event or to register for the account."

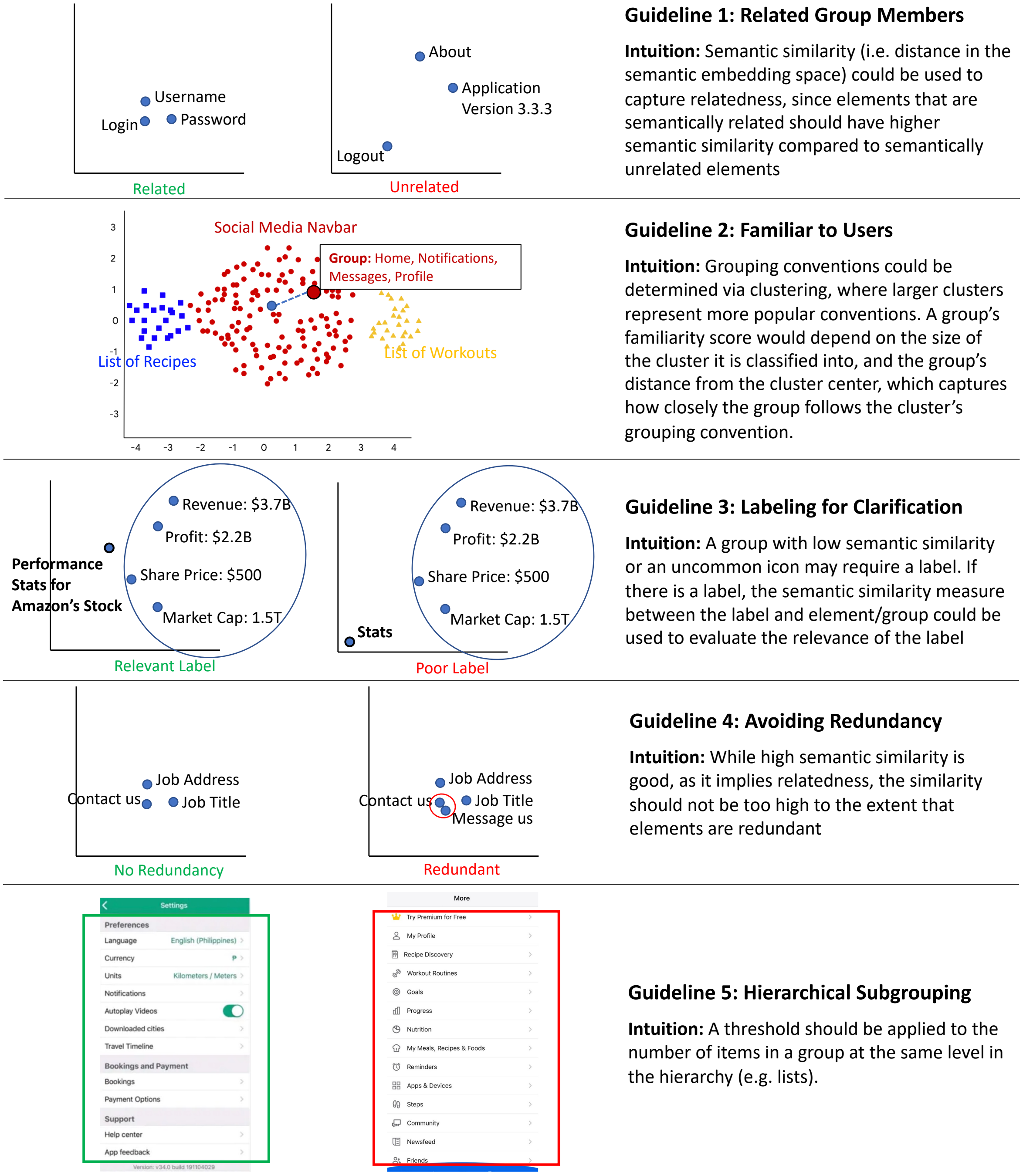
"I'm not sure what the biweekly info of $10 is about. Is that the share price?"

Figure 8. 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. The legend mapping the guideline violated to box color is at the top center.

4 – Usability Catastrophe 3 – Major Usability Problem 2 – Minor Usability Problem 1 – Cosmetic Problem 0 – Nonissue

	UI 1			UI 2		UI 3			UI 4		UI 5				UI 6	
	G1a	G3a	G3a	G5	G4	G1b	G4	G1c	G1d	G2	G1c	G4	G4	G3b	G2	G3b
Average Severity Rating	3.67	3.50	4.00	2.50	2.67	N/A	1.00	3.00	4.00	3.00	2.00	2.00	2.00	3.00	3.00	3.00

Figure 9. The average usability severity score for each guideline violation (averaged across all participants who identified the violation). The guidelines are color-coded to match the boxes around their violations in UIs shown in Figure 8. Guideline 1b has an average rating of "N/A" because no experts found the violation.



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

Preferences
Language English (Philippines)
Currency ₱
Units Kilometers / Meters
Notifications
Autoplay Videos
Downloaded cities
Travel Timeline
Bookings and Payment
Bookings
Payment Options
Support
Help center
App feedback
Version: v34.0 build 191104029

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).

Figure 10. An explanation and visualization for the intuition behind each guideline metric.