Integration of Relational Databases and Web Site Content for Product and Page Recommendation

Przemysław Kazienko
Department of Information Systems
Wrocław University of Technology
Wyb.Wyspiańskiego 27, 50-370 Wrocław, Poland
kazienko@pwr.wroc.pl

Maciej Kiewra
Fujitsu España
General Elio, 2 - entlo. dcha. - 46010 Valencia, Spain
mkiewra@mail.fujitsu.es

Abstract

The World Wide Web is the most popular area to which information retrieval and recommending systems are applied. The majority of techniques uses the web site content and usage as a source of data. Nevertheless, modern websites cooperate with more structured relational databases which can successfully enrich traditional approaches. In this paper a new technique integrating web page content, site usage and relational textual and non-textual data in one coherent system is presented. The final goal is to recommend web pages and products in an e-commerce site using unified data within the agent based ROSA system.

1. Introduction

Nowadays commercial web sites that want to attract potential customers should not restrict themselves to a product catalogue only, but they also ought to maintain a set of *white pages* closely related to the offer. For example, a hard disk manufacturer can present, in its web site, documents that explain how to avoid disk failures or how to increase a disk performance. The majority of today company or e-commerce web sites stores its product information in a database management system (DBMS). This information is presented dynamically on the web pages together with a "static" content.

We believe that the content duality (products and white pages) should be reflected in the recommendation process. The exclusive prompting of products can be regarded as aggressive and annoying, while exclusive recommendation of white pages is not in the interests of the company. The purpose of this document is to describe a recommendation method that permits product and white papers to be integrated.

2. Related Work

Many Information Retrieval and web mining techniques have been applied to web sites in order to process, categorize, search or/and recommend relevant information. Recommendation systems have become an important part of current e-commerce web sites that transform them into adaptive sites (see surveys in [13, 18]). Many of them implement data mining techniques, well known from traditional commerce systems (e.g. association rules), which are applied to customers orders stored in a database. Another approach is related to case-based reasoning [19] that is based on similarity calculation between the current case (e.g. customer's profile, customer's order) and the case database. The similarity measures between attributes and rows in a binary database have been considered in [3]. On the other hand, typical web systems analyse page content (using e.g. content-based filtering [4] or clustering), usage of the system (usage mining) [5, 20] or both [8, 9, 12] in their recommendation engine. These methods usually do not need any user cooperation and the required data is gathered automatically.

However, there are many recommendation methods that benefit from user preferences (user profiles) and explicitly expressed ratings. Having user opinions about previous web pages Pazzani and Billsus use the naïve Bayesian classifier for recommendation of next pages for this user [15]. The gained ratings may be also processed by means of collaborative filtering in which system recommends objects basing on the opinions of other users that are similar to the active one [2, 6].

3. Method Overview

The proposed recommendation method integrates two sources of information: a database and web site content. It operates on a single e-commerce web site. Structured product data comes from a relational database stored in many tables. We do not simplify too much the reality assuming that every product has a related single web page that describes it (Fig. 1). It means that a *product page* with a unique, separate URL corresponds to only one product record in the database. However every e-commerce site contains also other documents (denominated here *normal pages*) that possess static content: the latest company news, product reviews, some practical advise, etc. They can (but not necessarily) be related to particular products.

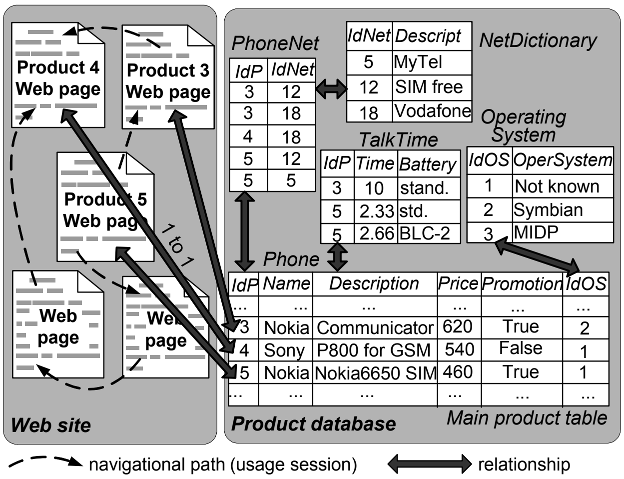

Figure 1. Information sources and their relationships for mobile phone e-commerce site

The main goal of our method is to recommend to the user both *product* and *normal pages*. To achieve it, page textual content and related database product attributes are integrated in a vector space model. Vector´s particular coordinates correspond to textual (terms from pages and textual database attributes) and non-textual database attributes. The obtained model is used to create a document ranking list for every page from the web site. An appropriate top ranked document list is presented to the user online, respectively to the web page which is currently viewed. Every element from the list is an URL address of a web site document. It can be *a product page* or *a normal page* so both page types are recommended together.

The whole process of ranking list generation (Fig. 2) consists of three main steps:

- Source data processing – needed data is acquired and adapted to the method's necessities: document textual content, product database and historical usage
- Vector creation – the data gathered in the previous step is used to create *integrated vectors* that are essential for similarity calculation between all pair of *product* and *normal pages.* The integrated vectors consist of two parts: *textual content part* and *non-textual attribute part*. The former describes occurrence of descriptors on a page or product's text attributes. The latter corresponds to values of non-textual attributes of a product.
- Ranking lists´ creation – once vectors are created, the similarity matrix for all pair of web pages is calculated. In consequence, a document ranking list for each web site page is obtained.

All the above steps are performed offline. The only process which is executed online is the page recommendation. The user browser sends HTTP request to the server which searches for the appropriate ranking list of required page. Links to top *n* pages from ranking list are incorporated into the returned page.

4. Source data processing

The entire source data processing consists of four tasks that reflect method characteristics: descriptor extraction from web content, database relation analysis and attribute selection, *product page* identification and web usage processing. All these tasks (except *product page* identification) should be periodically repeated due to the possible data inconstancy (for example new products have been added). The update problem was solved in the ROSA system by introduction of the multi-agent architecture and implementation of the update method presented in [9].

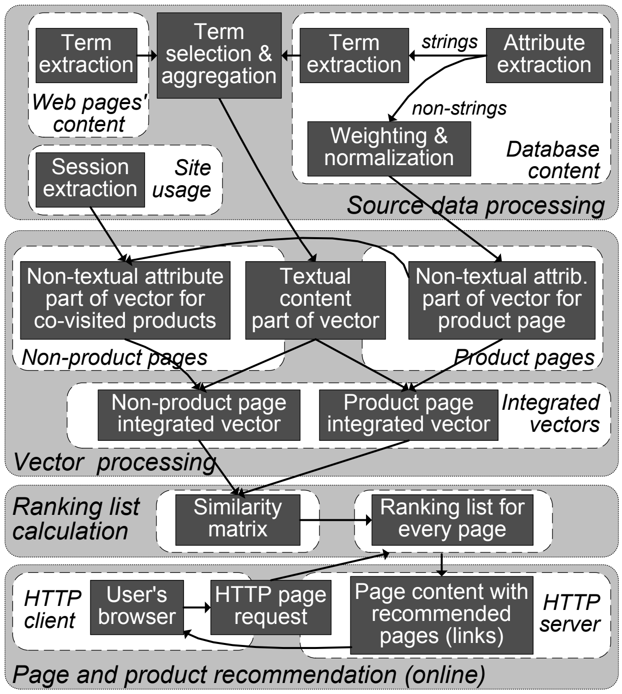

Figure 2. Recommendation process overview

4.1. Descriptor extraction from web content

Descriptors, that character the web pages, are discovered in the first step of the process. A set of occurring terms is extracted for each document. Then, terms matching a stop list are removed. The rest of terms is stemmed - ROSA uses Porter stemming algorithm [16]. The final web descriptor set T^W is composed of all terms that occur in more than $freq_{min}$ and less than $freq_{max}$ documents. Due to our experiments the constants are: $freq_{min} \approx 5$ documents and $freq_{max} \approx 80\%$ of the number of all pages.

4.2. Relation analysis and attribute selection

A database source must be identified before product attribute selection. Usually, product data is stored in many tables (for example values of dictionary fields are frequently maintained in separate tables) which should be joined [11]. In the ROSA system, a human expert must indicate which database tables store data related to the product. First, the *main product table* must be indicated (on the Fig. 1 and 3 - the *Phone* table). This table corresponds directly to the product. Then, the expert determines which attributes from the *main product table* (*Name*, *Description*, *Price*, *Promotion*) and from the other tables (*OperSystem*, *IdNet*, *Time*) are relevant and which should be omitted (*Battery*).

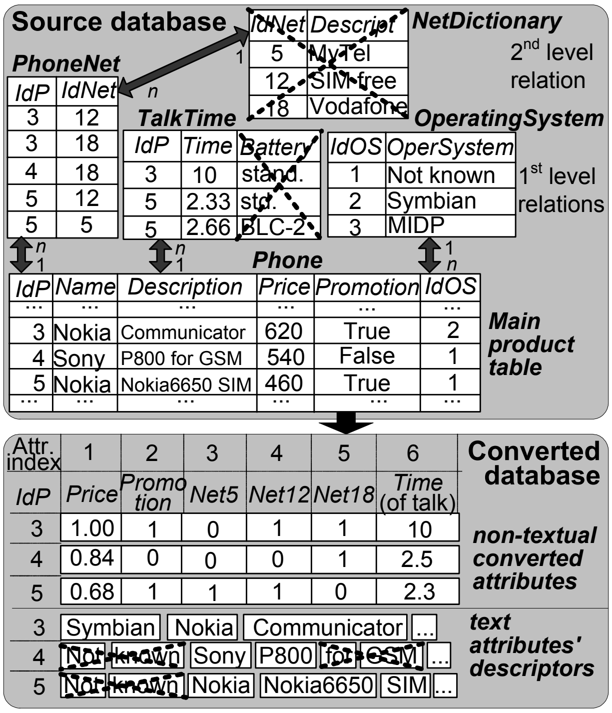

Figure 3. Relation analysis and attribute conversion. Max. value of *Price* attribute is 620 and min.: 120. Coordinates labelled *Net5*, *Net12*, *Net18* correspond to *IdNet*=5, *IdNet*=12 and *IdNet*=18, respectively

The method takes into account only those tables that remain in a direct relationship with the main product table. Such tables are called 1st level relations (Fig. 3). It means that only tables from direct "one to many" (*Phone*:*TalkTime*), "many to one" (*Phone*:*OperatingSystem*) and "one to one" relationships are processed. Please note, that the relationship "many to many" (*Phone*:*NetDictionary*) is composed of two relationships "one to many" (*Phone*:*PhoneNet* and *PhoneNet:NetDictionary*) so the first table is included (*PhoneNet*) and the second one is omitted (*NetDictionary*).

Once the tables are chosen ItemRecommender agent (the next ROSA agent) queries the database system for appropriate set of attributes. It is important to emphasize that apart from textual information (e.g. names, descriptions, notes) each product can be also characterized by non-textual attributes: numbers (e.g. prices, measurements), boolean values (e.g. attachment to current promotion), dates, etc. All non-textual attributes are normalized to the values from the range [0,1]:

- Booleans attributes are converted to 0 (false) and 1 (true) (*Promotion*).
- Numeric values x_i (*Price*) are converted to x_i^{conv} assigning 0 to the minimal attribute value and 1 to the maximal one .
- Date and time attributes are converted into a number of seconds that passed from the midnight of 01/01/1970
- Attributes from tables which are in "1 to many" relationship with the main product table are converted in the following way:
  - for numbers and dates average values are counted (*Talk*),
  - foreign keys to other tables (*IdNet*) and other coded values are replaced by a set of *m* boolean (0-1) attributes; where *m* - cardinality of attribute's domain. Each of them indicates if the given product is related to the foreign key or possesses the coded value - Fig 2. Such operation is called "flattening" and it is often used during data mining coding process [1]. Note, that for "one to many" relationships a product can have more than one attributes with "1" value.
- Each attribute that stores dictionary values without relationship to the *main table* (e.g. *IdOS* if leaving out *OperatingSystem* table) is also "flattened" into a set of boolean attributes in the same way as in related tables.

The number of non-textual attributes after processing (N^A) may increase (comparing to the initial number of attributes) owing to the conversion of some attributes into a set of new boolean attributes.

Textual attributes from the main (*Name*, *Description*) and other tables (*OperSystem*) are used to generate a set of descriptors. Similarly to web documents, stop list words are removed and the rest of terms is stemmed. As a result, a set of product descriptors T^P is obtained. It is summed with the set of web pages' descriptors and the common descriptor set T^C is calculated, as follows: $T^C = T^W \cup T^P$.

4.3. Product home page identification

There are two possible approaches to correlate a particular product to its *product page* what is needed for integration of web pages' with the database content. The first method assumes that a direct "one to one" relationship between *product web page* and a product's database row is explicitly known. Normally, it is true due to the fact that the majority of today web catalogues encodes a database product identifier in a URL query (for example http://companyhost/catalog.jsp?*id=3*).

If for any reasons, the above relationship is impossible to discover, another method, using text processing, must be applied. For example, a page can be regarded as a *product page* if it possesses the highest number of common descriptors with the product textual attributes.

4.4. Web usage processing

The last needed data is a set of products that are visited in one historical session with a given *normal page*. There are two major methods of session acquisition. The former (used in the ROSA system) is based on the online capture of all pages visited by the user. If for any reasons it would not be possible to recognize web sessions online, they must be retrieved offline from web log files. HTTP fields (user agent, referrer, etc.) and data-time stamps allow the HTTP requests to be joined with a certain probability.

5. Vector processing

Non-textual converted attributes and descriptors from pages' content and textual attributes are integrated within the vector space model. For every *normal page* (d^N_i) and *product page* (d^P_i) from the web site a N-dimensional integrated vector c_i is created (Fig. 4). It consists of two parts corresponding to textual (w^T_{ij}) and non-textual ($w^A_{i(N^T+j)}$) data respectively:

$$c_i = \langle w^T_{i1}, \ldots, w^T_{iN^T}, w^A_{i(N^T+1)}, \ldots, w^A_{i(N^T+N^A)} \rangle$$

where N^T=card(T^C) - number of descriptors from web pages and textual database attributes, N^A - number of non-textual converted attributes (after processing - Fig. 3). $N = N^T + N^A$.

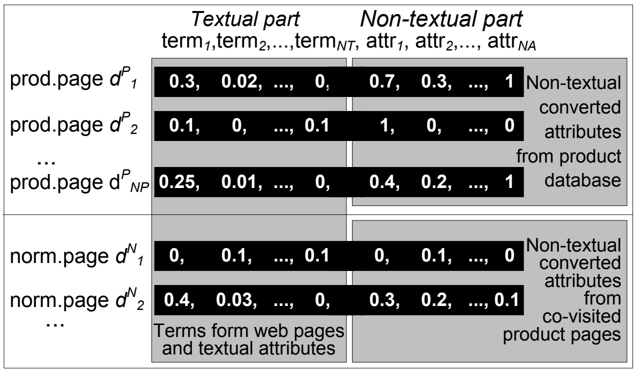

Figure 4. Page vectors

The coordinate w^T_{ij} (textual part) denotes the weight of the descriptor (term) t_j in the document d_i and textual attributes from related product database tuple:

$$w^T_{ij} = \left(tf^b_{ij} + \alpha\, tf^t_{ij} + \beta\, tf^d_{ij} + \gamma\, tf^k_{ij} + \delta\, tf^a_{ij}\right)\log_2\left(\frac{N^D}{n^{t_j}}\right)\frac{1}{\max_T} \qquad (1)$$

where: N^D - the number of all web site pages (both *product* and *normal* ones); n^{t_j} - the number of the web site pages or related textual attributes in which term t_j occurs; tf^b_{ij}, tf^t_{ij}, tf^d_{ij}, tf^k_{ij} - the term frequency of the term t_j in the body, title, description and keywords of the d_i-th HTML page respectively; tf^a_{ij} - the term frequency of the term t_j in all textual database attributes related to the d_i-th *product page*; $\max_T$ - the maximum value of w^T_{ij}, used for normalization.

Title, keywords and description are special parts of the HTML page. They usually contain a "brief page abstract" without needless words. It is worth to increase the importance of all terms they consist of. Descriptions and keywords are quite popular. They occur in 34% of all web pages approximately [10]. α, β, γ coefficients emphasize these special places of term occurrence in (1). Concerning the experiments from [7] α=10, β=γ=5. δ constant permits the terms from textual attributes to influence much more on coordinate value than terms from the body of the HTML page. It seems that the formers are "more descriptive" than the latters. It was assumed that δ=10.

For the *product page* d^P_i the coordinate $w^A_{i(N^T+j)}$ (non-textual vector's part) has the value of j-th non-textual converted attribute (after source data processing) of the product related to this page. See non-textual converted attributes on Fig. 3; *attr. index* corresponds to j.

6. User historical sessions

There are no non-textual attributes for *normal pages* (d^N_i) because they are not directly related to any product. However, we can benefit from web site usage. Users, navigating through the site, bind *normal pages* with *product pages* in a natural way. Thus, knowing historical usage sessions, *product pages* and their integrated vectors, it is quite easy to extract a set of *product pages* d^P_k visited together with a given *normal page* d^N_i.

Non-textual parts of vectors for *normal pages* are calculated using non-textual parts of vectors coming from all co-visited *product pages*. A single vector's coordinate $w^A_{i(N^T+j)}$ for *normal page* d^N_i is obtained, provided that at least one co-occuring *product page* exist, as follows:

$$w^A_{i(N^T+j)} = \frac{\sum_{k=1}^{N^P}\left(w^A_{k(N^T+j)} * conf\left(d^N_i \rightarrow d^P_k\right)\right)}{\sum_{k=1}^{N^P} conf\left(d^N_i \rightarrow d^P_k\right)} \qquad (2)$$

where N^P - number of all *product pages*; $conf\left(d^N_i \rightarrow d^P_k\right)$ - confidence coefficient denoting with what conditional probability $P\left(d^P_k \mid d^N_i\right)$ a session containing *normal page* d^N_i also contains *product page* d^P_k:

$$conf\left(d^N_i \rightarrow d^P_k\right) = P\left(d^P_k \mid d^N_i\right) = \frac{n_{ki}}{n_i} \qquad (3)$$

where n_{ki} - number of sessions with both d^N_i and d^P_k page; n_i - number of sessions containing *normal page* d^N_i.

Applying (3) to (2) we obtain:

$$w^A_{i(N^T+j)} = \frac{\sum_{k=1}^{N^P}\left(w^A_{k(N^T+j)} * n_{ki}\right)}{\sum_{k=1}^{N^P} n_{ki}} \quad (4)$$

The above formula permits non-textual part of integrated vector to be achieved for *normal pages* using non-textual parts of co-visited *product pages*.

The equation (4) is not valid for new or seldom visited *normal pages* which do not have common sessions with any *product pages*, $\sum_{k=1}^{N^P} n_{ki}$ =0. In such case, the coordinate $w^A_{i(N^T+j)}$ is calculated as the average value from non-textual parts of all *product pages*, as follows:

$$w^A_{i(N^T+j)} = \frac{\sum_{k=1}^{N^P}\left(w^A_{k(N^T+j)}\right)}{N^P}$$

7. Product and page recommendation

Once the integrated vectors are obtained the similarity $S(d_i,d_j)$ is computed for all pair of pages (d_i,d_j). In consequence, the full similarity matrix is generated. Our recommendation method takes advantage of the formula from [17] (usually known as Jaccard coefficient). It is used for both *product* and *normal pages* integrated vectors:

$$S(d_i,d_j) = \frac{\sum_{k=1}^{N} w_{ik}w_{jk}}{\sum_{k=1}^{N} w_{ik}^2 + \sum_{k=1}^{N} w_{jk}^2 - \sum_{k=1}^{N} w_{ik}w_{jk}}$$

Next, the ranking list of the closest pages is calculated separately for each page *d* The top *n* pages with the highest similarity value are taken from the ranking list of the page d_i, while generating content of the page d_i. Considering the presentation possibilities *n* is usually about 3-6. Please note that both *product* and *normal pages* are proposed by the system within one unified recommendation hyperlink list.

8. Relationships “many to many”

The assumption that the relationship between the main product table and *product pages* is “one to one” may be too restrictive. There are e-commerce sites in which: one product is presented on many *product pages* or one *product page* contains information about many products.

Thus, we have “many to many” relationship and the method presented above has to be extended. In the first case, the textual part of the vector c_i (w^T_{ij}) is obtained for each product page d^P_i by taking into account descriptors from textual attributes of the related product and the content of the page d^P_i, using (1). Similarly, non-textual attributes are used in non-textual part of the vector ($w^A_{i(N^T+j)}$) for each related *product page*. In this way for *n* product pages the database information is copied *n* times.

The disadvantage of this approach is the possibility of multiple recommendation of the same product on one web page which means the recommendation of several *product pages* related with the given product. If we wanted to avoid a such situation, a mechanism of exclusion should be implemented at the last stage of recommendation (incorporation hyperlinks into web pages).

The second case refers to many products being offered on a single *product page*. For such page d^P_i attribute values of all related products should be combined. Textual attributes are concatenated with the page content in the same way as in the first case. Non-textual attributes could not be simply added to non-textual coordinates $w^A_{i(N^T+j)}$. We suggest using either average or consensus value. The former minimizes the squares’ sum of distance between all products’ attributes and the final mean value $w^A_{i(N^T+j)}$. The latter however seems to be more representative solution because it minimizes the sum of distances (the appropriate algorithm can be found in [14]).

Non-textual parts of vector for *normal pages* $w^A_{i(N^T+j)}$ are calculated without modification, using (4).

9. Implementation and evaluation

The method presented above was implemented in the ROSA system [8, 9]. User Assistant agent recommends the relevant documents and products by means of the DHTML movable layer that is incorporated into every web page (Fig 5).

Figure 5. The ROSA system recommends a mobile hard disk: *HandyDrive*. The ranking list is scrolled, automatically so only one recommendation is visible at once.

The experiments were performed with the ROSA system using data from the intranet of Fujitsu Spain. This site contains around 3000 documents and receives 1000 visits per day. For the purposes of this paper only two, selected groups of products were analysed: hardware and software products.

The practical usage of the method has revealed that there are more products that are presented on many *product pages* (the first case from the section 8) than pages that describe various products (the second case). The recommended pages (top ranked) for selected *product* and *normal pages* were presented in the tab. 1. Note that some recommendation consist of both: *product* and *normal pages*. Multiple occurrences of the same product in one ranking list were removed.

Table 1. Recommendations for selected *product* and *normal pages*. *Normal pages* are italicised

Page (products)	Recommended pages (similarity)
impresoras/DL3700.htm (PrinterDL3700)	*impresoras/index.html* (85%) */noticias/noticias35_4.htm* (81%) */noticias/novedades33_1.html*(75%)
/productos/hd/v_all4.html (Harddisk ALLEGRO)	/intranet/v_all5.html (89%) /intranet/Allegro7.htm (69%)
/productos/hd/v_all4e.html (Harddisk ALLEGRO)	/intranet/v_all5.html, (78%) /intranet/v_all6.html (61%)
/ handydrive/index.htm (HandyDrives:All in one, Photo edition, Data edition, Music edition, video edition)	*/noticias/novedades30_2.htm* (92%) */noticias/noticias31_1.htm* (83%) */noticias/noticias31_1b.htm* (82%)
/productos/eventos.htm (normal page)	*/productos/index.html* (90%) *noticias/noticias31_1b.htm* (85%) /handydrive/index.htm (80%)
/intranet/catering.htm (normal page)	*/intranet/pirp.htm* (86%) */recursoshumanos/index.htm* (80%)

10. Conclusions and future work

Our recommendation method is based on the integration of web pages' content and attributes from a related database. As a result, *product web pages* and white pages (*normal pages*) are recommended together in one mixed list. In consequence, product promotion is not so invasive and it is more acceptable for a user. The introduction of web usage mining permits products to be recommended also on white pages. All these features enable the web site content to be dynamically adapted according to the variable e-commerce offer and the structure of the site. It is important to emphasize that our solution can be used not only for a typical e-commerce site. ROSA can recommend masterpieces in a virtual museum, houses in a real estate agency, scholarships in a student portal, etc. The ranking list creation is not the only possible application of the data integration presented in this paper. The obtained page vectors can be clustered, the outliers may be detected and removed, etc. It is possible to extend the method including other data sources available in e-commerce, e.g. purchased product lists, products placed in the basket, etc.

The future works will concentrate on the introduction of the time factor and importance variation (weighting) of non-textual attributes.

11. References

[1] Adriaans P., Zantinge D.: *Data mining*. Addison Wesley Longman, Harlow (1996).

[2] Buono P., Costabile M.F., Guida S., Piccinno A.: Integrating User Data and Collaborative Filtering in a Web Recommendation System. *OHS-7, SC-3, and AH-3 (2001). LNCS 2266*, Springer Verlag (2002) 315-321.

[3] Das G., Mannila H.: Context-Based Similarity Measures for Categorical Databases. *PKDD 2000, LNCS 1910*, Springer Verlag (2000) 201-210.

[4] Durand N., Lancieri L., Cremilleux B.: Recommendation System Based on the Discovery of Meaningful Categorical Clusters. *KES 2003, LNAI 2773*, Springer (2003) 857-864.

[5] Ishikawa H., et al.: An Intelligent Web Recommendation System: A Web Usage Mining Approach. *ISMIS 2002. LNAI 2366*, Springer Verlag (2002) 342-350.

[6] Jung K.-Y, Jung J.J., Lee J.H.: Discovery of User Preference in Personalized Design Recommender System through Combining Collaborative Filtering and Content Based Filtering. *DS'03. LNCS 2843*, Springer Verlag (2003) 320-327.

[7] Kazienko P.: Hypertekst Clustering based on Flow Equivalent Trees. Wrocław University of Technology, Dep. of Inf. Systems, Ph.D. Thesis (2000) in Polish http://www.zsi.pwr.wroc.pl/~kazienko/pub/Ph.D.Thesis2000/PhD.zip.

[8] Kazienko P., Kiewra M.: Link Recommendation Method Based on Web Content and Usage Mining. *IIS: IIPWM'03 Conference, Advances in Soft Computing*, Springer Verlag (2003) 529-534, http://www.zsi.pwr.wroc.pl/~kazienko/pub/IIS03/pkmk.pdf.

[9] Kazienko P., Kiewra M.: ROSA - Multi-agent System for Web Services Personalization. *AWIC'03, LNAI 2663*, Springer Verlag, (2003) 297-306.

[10] Lawrence S., Lee Giles C.: Accessibility of information on the web. *Nature*, Vol. 400, 8 July (1999) 107-109.

[11] Lim E.-P., Chiang R.H.L.: The integration of relationship instances from heterogeneous databases. *Decision Support Systems* 29 (2000) 153-167.

[12] Mobasher B., Dai H., Luo T., Sun Y., Zhu J.: Integrating Web Usage and Content Mining for More Effective Personalization. *EC-Web 2000, LNCS 1875* Springer (2000) 156-176.

[13] Montaner M., López B., de la Rosa J. L.: A Taxonomy of Recommender Agents on the Internet, *Artificial Intelligence Review*, 19 (4) June (2003) 285-330.

[14] Nguyen N.T.: Consensus systems for conflict solving in distributed systems. *Information Sciences* 147 (1-4) (2002) 91-122.

[15] Pazzani M., Billsus D.: Learning and revising user profiles: The identification of interesting web sites. *Machine Learning* 27 (1997) 313-331.

[16] Porter M.F.: An algorithm for suffix stripping. *Program* 14, no. 3 (1980) 130-137.

[17] Salton G., McGill M.J.: *Introduction to Modern Information Retrieval*. McGraw-Hill Book Co. (1983).

[18] Schafer J.B., Konstan J.A., Riedl J.: Recommender systems in e-commerce. *EC-99*, ACM (1999) 158-166.

[19] West G.M., McDonald J.R.: An SQL-Based Approach to Similarity Assessment within a Relational Database. *ICCBR 2003, LNCS 2689*, Springer Verlag (2003) 610-621.

[20] Yao Y.Y., Hamilton H.J., Wang X.: PagePrompter: An Intelligent Agent for Web Navigation Created Using Data Mining Techniques. *RSCTC 2002, LNCS 2475* Springer Verlag (2002) 506-513.