Multidimensional Sort of Lists in Mobile Devices

Emanuele Panizzi
Sapienza Università di Roma
Via Salaria, 113
00198 Rome, Italy
panizzi@di.uniroma1.it

Giuseppe Marzo
Sapienza Università di Roma
Via Salaria, 113
00198 Rome, Italy
giumarzo@gmail.com

ABSTRACT

Many apps for mobile devices show lists of elements that the user can sort by choosing a single metric, e.g. price or date or alphabetical order. Often, none of the different sorting is ideal, in fact elements that are more interesting to the user are often spread along the list as there is no single sorting that can group them together at the top. We propose that the user can sort the list by including two or more metrics to create a personalized ranking. Sorting is thus multidimensional and the order of the chosen metrics defines a different ranking according to predefined weights. We designed and developed an iOS framework that implements a multidimensional sortable list, with a drag-and-drop interface to let the user choose the personal order of metrics. We performed qualitative user tests with 8 users.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Interaction styles

General Terms

Design, Experimentation, Human Factors

Keywords

sort, lists, ranking

1. INTRODUCTION

Everyday we interact with lists within mobile apps, like lists of email messages, search engines results, news, list of popular apps, etc. In fact, lists are quite usable as they enable recall, are easy to scroll, and may contain relevant information about each item, like title, details, photos, and so on. When an item is tapped, a new view is shown with all relevant information about the chosen item.

Lists are usually sorted in a predefined order. For example, search engine results are sorted based on their relevance for the given query, and the user cannot change the predefined sort. In other cases, like in hotel booking applications for example, the user can sort the list of hotels according to his/her preferred dimension like price, stars, popularity, user rating, distance, etc. The goal of sorting is to show the most interesting results in the first position in the list. However, this goal is not always reachable, because the 'most interesting' results for a user are the ones that combine a mix of characteristics that can satisfy the user. The user will not choose the cheapest hotel or the nearest one, but will instead look for a good balance among (low) price, distance and user rating (for example). So the user is forced to sort the list in different ways in order to display different items at the top positions and compare them, or to scroll the list and look for his/her favorite hotel by an exhaustive scan of results.

We propose a multidimensional sorting. The user can sort the list according to a personalized ranking computed by assigning a weight to each dimension. In the hotel example, the user can sort hotels assigning a higher weight to the price dimension and lower weights to distance and user rating, in order to move to the top of the list the hotels that best match his/her expectations.

AVI '14, May 27 - 30, 2014, Como, Italy
ACM 978-1-4503-2775-6/14/05.
http://dx.doi.org/10.1145/2598153.2600042

2. RELATED WORK

Google Analytics [1] uses a similar approach in order to sort report tables in the dashboard [2]. The Google Analytics weighted sort is a function that evaluates results based on the selected table columns (dimensions) and moves at the top the most important rows. For example, bounce rate and number of visits can be mixed together to sort the table so as to have in the first rows high bounce rate with a high number of visits, while ignoring cases of high bounce rate if the number of visits is not relevant. This is , to our knowledge, the first proposal of a multidimensional sort of elements in a list on a mobile device.

3. INTERFACE

We ran several qualitative user tests on different existing mobile applications in order to better understand how users work with lists in a mobile environment, what kind of operations they perform (scroll, sort, filter, selection), how they do memorize and recall results, what are the elements most interesting to users in each list item.

According to this analysis, we developed and tested an interface to sort a list multidimensionally. We based our first example on a list of hotels, but we then generalized it, as described in the next section. It is an iPhone interface, implemented on the iOS operating system [3].

Our interface main view contains i) the navigation bar including a title ("SortBy") and a sort button "Change Sort"; ii) a horizontal scroll view that shows the current sort ("0.Stars 1.Price...") and can be shown by swiping right to left; iii) the list of hotels (that scrolls vertically as usual);

When the sort button is pressed, a modal view appears where the user can choose and order the dimensions for his/her personalized ranking.

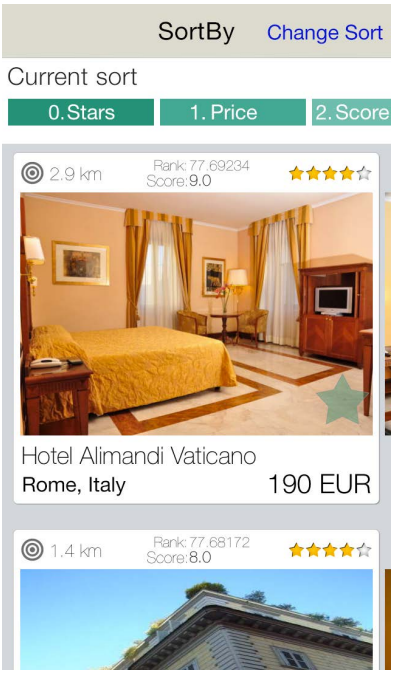

Figure 1: Interface main view.

3.1 Scroll View and Modal View

The horizontal scroll view shows the current sort, and at the beginning a default sort is displayed. A multidimensional sort is displayed by showing the names of the four chosen dimensions in the given order. Each dimension is assigned a predefined weight according to its position. Weights have been determined empirically and dimensions in each position weight double than the ones in the following position. The first dimension is thus weighted 0.8, the second one 0.4, the third one 0.2 and the last one 0.1. The user is not shown weights, but only the four positions in the sort. The intensity of the background color of each dimension is proportional to the dimension's weight.

In order to change sorting, the user taps on the sort button in the navigation bar. A modal view opens up containing four empty positions followed by the list of possible dimensions.

The user can drag and drop a dimension in any position, thus creating his/her personalized sort. The user can include one or more dimensions: a dimension not moved in a position will not be included in the computation. Two or more dimensions can be placed in the same position, and will be given the same weight. Some positions can be left empty and that weight will not be assigned to any of the dimensions. When the save button ("Save", at the top right) is pressed:

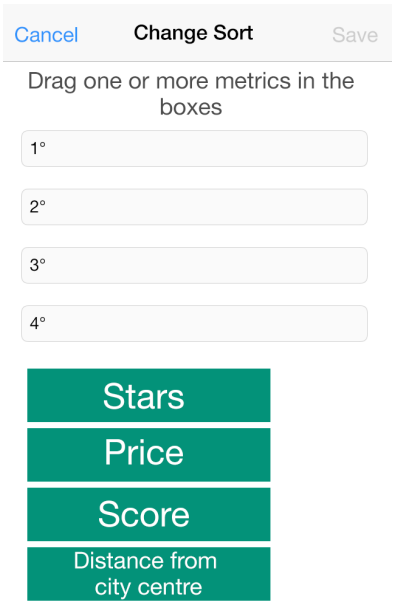

Figure 2: Sort modal view.

- for each chosen dimension, we compute the max and min values of that dimension of all the hotels in the list;
- for each hotel in the list, the value of each dimension is normalized, and then multiplied by the dimension's weight; the sum of such results is the hotel multidimensional ranking;
- the modal view is dismissed and the hotels in the list are sorted according to the computed multidimensional ranking

4. IMPLEMENTATION

We developed an Objective C library that implements this interface for a generic list, allowing a developer to include it in his/her app, choosing list items information to display and dimensions among which the user will choose, and connecting it to the app server and database.

5. CONCLUSIONS

We proposed and implemented a multidimensional sort interface for generic lists of elements in mobile devices. Future developments will include evaluation on a larger user base as well as in different domains.

6. REFERENCES

[1] Google (2010) `http://www.google.com/analytics/`
[2] Cilia M. (2010) *Come funziona l'ordinamento ponderato?* `http://www.goanalytics.info/come-funziona-lordinamento-ponderato/`
[3] Apple (2013) *iOS Human Interface Guidelines* `https://developer.apple.com/library/ios/documentation/userexperience/conceptual/MobileHIG/index.html`