A Graphical Shopping Interface based on Product Attributes

Authors:
Martijn Kagie
Michael van Wezel
Patrick J.F.Groenen

presentation prepared by:
Patrick Lamber
Diana Zverelo
Outline

- Problem definition
- Recommender Systems
- Methodology
- Graphical Recommender System
- Graphical Shopping Interface
- Evaluation of the Graphical Shopping Interface
- Conclusions
Recommender systems loose much information about the mutual similarity between two or more products

- Paradox of choice: more difficult to find an ideal product when there are too many options

Users normally search by

- Limited filter criteria search
- Let the customer describe the "ideal" product

Disadvantage

- Products with same similarity can differ on a completely different set of attributes
Problem definition (2/2)

- New approach: use a 2D visualization to show those differences
  - Show similar products near to each other
- Two prototypes provided
  - Graphical recommender system (GRS)
  - Graphical shopping interface (GSI)
- Inspirations taken from the field of industrial design engineering
  - Explore databases in an interactive way
Outline

- Problem definition
- Recommender Systems
- Methodology
- Graphical Recommender System
- Graphical Shopping Interface
- Evaluation of the Graphical Shopping Interface
- Conclusions
Recommender Systems (1/3)

- Systems that are used by E-Commerce sites to suggest products to their customers and to provide consumers with information to help them decide which products to purchase” [Shafer et al., 2001]

- Suggestions
  - the same for all users
  - dependent on the user's preferences
    - using past purchases
    - navigation behaviour
    - rating systems
    - asking his preferences directly
Recommender Systems (2/3)

- Types of recommender Systems:
  - **Content-based** (suggests products that are similar to the products the customer liked in the past)
  - Collaborative filtering (suggests products that other people with similar taste bought or liked in the past)
  - Hybrid approaches
Case-based reasoning (CBR) not all steps have to be implemented by case-based reasoning recommender system (CBR-RS) data is stored in case library

*domain model* consists of features describing at least one of sub models (*content model*, user model, session model, evaluation model)

CBR-RS gives recommendation based on:
- similarity between cases in the case library
- problem (input of the customer)
Outline

- Problem definition
- Recommender Systems
- **Methodology**
  - Graphical Recommender System
  - Graphical Shopping Interface
- Evaluation of the Graphical Shopping Interface
- Conclusions
Methodology – (dis)similarity measure (1/4)

- products \{x_i\}^n_i in data set D
- products have K attributes: \(x_i = (x_{i1}, x_{i2}, \ldots, x_{iK})\)
- attributes have mixed types: numerical, binary or categorical
- (dis)similarity measures, like Euclidean distance, Person's correlation coefficient, and Jaccard's similarity measure are to handle one attribute type
- general coefficient of similarity proposed by Gower can cope with mixed attribute types
- Similarity \(s_{ij}\) between products i and j is the average of the nonmissing similarity scores \(s_{ijk}\) over the K attributes, where \(m_{ik}\) is 0 when the value for attribute K is missing, 1 when not missing
Similarity score $s_{ijk}$ depends upon the type of the attribute

- for numerical attributes $s_{ijk}$ is based on the absolute distance divided by the range

\[ s_{ijk}^N = 1 - \frac{|x_{ik} - x_{jk}|}{\max(x_k) - \min(x_k)} \]

where $x_k$ is a vector containing the values of the $k^{th}$ attribute for all $n$ products

- for binary and categorical attributes $s_{ijk}$ is defined as

\[ s_{ijk}^C = 1(x_{ik} = x_{jk}) \]

objects having the same category value get similarity score 1, and 0 otherwise
adaptations have to be made:

- the similarity has to be transformed to a dissimilarity
- some variables are more important than others
- influence of categorical/binary attributes on the general coefficient turns out to be too large.

the following adaptations are made

- both types of dissimilarity scores are normalized to have an average dissimilarity score of 1 between two different objects
- since $\delta_{ij} = \delta_{ji}$, dissimilarities having $i \geq j$ are excluded from the sum without loss of generality
Methodology – (dis)similarity measure (4/4)

- the numerical dissimilarity score

$$\delta^N_{ijk} = \frac{|x_{ik} - x_{jk}|}{(\sum_{i<j} m_{ik}m_{jk})^{-1} \sum_{i<j} m_{ik}m_{jk}|x_{ik} - x_{jk}|}$$

- the categorical dissimilarity score

$$\delta^C_{ijk} = \frac{1(x_{ik} \neq x_{jk})}{(\sum_{i<j} m_{ik}m_{jk})^{-1} \sum_{i<j} m_{ik}m_{jk}1(x_{ik} \neq x_{jk})}$$

- the combined dissimilarity measure

$$\delta_{ij} = \sqrt{\frac{\sum_{k \in C} w_k m_{ik} m_{jk} \delta^C_{ijk} + \sum_{k \in N} w_k m_{ik} m_{jk} \delta^N_{ijk}}{\sum_{k=1}^{K} w_k m_{ik} m_{jk}}}$$

vector $w$ is incorporated to emphasize attributes differently
Methodology – Multidimensional Scaling

- dissimilarity scores are used to represent products in 2D space
- low dimensional Euclidean representation can be formalized by minimizing the raw Stress function

$$\sigma_r(Z) = \sum_{i<j}(\delta_{ij} - d_{ij}(Z))^2$$

where the matrix $Z$ is the $n \times 2$ coordinate matrix representing the $n$ products in two dimensions. $\delta_{ij}$ is dissimilarity between objects $i$ and $j$. $d_{ij}(Z)$ is the Euclidean distance between row points $i$ and $j$

- To minimize $\delta_r(Z)$ SMACOF algorithm based on majorization can be used.
Outline

- Problem definition
- Recommender Systems
- Methodology
- Graphical Recommender System
- Graphical Shopping Interface
- Evaluation of the Graphical Shopping Interface
- Conclusions
input is the ideal product described by the costumer

new problem is constructed

in the retrieval phase, a set of cases from the case library is selected and is reused as solutions (outcome)

if results are not satisfying, user adapts his product description or weights of attributes to start the process again in the iterate step
compute weighted dissimilarities $\delta_{i*}$ between $x^*$ and all $x_i$ in data set D.

- $p-1$ products are selected that are most similar to $x^*$
- $p-1$ selected products are combined with $x^*$ in $D^*$ and dissimilarities are computed again
- $p \times p$ matrix $\Delta^*$ with dissimilarities between products is constructed and is an input for MDS algorithm
- The algorithm returns the $p \times 2$ coordinate matrix $Z$
Problem definition
Recommender Systems
Methodology
Graphical Recommender System
Graphical Shopping Interface
Evaluation of the Graphical Shopping Interface
Conclusions
Customers not always know what exactly they want

Help them by providing a set of potential products and let them navigate through the product space

Problem

Which products should be shown to the customer at the beginning?
- input is a product, selected in the 2D space that was created in the previous iteration. It is a new problem.
- in the retrieval phase, a large set of cases that is most similar to the input is selected.
- set is reused as solution.
- In the revise stage, a smaller subset of products is chosen. New set is shown in 2D space.
- user selection is new input for next iteration.
- Implementation not trivial of revise step.
  - the random system
  - the clustering system
  - hierarchical system
First iteration is an initialization iteration
- D will contain the complete case library
- Select p products at random (without replacement)

Next iteration after new user selection
- Take smaller D, with size \( \max(p - 1, a^t n-1) \)
- Select p random products in D and compute dissimilarity matrix
- Compute MDS to get 2-dimensional representation.
Algorithm 1 GSI implementation using random selection

procedure RANDOM_GSI(D, p, α)
    \[ D_0 = D. \]
    Generate random \( D_0^* \subset D_0 \) with size \( p \).
    Compute \( \Delta_0^* \) given \( D_0^* \) using (6).
    Compute \( Z_0 \) given \( \Delta_0^* \) using MDS.
    \( t = 0. \)
    repeat
        \( t = t + 1. \)
        Select a product \( x_t^* \in D_{t-1}^* \).
        Get \( D_t \subset D \) containing \( \max(p - 1, \alpha^t n - 1) \) products most similar to \( x_t^* \)
        using (6).
        Generate random \( D_t^* \subset D_t \) with size \( p - 1 \).
        \( D_t^* = D_t^* \cup x_t^* \).
        Compute \( \Delta_t^* \) given \( D_t^* \) using (6).
        Compute \( Z_t \) given \( \Delta_t^* \) using MDS.
    until \( D_t^* = D_{t-1}^* \).
end procedure
Random selection replaced by a clustering solution

Hierarchical clustering method (average linkage algorithm)
- Calculate dissimilarity matrix
- Clusters are calculated based on it (tree of clusters)
- System only uses the p clusters solution in the tree
- Prototypical product selected in each cluster for p clusters
GSI – cluster system (6/10)

- Prototypical product
  - Smallest total dissimilarity to the other products in the cluster
    \[ i_c = \arg \min_i \sum_{j=1}^{nc} \delta_{ij} \]
  - Get all prototypical products and compute the dissimilarity matrix in combination with MDS
- Disadvantage
  - Becomes quite slow as product space gets larger
Algorithm 2 GSI implementation using clustering

procedure CLUSTERING_GSI(D, p, α)
    \( D_0 = D \).
    Compute \( \Delta_0 \) given \( D_0 \) using (6).
    Compute \( T_0 \) given \( \Delta_0 \) using average linkage.
    Find \( p \) clustering solution in \( T_0 \).
    Determine prototypical products of clusters using (8).
    Store prototypical products in \( D_0^* \).
    Compute \( \Delta_0^* \) given \( D_0^* \) using (6).
    Compute \( Z_0 \) given \( \Delta_0^* \) using MDS.
    \( t = 0 \).
    repeat
        \( t = t + 1 \).
        Select a product \( x_t^* \in D_{t-1}^* \).
        Get \( D_t \subset D \) containing \( \max(p - 1, \alpha t n - 1) \) products most similar to \( x_t^* \) using (6).
        Compute \( \Delta_t \) given \( D_t \) using (6).
        Compute \( T_t \) given \( \Delta_t \) using average linkage.
        Find \( p \) clustering solution in \( T_t \).
        Determine prototypical products of clusters using (8).
        Store prototypical products in \( D_t^* \).
        Compute \( \Delta_t^* \) given \( D_t^* \) using (6).
        Compute \( Z_t \) given \( \Delta_t^* \) using MDS.
    until \( D_t^* = D_{t-1}^* \).
end procedure
Don't compute each time clusters
- Compute one cluster at the beginning and reuse it
- First iteration take the root node of the computed cluster

Next iterations
- Go down the tree until we find the p cluster solution and get prototypical products
  - If such a cluster solution does not exist, show the remaining products
- Compute dissimilarity matrix and use MDS
Users selects X from this solution
  - the cluster it represents is the new root node for the next computations
Procedure terminates when p is higher than the number of clusters
Algorithm 3 GSI implementation using hierarchical clustering.

```
procedure HIERARCHICAL_GSI(D, p)
    Compute $\Delta$ given $D$ using (6).
    Compute $T$ given $\Delta$ using average linkage.
    $T_0 = T$.
    $t = 0$.
    $n_c = size(D)$.
    repeat
        Find min($n^c$, $p$) clustering solution in $T_t$.
        Determine prototypical products of clusters using (8).
        Store prototypical products in $D_t^*$.
        Compute $\Delta_t^*$ given $D_t^*$ using (6).
        Compute $Z_t$ given $\Delta_t^*$ using MDS.
        Select $x_t^* \in D_t^*$ en determine cluster $D_t^c$ it represents.
        $n_c = size(D_t^c)$.
        $D_t^c$ is root of $T_{t+1}$.
        $t = t + 1$.
    until $n^c \leq p$.
end procedure
```
Outline

- Problem definition
- Recommender Systems
- Methodology
- Graphical Recommender System
- Graphical Shopping Interface
- Evaluation of the Graphical Shopping Interface
- Conclusions
the quality of the 2D spaces was studied by considering the Stress values

goal is to evaluate how easily a customer can find the product he wants using GSI

normalization of Stress value:

$$\sigma_n = \frac{\sum_{i<j}(\delta_{ij} - d_{ij}(Z))^2}{\sum_{i<j} \delta_{ij}^2}$$
Evaluation of GSI (2/5)

- Estimation of goodness of the representations in the GRS:
  - one product from the data set is taken as an ideal product and all other products are case library
  - all attributes are used to compute similarities and dissimilarities, weights are set to 1
  - p-1 most similar products to this ideal product are selected and 2D space is created using MDS
  - procedure is repeated, until each product has functioned once as an ideal product description
  - procedure is done for p = 2 to 10
  - results for the average normalized Stress values:

<table>
<thead>
<tr>
<th>p</th>
<th>Mean Normalized Stress</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$3.36 \cdot 10^{-32}$</td>
</tr>
<tr>
<td>3</td>
<td>$3.13 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>4</td>
<td>$5.77 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>5</td>
<td>$1.21 \cdot 10^{-2}$</td>
</tr>
<tr>
<td>6</td>
<td>$1.96 \cdot 10^{-2}$</td>
</tr>
<tr>
<td>7</td>
<td>$2.63 \cdot 10^{-2}$</td>
</tr>
<tr>
<td>8</td>
<td>$3.16 \cdot 10^{-2}$</td>
</tr>
<tr>
<td>9</td>
<td>$3.62 \cdot 10^{-2}$</td>
</tr>
<tr>
<td>10</td>
<td>$4.05 \cdot 10^{-2}$</td>
</tr>
</tbody>
</table>
Estimation of the navigation in the different implementations of the GSI

- assumptions about the navigation behaviour of the user has to be made
  - customer explicitly or implicitly can specify what his ideal product looks like
  - user compares products using the same dissimilarity measure as the system uses
  - in each step the customer chooses the product that is most similar to the ideal product
- each time, one product is selected as the ideal product and all other products are used as the case library
- procedure is repeated, until every product is left out once
- evaluation is done on the three different implementations with p set to 4, 6, 8 and 10
- for the random and clustering system parameter α varies to the values 0.2, 0.4, 0.6 and 0.8
- before starting a single experiment, we determine which product in the case library is most similar to the product we left out. During each step in a single experiment we use the assumptions above to compute the product the user will select. We stop when the most similar product is in shown set
### Evaluation of GSI (4/5)

#### Results for different specifications of the random system.

<table>
<thead>
<tr>
<th>p</th>
<th>α</th>
<th>Successes</th>
<th>In 5 Steps</th>
<th>Average number of Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0.2</td>
<td>16.8%</td>
<td>16.8%</td>
<td>5.02</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>29.3%</td>
<td>19.3%</td>
<td>6.38</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>41.7%</td>
<td>10.0%</td>
<td>9.12</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>56.1%</td>
<td>5.9%</td>
<td>15.99</td>
</tr>
<tr>
<td>6</td>
<td>0.2</td>
<td>29.3%</td>
<td>28.7%</td>
<td>4.77</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>42.7%</td>
<td>34.9%</td>
<td>5.83</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>52.7%</td>
<td>17.1%</td>
<td>8.01</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>70.1%</td>
<td>10.9%</td>
<td>13.12</td>
</tr>
<tr>
<td>8</td>
<td>0.2</td>
<td>43.0%</td>
<td>42.7%</td>
<td>4.34</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>47.0%</td>
<td>43.6%</td>
<td>5.29</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>66.4%</td>
<td>30.5%</td>
<td>6.96</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>80.1%</td>
<td>16.2%</td>
<td>11.22</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>46.4%</td>
<td>46.4%</td>
<td>4.09</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>58.9%</td>
<td>56.7%</td>
<td>4.82</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>70.4%</td>
<td>37.1%</td>
<td>6.27</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>83.8%</td>
<td>19.9%</td>
<td>9.92</td>
</tr>
</tbody>
</table>

#### Results for different specifications of the clustering system.

<table>
<thead>
<tr>
<th>p</th>
<th>α</th>
<th>Successes</th>
<th>In 5 Steps</th>
<th>Average number of Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0.2</td>
<td>14.6%</td>
<td>14.6%</td>
<td>4.85</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>20.3%</td>
<td>12.2%</td>
<td>6.68</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>19.3%</td>
<td>8.7%</td>
<td>8.90</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>13.7%</td>
<td>10.0%</td>
<td>7.83</td>
</tr>
<tr>
<td>6</td>
<td>0.2</td>
<td>22.1%</td>
<td>21.2%</td>
<td>4.83</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>32.7%</td>
<td>27.1%</td>
<td>6.17</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>44.9%</td>
<td>19.0%</td>
<td>8.30</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>25.9%</td>
<td>11.5%</td>
<td>9.10</td>
</tr>
<tr>
<td>8</td>
<td>0.2</td>
<td>31.5%</td>
<td>29.9%</td>
<td>4.57</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>38.9%</td>
<td>35.5%</td>
<td>5.38</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>60.4%</td>
<td>25.6%</td>
<td>7.34</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>45.8%</td>
<td>13.7%</td>
<td>10.20</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>30.6%</td>
<td>38.9%</td>
<td>4.34</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>50.5%</td>
<td>45.5%</td>
<td>5.04</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>72.6%</td>
<td>29.6%</td>
<td>6.40</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>68.2%</td>
<td>16.2%</td>
<td>11.02</td>
</tr>
</tbody>
</table>

#### Results for different specifications of the hierarchical system.

<table>
<thead>
<tr>
<th>p</th>
<th>Successes</th>
<th>In 5 Steps</th>
<th>Average number of Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>47.4%</td>
<td>11.5%</td>
<td>8.03</td>
</tr>
<tr>
<td>6</td>
<td>47.7%</td>
<td>18.4%</td>
<td>5.69</td>
</tr>
<tr>
<td>8</td>
<td>48.9%</td>
<td>24.6%</td>
<td>5.02</td>
</tr>
<tr>
<td>10</td>
<td>52.3%</td>
<td>38.0%</td>
<td>4.10</td>
</tr>
</tbody>
</table>
### Evaluation of GSI (5/5)

#### Proportions of cases that the ranking of the recommended product was in the specified ranges for the random system.

<table>
<thead>
<tr>
<th>$p$</th>
<th>$\alpha$</th>
<th>1</th>
<th>$\leq 2$</th>
<th>$\leq 3$</th>
<th>$\leq 5$</th>
<th>$\leq 10$</th>
<th>$\leq 25$</th>
<th>$\leq 50$</th>
<th>$\leq 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0.2</td>
<td>16.8%</td>
<td>24.6%</td>
<td>29.9%</td>
<td>36.5%</td>
<td>47.0%</td>
<td>66.7%</td>
<td>84.4%</td>
<td>95.0%</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>29.3%</td>
<td>36.5%</td>
<td>42.4%</td>
<td>49.5%</td>
<td>60.1%</td>
<td>78.5%</td>
<td>91.0%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>41.7%</td>
<td>50.5%</td>
<td>58.9%</td>
<td>64.2%</td>
<td>75.7%</td>
<td>87.9%</td>
<td>94.4%</td>
<td>100.0%</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>56.1%</td>
<td>67.3%</td>
<td>73.2%</td>
<td>80.4%</td>
<td>89.4%</td>
<td>96.0%</td>
<td>98.4%</td>
<td>99.7%</td>
</tr>
<tr>
<td>6</td>
<td>0.2</td>
<td>29.3%</td>
<td>40.8%</td>
<td>45.2%</td>
<td>49.2%</td>
<td>60.4%</td>
<td>75.1%</td>
<td>86.6%</td>
<td>96.0%</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>42.7%</td>
<td>50.5%</td>
<td>57.9%</td>
<td>64.5%</td>
<td>74.1%</td>
<td>88.5%</td>
<td>96.0%</td>
<td>97.8%</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>52.7%</td>
<td>64.8%</td>
<td>69.2%</td>
<td>75.7%</td>
<td>84.1%</td>
<td>91.9%</td>
<td>96.9%</td>
<td>97.8%</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>70.1%</td>
<td>80.1%</td>
<td>82.2%</td>
<td>86.3%</td>
<td>94.1%</td>
<td>97.8%</td>
<td>99.7%</td>
<td>100.0%</td>
</tr>
<tr>
<td>8</td>
<td>0.2</td>
<td>43.0%</td>
<td>54.8%</td>
<td>59.5%</td>
<td>64.5%</td>
<td>75.4%</td>
<td>89.1%</td>
<td>95.3%</td>
<td>98.8%</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>47.0%</td>
<td>58.6%</td>
<td>63.9%</td>
<td>71.3%</td>
<td>77.3%</td>
<td>90.0%</td>
<td>94.7%</td>
<td>98.1%</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>66.4%</td>
<td>76.0%</td>
<td>79.4%</td>
<td>84.7%</td>
<td>91.0%</td>
<td>96.6%</td>
<td>97.5%</td>
<td>98.8%</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>80.1%</td>
<td>87.2%</td>
<td>91.0%</td>
<td>93.2%</td>
<td>97.5%</td>
<td>99.4%</td>
<td>99.4%</td>
<td>99.4%</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>46.4%</td>
<td>56.1%</td>
<td>60.8%</td>
<td>67.0%</td>
<td>79.8%</td>
<td>92.5%</td>
<td>96.6%</td>
<td>99.1%</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>58.9%</td>
<td>70.4%</td>
<td>74.8%</td>
<td>80.4%</td>
<td>91.0%</td>
<td>95.3%</td>
<td>98.8%</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>70.4%</td>
<td>76.0%</td>
<td>81.6%</td>
<td>87.5%</td>
<td>93.2%</td>
<td>97.2%</td>
<td>98.4%</td>
<td>98.8%</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>83.8%</td>
<td>89.7%</td>
<td>92.2%</td>
<td>93.8%</td>
<td>96.0%</td>
<td>98.8%</td>
<td>99.1%</td>
<td>99.1%</td>
</tr>
</tbody>
</table>

#### Proportions of cases that the ranking of the recommended product in the specified ranges for the hierarchical system.

<table>
<thead>
<tr>
<th>$p$</th>
<th>1</th>
<th>$\leq 2$</th>
<th>$\leq 3$</th>
<th>$\leq 5$</th>
<th>$\leq 10$</th>
<th>$\leq 25$</th>
<th>$\leq 50$</th>
<th>$\leq 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>47.4%</td>
<td>56.4%</td>
<td>64.2%</td>
<td>71.3%</td>
<td>77.6%</td>
<td>85.4%</td>
<td>89.7%</td>
<td>95.6%</td>
</tr>
<tr>
<td>6</td>
<td>47.7%</td>
<td>55.1%</td>
<td>62.3%</td>
<td>70.1%</td>
<td>76.0%</td>
<td>85.4%</td>
<td>90.3%</td>
<td>95.6%</td>
</tr>
<tr>
<td>8</td>
<td>48.9%</td>
<td>56.4%</td>
<td>62.3%</td>
<td>70.4%</td>
<td>77.3%</td>
<td>86.9%</td>
<td>94.1%</td>
<td>97.5%</td>
</tr>
<tr>
<td>10</td>
<td>52.3%</td>
<td>58.6%</td>
<td>67.9%</td>
<td>74.8%</td>
<td>81.6%</td>
<td>90.0%</td>
<td>93.8%</td>
<td>98.8%</td>
</tr>
</tbody>
</table>
Outline

- Problem definition
- Recommender Systems
- Methodology
- Graphical Recommender System
- Graphical Shopping Interface
- Evaluation of the Graphical Shopping Interface
- Conclusions
Conclusions (1/3)

- New way to display similarities in a 2D space
  - Two prototypes
    - GRS with user explicit input
    - GSI that can be used to navigate through products
  - GRS
    - Representations acceptable up to 2D spaces with 10 products
    - Quality decreases if p increases
Conclusions (2/3)

- GSI
  - Results of the clustering method is not good enough to be applied in practice
  - Random system with high alpha value should be preferred
  - Hierarchical system should be preferred if we need few steps
Conclusions (3/3)

- Improve the system
  - Extend the domain model with other sub models
  - Allow the customer to select any point in the space
  - Computing weights costs time, instead try to learn ideal weights for a population