Przemysław
Kazienko

Associations: Discovery, Analysis and Applications


EX LIBRIS
IZ
BIBLIOTEKI
POLITECHNIKI
WROCŁAWSKIEJ

Associations: Discovery, Analysis and Applications

Przemysław Kazienko

Oficyna Wydawnicza Politechniki Wrocławskiej
Wrocław 2008

Reviewers
Jerzy Kisilewicz
Tadeusz Morzy

Editorial layout and proof-reading
Halina Marciniak

Cover design
Justyna Godlewska-Iskierka

Oficyna Wydawnicza Politechniki Wrocławskiej
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
http://www.oficyna.pwr.wroc.pl
e-mail: oficwyd@pwr.wroc.pl

ISBN 978-83-7493-444-2

Drukarnia Oficyny Wydawniczej Politechniki Wrocławskiej.

To my lovely wife and children

Acknowledgements

The author is indebted to his graduate students Katarzyna Kuźmińska, Katarzyna Musiał, Michał Adamski, Mariusz Matrejek, Marek Muzyka, Marcin Pilarczyk, Janusz Wiśniowski as well as the former manager of the WUT site Marek Zimnak, and the content managers of other web sites for their help at experiments.

Special thanks are due to my wife and children for their patience and support.

The monograph has been partly supported by The Polish Ministry of Science and Higher Education, grant no. N516 037 31/3708.

Table of Contents

Symbols

$\boldsymbol{as}_j$ – active user ad session vector for the jth active session

$\boldsymbol{cs}_{k'}$ – the k'th usage pattern for cluster k'

$\boldsymbol{cta}_k$ – advertising conceptual space for cluster k

$\boldsymbol{ctp}_k$ – publisher's conceptual space for cluster k

$\boldsymbol{cv}_{k'}$ – ad visiting pattern corresponding to $\boldsymbol{cs}_{k'}$

$con(X \rightarrow Y)$ – confidence for the association rule $X \rightarrow Y$

$con(X \rightarrow \sim Y)$ – confidence of a confined negative rule $X \rightarrow \sim Y$

$con^{P\#}(d_i \rightarrow^{P\#} d_j, d_k)$ – partial indirect confidence for the partial indirect association rule $d_i \rightarrow^{P\#} d_j, d_k$

$con^{s^-}(q \rightarrow \sim X)$ – confidence of sequential pattern with negative conclusion s^- $(q \rightarrow \sim X)$

$con^{\#}(d_i \rightarrow^{\#} d_j)$ – complete indirect confidence for complete indirect association rule $d_i \rightarrow^{\#} d_j$

$con^{*}(d_i \rightarrow^{*} d_j)$ – confidence for the complex association rule $d_i \rightarrow^{*} d_j$

$C(y \rightarrow x)$ – the commitment function from user x to y in the virtual social network

$C(q,t,K)$ – complement of subsequence q in sequence t with respect to index K

$C^{max}(q,t)$ – maximum complement of subsequence q in sequence t

$CC(x)$ – closeness centrality of user x in the virtual social network

d_i – the ith document in the domain D

$d_i \rightarrow^{P\#} d_j, d_k$ – partial indirect association rule from d_i to d_j with the transitive page d_k

$d_i \rightarrow^{P\#} d_j, D_K$ – partial indirect association rule with the set of transitive elements D_K

$d_i \rightarrow^{\#} d_j$ – complete indirect association rule from d_i to d_j

$d_i \rightarrow^{*} d_j$ – complex association rule from d_i to d_j

D – domain, the set of domain documents, the multiset of domain user sessions

$DC(x)$ – degree centrality of user x in the virtual social network
$DP(x)$ – degree prestige of user x in the virtual social network
$\boldsymbol{e}_j$ – ad emission vector for the jth active session
$\boldsymbol{eau}_j$ – ad emission acceptance for the jth active session
$\boldsymbol{epu}$ – emission per user vector
ε – constant coefficient of influence of others in the social position function
$iconmin$ – minimum complete indirect confidence
K – index (positions) of subsequence q in source sequence t
max_T – maximal number of component partial rules for a pair of pages in the site
$mincon$ – minimum confidence for association rules
$minconneg$ – minimum confidence for negative association rules
$minconpos$ – minimum confidence for positive association rules
$mincon^{s^-}$ – minimum confidence for sequential patterns with negative conclusions
$minsup$ – minimum support for association rules
$minsup^{s^-}$ – minimum support for sequential patterns with negative conclusions
$NR(p_i,p_j)$ – Negative Recommendation function from page p_i to p_j calculated from association rules
$NR^{seq}(p_i,p_j)$ – Negative Recommendation function from page p_i to p_j calculated from sequential patterns with negative conclusions
$\boldsymbol{p}$ – ad priority vector
p_i – the ith web page in the web site
$p_1 \Rightarrow p_2$ – hyperlink from page p_1 to p_2
$piconmin$ – minimum partial indirect confidence
$\boldsymbol{ps}_j$ – active user page session vector for the jth active session
P – set of web pages in the web site
$PP(x)$ – proximity prestige of user x in the virtual social network
$PR(p_i,p_j)$ – Positive Recommendation function from page p_i to p_j calculated from association rules
$PR^{seq}(p_i,p_j)$ – Positive Recommendation function from page p_i to p_j calculated from positive sequential patterns
q – sequence, sequential pattern
$rank(p_i \rightarrow p_j)$ – ranking function of page p_j on page p_i
$\boldsymbol{rank}_j$ – rank vector for the jth active session
R – set of relationships in the virtual social network
$\boldsymbol{s}_j$ – the jth session vector
$s^-(q \rightarrow \sim X)$ – sequential pattern q with negative conclusion $\sim X$

$sim(p_i,p_j)$ – similarity between the page p_i and p_j
$sup(q)$ – support of sequence q in source sequences T
$sup(X \rightarrow Y)$ – support for the association rule $X \rightarrow Y$
$sup(X \rightarrow \sim Y)$ – support of confined negative rule $X \rightarrow \sim Y$
$sup^{s^-}(q \rightarrow \sim X)$ – support of sequential pattern with negative conclusion $s^-(q \rightarrow \sim X)$
S, S_i – user session, the ith user session in the web site
S^+ – set of pages visited during the user session
S^- – set of pages not visited during the user session
S^S – the set of all user sessions
$S(x \rightarrow y)$ – strength of email communication from user x to y
$SP(x)$, $SP_0(x)$, $SP_n(x)$ – social position of user x in the virtual social network, initial social position, social position after n iterations, respectively
SSP_n – sum of social positions for all nodes after n iterations
$\sigma(A,B)$ – Spearman's coefficient between two rankings A and B
τ – error level (precision) at iterative calculation of social position
$\tau(A,B)$ – Kendall's coefficient of concordance between two rankings A and B
τ^p_k/τ^n_k – limit inferior of the kth positive/negative interval in classification of recommendation functions
t_i – the ith source sequence (navigational path)
tf_{ki} – term frequency of term t_k's in page (document) p_i
T – multiset of source sequences
T_{ij} – set of transitive pages for partial indirect association rules from d_i to d_j
$\boldsymbol{ta}_j$ – advertiser term vector related to term t_j
$\boldsymbol{tp}_j$ – the jth term page vector related to term t_j
$\boldsymbol{v}_j$ – visited ad vector corresponding to session $\boldsymbol{s}_j$
$verif^+(p_i,p_j)$ – positive verification function from page p_i to p_j
$verif^-(p_i,p_j)$ – negative verification function from page p_i to p_j
$VSN=(M,R)$ – virtual social network with nodes (members) M and relationships R
w_{ki} – weight of the term t_k in page (document) p_i
$X \rightarrow Y$ – association rule/direct association rule/positive association rule between set X and Y
$X \rightarrow \sim Y$ – confined negative association rule between X and Y

1 Introduction

The world surrounding us contains a vast amount of objects that are somehow connected with each other. These connections, called associations in this monograph, reflect correlations, dependencies and relationships of various kind within pairs or sets of objects. Some associations are directly visible and available for immediate processing like relationships between human beings derived from their mutual email exchange or hierarchical relations between products in a retail shop.

However, some associations are hidden in the data and can be exposed only by means of some specialized methods. These association mining techniques are an important component of the more general domain called *data mining* or *knowledge discovery in databases* (*KDD*). As the output of association discovery, we obtain some frequent patterns invisible in advance, which reflect most popular and reliable associations between objects.

Both calculated and discovered associations often require some pre-processing and after extraction can be further analyzed and processed. This includes especially filtering mechanisms that facilitate the extraction of only potentially useful patterns.

Additionally, depending on the application, associations together with their strength measures can be used to gain new knowledge and provide new patterns or measures. For example, based on the associations computed from mutual user activities, we can calculate measures that portray importance of individual people in the community. In another solution, association patterns extracted from web logs and reflecting typical user behaviour, can be utilized to verify usability of hyperlinks incorporated into web pages. Associations are also widely used in recommender systems when creating recommendation list based on the item-to-item correlation and association strength measures.

The main subject of this monograph are some selected types of associations, their discovery or calculation, analysis as well as some of their specific applications, in particular in recommender systems, hyperlink assessment, web advertising and social network analysis.

1.1. Motivation

Data mining is one of the most extensively studied, fascinating domains in computer science. It is useful in almost every environment with large datasets - there are plenty of commercial applications of methods developed within data mining. Association discovery and processing are crucial directions in data exploration, which have been studied for almost 20 years now. Due to their importance, associations appear to be an interesting research subject.

However, there exist many different types of associations and associative patterns. Some of them are quite well recognized like association rules whereas the others still require some studies like negative association rules or sequential patterns. The latter still yield for studies on their application domains. Besides, all the methods often need to be specially adopted to new, emerging application areas such as recommender, personalized or evolving systems or systems based on social collaboration.

There is also space for research on completely new patterns and association types, which would be able to provide yet another type of knowledge. This especially refers to negative patterns and dynamic associations created ad hoc for temporary purpose.

All these issues inspired the research on associations, their analysis and applications, which resulted in creation of this monograph.

1.2. Aims of the Monograph

The main goal of this monograph is to describe some selected, recent advances in research on various kinds of associations, their analysis and application domains. In particular, it introduces two new kinds of patterns in data mining: indirect association rules and sequential patterns with negative conclusions. In addition, some other patterns like regular association rules and especially negative association rules together with the above mentioned new patterns are studied in the context of their application, i.e. recommender systems and hyperlink verification systems.

Besides, personalized recommendations that operate on web advertisements make use of yet another data mining technique called clustering. Based on prior clustering processes, some dynamic and personalized associations between the current user interest and the contents, which are linked by ads, can be established on the fly. These flexible associations enable exposed advertisements to be adjusted to the constantly changing user needs.

Even more impenetrable associations occur between human beings that interact with one another in virtual societies. These associations are considered as part

of the broader area, that is, social network analysis. Making use of personal relationships, the importance analysis for individual members within human community can be performed.

To enable association analysis and its application, it was necessary to develop some specialized algorithms that facilitate extraction of condensed patterns from large amounts of data. The description of these new dedicated algorithms can be found in Sec. 3.6, 4.3.3, 5.3.2 and 7.2.4.

Note that the monograph is not a comprehensive thesis on all kinds of associations, all possible methods of their extraction nor all their applications. It only touches those research domains that were subjectively attractive for the author. Hence, it rather focuses on some selected types of associations and pinpoints their significant application fields.

A more detailed description of the monograph's contribution can be found in Sec. 8.1.

1.3. Summary of Sections

The monograph consists of six essential Sections 2 through 7 and final remarks - summary in Section 8.

Section 2 generally introduces some selected types of associations considered next in further sections: association rules, indirect association rules, negative association rules, sequential patterns, sequential patterns with negative conclusions, dynamic, personalized associations utilized for adaptive, personalized web advertising, and, finally, associations between humans in social networks. Additionally, Section 2 contains a general overview of processes related to association computing, including some of their more specific application domains like web advertising, hyperlink verification as well as social network analysis.

Indirect association rules are introduced and studied in Section 3. The IDARM* algorithm for mining these patterns is also presented. Indirect association rules are examined in the domain of recommender systems, where they can successfully extend short recommendation lists.

Section 4 addresses negative association rules. Together with regular positive association rules, these negative patterns are applied to verification of hyperlinks in a single web site. Both positive and negative association rules, which are extracted from historical user sessions by means of PANAMA algorithm, reflect typical user behaviour. Experiments proved the usefulness of these usage patterns to evaluate usability of individual hyperlinks.

The following Section 5 is devoted to other negative patterns, namely sequential patterns with negative conclusions which are in some ways a combination of regular sequential patterns and negative association rules. The SPAWN algorithm

for mining sequential patterns with negative conclusions and their application to verification of hyperlinks as well as content-based recommendation lists are presented as well. Positive and negative sequential patterns are merged with positive and negative association rules to provide a comprehensive view on typical user behaviours.

Another approach to associations is considered in Section 6, where dynamic, personalized relationships between the current user and previously discovered content- and usage-based clusters are studied. Associations created on the fly enable adaptive, personalized recommendation of web advertisements. Additionally, some clustering methods are used to reduce necessary calculations in the assignment performed online.

Finally, Section 7 is related to social network analysis. Weighted associations derived from user activities or communication data are utilized to set up the social network of system users. Next, for each network member, their social position is computed based on the associations incoming from the first level (nearest) neighbours. Social position measure reflects general statement of the member in the entire community. The features of social position measures have been studied both formally and experimentally.

2 Associations

Associations are an important concept, which is very useful in modelling the world around. In general, associations reflect relationships between individual objects. These objects can be for example products in the e-commerce, web pages in the web site, users in the online service and even diverse, possible values of the given attribute describing some concepts.

Association can also be called "relations", "relationships", "links", "binds", "linkages", "ties" or "connections".

All the associations considered in this monograph have two sides and the form $X \rightarrow Y$. Hence, they are directed, i.e. $X \rightarrow Y$ differs from $Y \rightarrow X$. Additionally, both sides differ from each other: X is different from Y. Besides, they have some strength measures - assigned values that reflect their quality or significance.

An exception are personalized associations considered in Sec. 6, which are undirected and unweighted.

2.1. Types of Associations

There are many different kinds of associations studied in the scientific literature. Two main types of associations can be distinguished among other patterns known in the data mining domain: association rules (see Sec. 2.1.1) and sequential patterns (see Sec. 2.1.2). Less typical association rules are indirect association rules, negative association rules and generalized association rules. Sequential patterns can be combined with negative association rules - in consequence, we obtain sequential patterns with negative conclusions.

Only some types of associations have been considered in this monograph, Fig. 2.1. It regards some types of association rules, i.e. indirect association rules (see Sec. 2.1. and 3), negative association rules (see Sec. 2.1.3 and 4) as well as sequential patterns with negative conclusions, see Sec. 2.1.5 and 5.

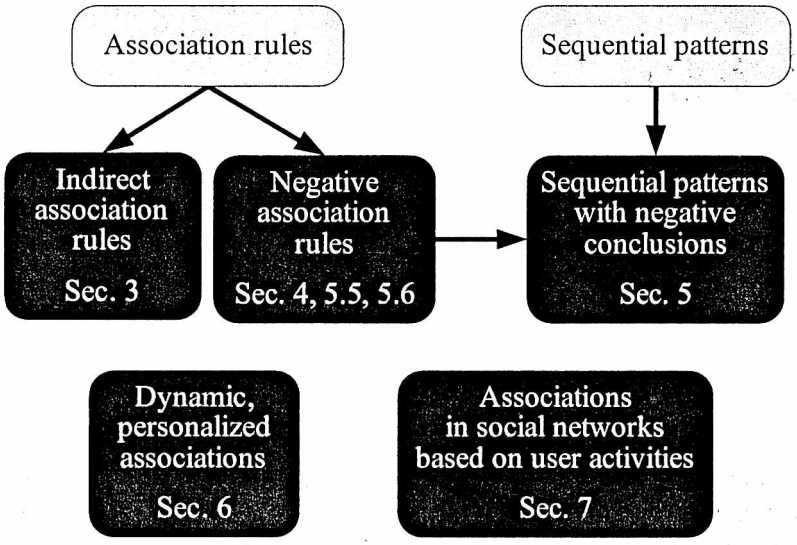

Fig. 2.1. Types of associations studied in the monograph

Besides, associations can be created dynamically, which is essential in personalized recommender systems, see Sec. 2.1.6 and 6.

Associations may be extracted based on user activities directed towards other users. In consequence, some associations between users, which are crucial component of social networks, are created, see Sec. 2.1.7 and 7.

There are plenty of other associations analyzed in scientific literature, for example, connections between concepts in semantic networks or topic maps [Kaz03d], links between ontologies [Kaz08d], simple hyperlinks in HTML [Kaz01] or complex hyperlinks XLinks [Gwi01, Kaz02a, Kaz02c, Kaz03a, Kaz04f] being an element of XML standards [Kaz02a, Kaz02b, Kaz02c, Kaz02d, Kaz02e], associations between users reflecting the level of their mutual trust [Kop07], associations between the same products offered by different e-commerce sites [Pok04], etc.

2.1.1. Association Rules

Association rules indicate frequent relationships between two sets of objects derived from the source datasets. Typical examples of source datasets are transactions in the shop, i.e. products purchased at once by the customer, or user sessions in web site (see Definition 3.2 and 4.1). For instance, a rule $\{p_3,p_6,p_7\}\rightarrow\{p_1,p_4\}$ means that if the set of web pages $\{p_3,p_6,p_7\}$ occurs in some user sessions, then we know with certain confidence that these user sessions also contain the set $\{p_1,p_4\}$. See also Definition 3.3 and 4.2 for the precise definitions of association rules in the web environment.

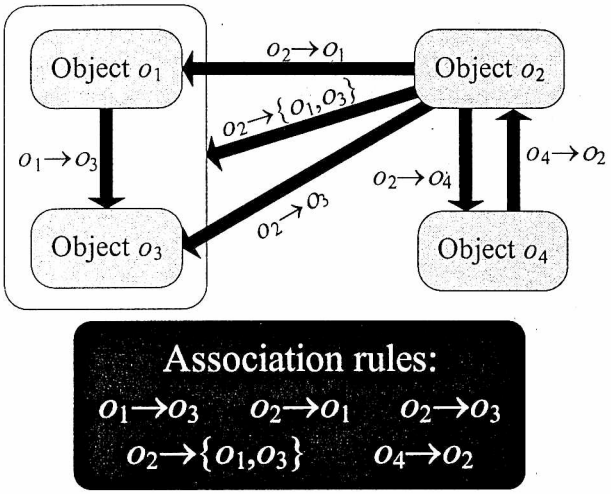

Fig. 2.2. Association rules for sets of objects

Indeed, association rules reflect relationships between sets but to simplify notation for 1-element sets instead of $\{o_i\} \rightarrow \{o_j\}$ the following expression will be used: $o_i \rightarrow o_j$, see Fig. 2.2.

Some example association rules between four objects o_1, o_2, o_3, and o_4 as well as their supersets, e.g. $\{o_1, o_3\}$, are presented in Fig. 2.2. As we can see, there can exist both $o_2 \rightarrow o_1$, $o_2 \rightarrow o_3$ and $o_2 \rightarrow \{o_1, o_3\}$. Moreover, the rule $o_2 \rightarrow \{o_1, o_3\}$ can exist only if there simultaneously exist rules $o_2 \rightarrow o_1$ and $o_2 \rightarrow o_3$.

Association rules are usually described by two main measures: support, Eq. (3.1), (4.1) and confidence, Eq. (3.2), (4.2). Nevertheless, there also exist some other measures like lift, Gini index or κ [Tan02c]. An analysis of the various measures for association rules can be found in [Tan02c].

There are many papers related to algorithms for mining association rules, for example, classical apriori [Agr93, Agr94, Agr96a, Hoe05], parallel ones based on apriori [Agr96b], Eclat [Zak97b, Zak00] and its parallel version [Zak97a], FP Growth [Han00, Han04]. An incremental algorithm FUP was presented in [Cheu96] and improved in [Cheu97]. Another incremental method, DLG was proposed in [Yen96] and next extended to DLG* and DUP [Lee01]. There are also some handbooks that survey knowledge about association rules, e.g. [Han01 - Chap. 6, Mai05, Morz03, Tan06 - Chap. 6 and 7].

2.1.2. Indirect Association Rules

Indirect association rules are specific type of rules derived from direct ones. They reflect indirect relationships between objects. If object o_1 is associated to o_2 and

object o_2 to o_3, then there exists indirect association rule from o_1 to o_3 with respect to o_2, Fig. 2.3 and 3.3, Definition 3.4. Object o_2 is called transitive one. Since there may be many transitive objects between o_1 and o_2, all of them need to be merged into one comprehensive complete indirect association rule, Definition 3.6, Fig. 3.4. Indirect association rules are further studied in Sec. 3.

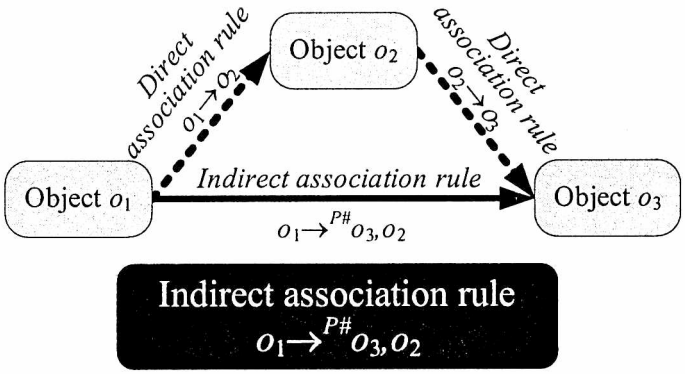

Fig. 2.3. Indirect association rule

2.1.3. Negative Association Rules

Negative association rules are yet another type of association rules. They indicate negative relationships between sets of objects. For example, a rule $\{o_3,o_6,o_7\} \rightarrow \sim\{o_1,o_4\}$ means that if the set of objects $\{o_3,o_6,o_7\}$ occurs in the source transactions (sets), then the set $\{o_1,o_4\}$ does not occur frequently in these transactions. Similarly to regular (positive) association rules, negative ones are characterized by support and confidence measures.

There are several algorithms that extract both positive and negative association rules [Ant04a, Ant04b, Cor06, Don07, Don06, Ten02, Wu04, Yua02], also based on genetic algorithm [Ala05]. Another method to extract multilevel association rules in the specific environment, namely spatial datasets, was presented in [Sha05]. There are also some algorithms for mining only frequent itemsets [Kry05, Pal05] or only negative association rules like algorithm AMENAR [Gan06], MINR [Koh07], DI-apriori [Morz06, Sav98]. However, the first attempt which contained discussion about some constraints on the mining of negative association rules was made in [Bou00], whereas Fortez *et al.* proposed bounded-neg-apriori algorithm [For01]. Some approaches focus on distinguishing various types of negative association rules, i.e. $X \rightarrow \sim Y$, $\sim X \rightarrow Y$, $\sim X \rightarrow \sim Y$ [Don06] or provide new, very similar types of patterns like fuzzy negative association rules [Yan04] or dissociation rules [Morz06]. Some authors introduce additional parameters to filter candidates for negative rules like *multi-confidence* and *Chi-squared test* [Don06], *MinRI* and an external taxonomy [Sav98], *maxjoin* [Morz06] or

correlation [Ant04b]. There are also some specific algorithms that can be applied to unusual datasets like queries to XML documents [Chen05]. Some disadvantages of the above algorithms are discussed in Sec. 4.3.2. Another algorithm to discover both negative and positive association rules, the PANAMA algorithm, which overcomes these drawbacks, is described in Sec. 4.3.

The main application domain for negative association rules is verification of positive knowledge known in advance. For example, they can be used together with positive association rules to assess usability of hyperlinks in the web environment, see Sec. 4.4 and 5.5. Another interesting application is verification of recommendation lists created upon content analysis, see Sec. 5.6.

2.1.4. Sequential Patterns

Sequential patterns are yet another type of associations. They reflect relationships with respect to time, or more precisely, with respect to the order over time. Sequential patterns are extracted from ordered source sequences as their frequent subsequences (see Definition 5.1 and 5.2). For example, for the list of source sequences from Table 2.1, we can extract one subsequence $q=<o_1,o_4,o_3>$ with four occurrences. This frequent subsequence is the sequential pattern q.

Table 2.1. Source sequences and sequential pattern

ID	Source sequences	Frequent subsequence (sequential pattern q)
1	$<o_1, o_2, o_4, o_3>$	$<o_1, o_4, o_3>$
2	$<o_4, o_1, o_2, o_6, o_4, o_8, o_3>$	
3	$<o_6, o_1, o_4, o_6, o_3, o_6>$	
4	$<o_1, o_1, o_4, o_6, o_4, o_3, o_1>$	
5	$<o_1, o_6, o_3>$	

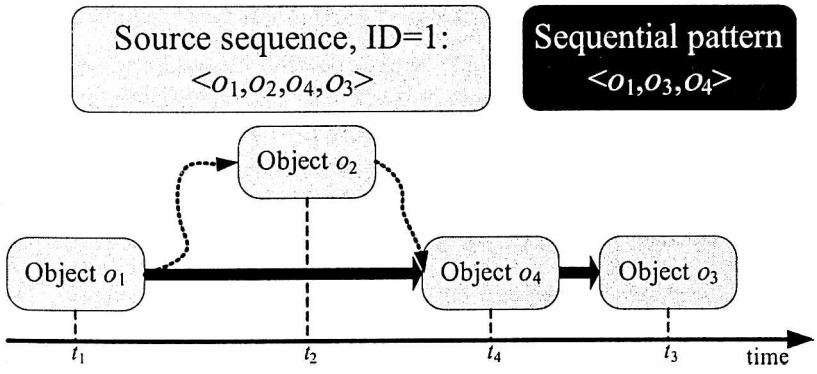

Fig. 2.4. Sequential pattern

Note that in source sequences, elements from the frequent subsequence can be separated by other objects. For instance, object o_2 is between o_1 and o_4, following each other in the sequential pattern q, Fig. 2.4.

There are many algorithms used to mine regular sequential patterns, such as AprioriSome, AprioriAll [Agr95, Tan06 - Sec. 7.4], GSP [Sri96], SPAM [Ayr02] or based on a lattice SPADE [Zak01]. Some algorithms like FreeSpan [Han01] or PrefixSpan [Pei01, Pei04] utilize data projection. Algorithm MEMISP is based on memory indexing [Lin02], whereas SPIRIT integrates constraints by using regular expression [Gar99]. There also exist some incremental approaches [Chen04, Chen07, Ho06, Lau07, Mas03, Ren06, Zha02] and parallel ones [Con05, Zhu07].

Sequential patterns are used in another specific type of patterns, namely sequential patterns with negative conclusion, see Sec. 5. Sequential patterns can also be used as the positive information both in verification of existing hyperlinks, see Sec. 5.5, and content-based recommendation lists, see Sec. 5.6.

2.1.5. Sequential Patterns with Negative Conclusions

Sequential patterns with the negative conclusions are new patterns that are studied in detail in Sec. 5. They are in a sense a kind of combination of regular sequential patterns and negative association rules.

For example, sequential pattern with negative conclusion $s^-(q \rightarrow \sim X)$ denotes how much sequence q is not followed by any element of set X. Such statement can be expressed with certain confidence and support, which are two basic measures of sequential patterns with the negative conclusions.

Note that sequence q in $s^-(q \rightarrow \sim X)$ is the regular frequent sequence, i.e. sequential pattern (see Sec. 2.1.4), which is combined with the set X in the negative way, similarly to the case of negative association rules, see Sec. 2.1.3.

Sequential patterns with the negative conclusions can be discovered from the previously obtained sequential patterns, using the SPAWN algorithm described in Sec. 5.3.2.

Similarly to negative association rules, sequential patterns with the negative conclusions can be useful in verification of any positive knowledge coming from external sources. For example, they can be utilized to evaluate usability of hyperlinks in web sites, see Sec. 5.5. Besides, sequential patterns with the negative conclusions and negative association rules extracted from web logs can in a comprehensive way validate recommendation lists based on similarity of HTML contents of web pages, see Sec. 5.6.

2.1.6. Dynamic, Personalized Associations

A completely different approach to association is presented in Sec. 6. Associations are links between the current, active user visiting a single web site and some pre-

viously extracted patterns, both content- and usage-based. The patterns correspond both to the content of the site and the content of the external sites linked by advertisements, Fig. 2.5.

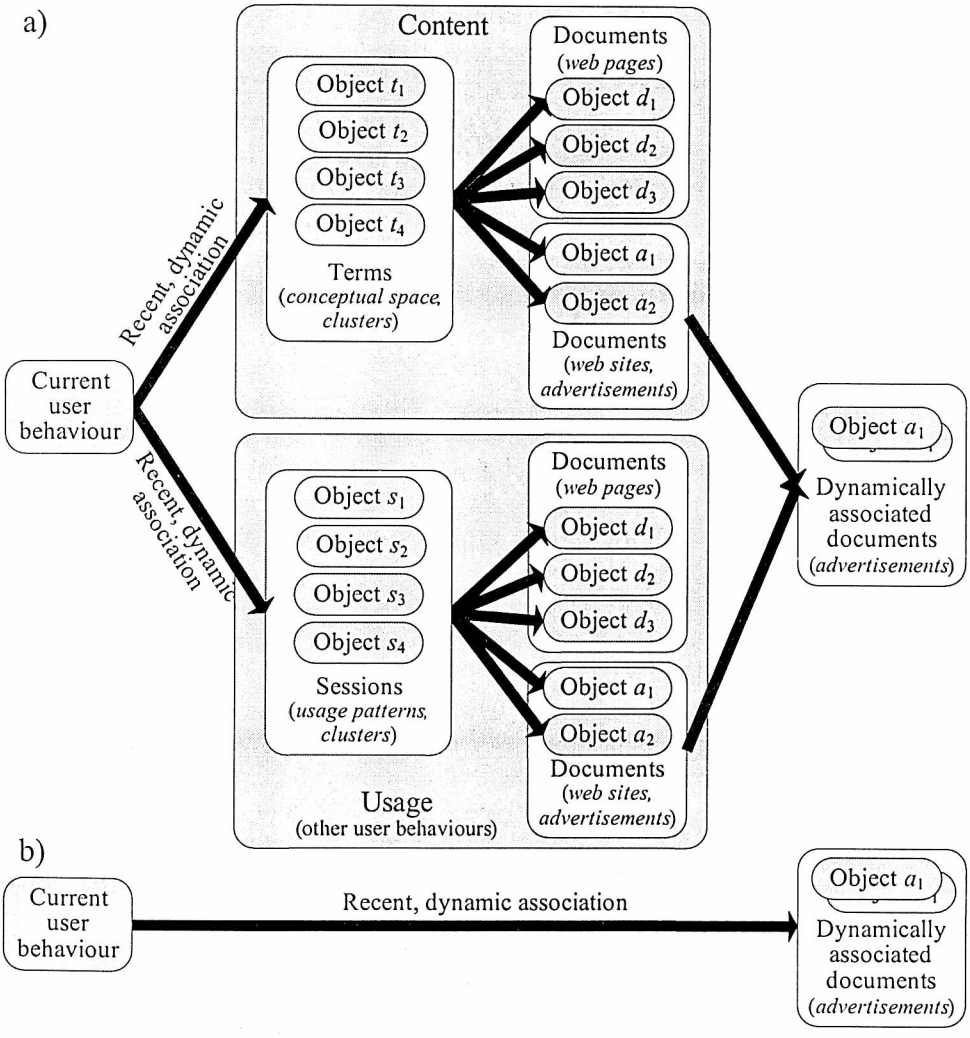

Fig. 2.5. Dynamic, personalized associations

Content-based patterns are called conceptual spaces. They are, in fact, clusters of terms extracted from the textual HTML contents of both web documents (pages) d_j the user can navigate through and the entire web sites a_k linked by advertisements considered for the attachment to the displayed web pages d_j, Fig. 2.5a. Hence, the term t_i can be more or less related to web pages and entire web

sites. Similarly to content-based associations, we also have some usage-based patterns extracted from web logs and clicked advertisements. Usage patterns are representatives (average) of clusters created from user sessions. As in the case of conceptual spaces, usage patterns are related to individual web pages in the web sites considered as well as to entire sites linked by ads.

A single user can be assigned to only one conceptual space and one usage pattern at a time; this is discussed in detail in Sec. 6.6.1. However, this association can change over time, even at each user step in the navigational path. Hence, associations are created dynamically, separate for each active user. In consequence, we obtain dynamic, personalized associations, based on which the system is able to prepare adaptive advertisements, personalized for each single user, i.e. from associations: current user–conceptual space and current user–usage pattern, some new associations are derived: current user–the most appropriate ad sites, Fig. 2.5b.

2.1.7. Associations in the Social Network

An association in the social network is the connection from one member to another that reflects their common acquaintance, private or professional relation or even high similarity of their inclinations or activities. The maintenance or even only creation of the association usually requires member's trust, commitment, emotion, or dedication of time and effort.

Several significant social features can characterize a human association, such as mutuality, durability, intensity, intentions, culture conditioning, emotional level and finally strength [Han06, Was94].

Associations between social network members can be extracted from the data about user activities like published photos [Mus08a, Mus08b], sent emails [Cul04, Kaz08c, Kaz *b*, She05], comments to published contents [Mus08a, Mus08d], etc.

Three main kinds of associations in computer-based social networks can be distinguished:

1. Direct association is the relation that connects two users with a direct connector, Fig. 2.6. For example, when one user u_1 sends emails to another user u_2, then the emails sent constitute an association from u_1 to u_2.

2. Quasi-direct association: two users are in the relationship but it is not required that they maintain the association in the direct way, e.g. two people who comment on the same blog. There is always a meeting object which serves as the communication medium between users, Fig. 2.7. Quasi-direct association can be either with equal or different roles. In the former, two users u_1 and u_2 meet each other through the meeting object and their role in relation to this object is the same: a=b. In other words, they participate in common activity related to an object with the same role. For instance, two users comment the same picture, both of them add the same object to their favourites or both use the same tags as meta-

data to describe their photos. Quasi-direct association with different roles is the association between two users u_1 and u_2 that are connected through the meeting object (e.g. multimedia object or their additional feature like tag) - they partici-

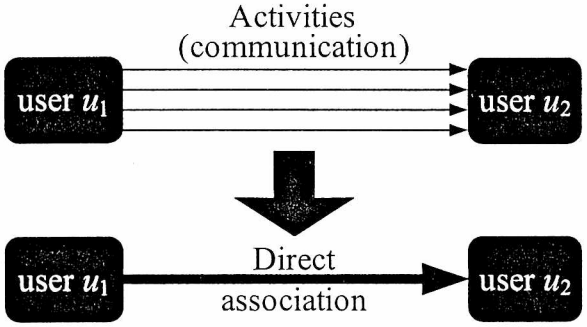

Fig. 2.6. Direct associations based on user activities

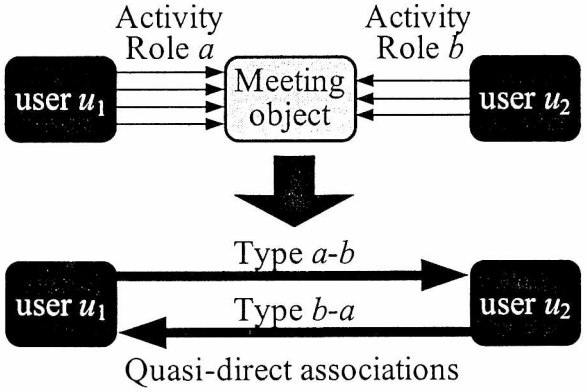

Fig. 2.7. Quasi-direct associations based on user activities

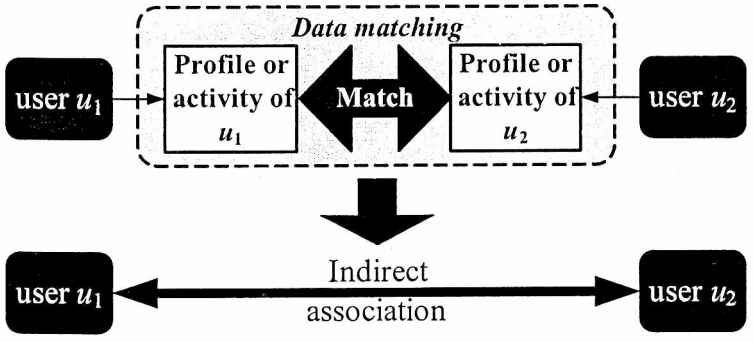

Fig. 2.8. Indirect associations based on similarity of user profiles or activities

pate in common activity but their roles a and b are different, e.g. u_1 comments a photo (role a - commentator) that has been published by u_2 (role b - author). Note that association from u_1 to u_2 differs from that from u_2 to u_1. For example, we can have two separate associations: commentator-author (type a-b) and author--commentator (type b-a). Quasi-direct associations can be applied in recommender systems to suggest new direct connections between users [Mus08d].

3. Indirect association: this kind of relations exists when the user is not aware of the fact of being similar to another user. Two users are connected by indirect association when their demographic profiles or activity profiles (behaviour) are similar, Fig. 2.8. Such associations in the context of recommender systems were considered in [Kaz06b].

It is worth noting that the direct associations can be supported and developed by utilizing the knowledge derived from the characteristic of indirect associations, e.g. the recommendation systems can use the demographic filtering and suggest some new friends who are used to watching similar movies. The usage of association matching in order to expand telecommunication social networks was studied in [Kaz07b].

In one system, many distinct association types can be discovered. For example, nine separate layers of associations were distinguished within the Flickr photo publishing system [Kaj07, Kaz08e, Mus08a, Mus08d].

Since data about user activities is quantitative, extracted associations can have values assigned - weighted associations. The association from user Bob to Alice based on 100 emails sent by Bob to Alice and these were all emails sent by Bob in the system should be much weaker than the similar association from Mary to Paul created upon Mary's single email sent to Paul among total 100 emails sent by Mary to other users. A discussion on the association weight evaluation can be found in Sec. 7.2.3 and 7.5.4.

Associations are the crucial component of social networks and the knowledge hidden inside them can be used in further analysis. An example of such utilization is described in Sec. 7: the social network data has been exploited to compute social positions of network members that reflect importance of individuals within the community.

2.2. Processes Addressed to Association Computing

Associations are usually only an element in the entire process of knowledge acquisition and processing. The standard, general process work flow is presented in Fig. 2.9. During the gathering stage source data is stored in the database or simple flat files, e.g. log files. This data usually requires some kinds of pre-processing, which includes cleansing and transformations. As a result, we obtain input data for association dis-

covery or calculation stage. Computed associations are often filtered (post-processed) in order to achieve only meaningful associations. The filtering process typically requires some external knowledge about what is useful and what is not. Afterwards, the set of selected associations that is still large is often aggregated to provide more condensed knowledge helpful in individual applications.

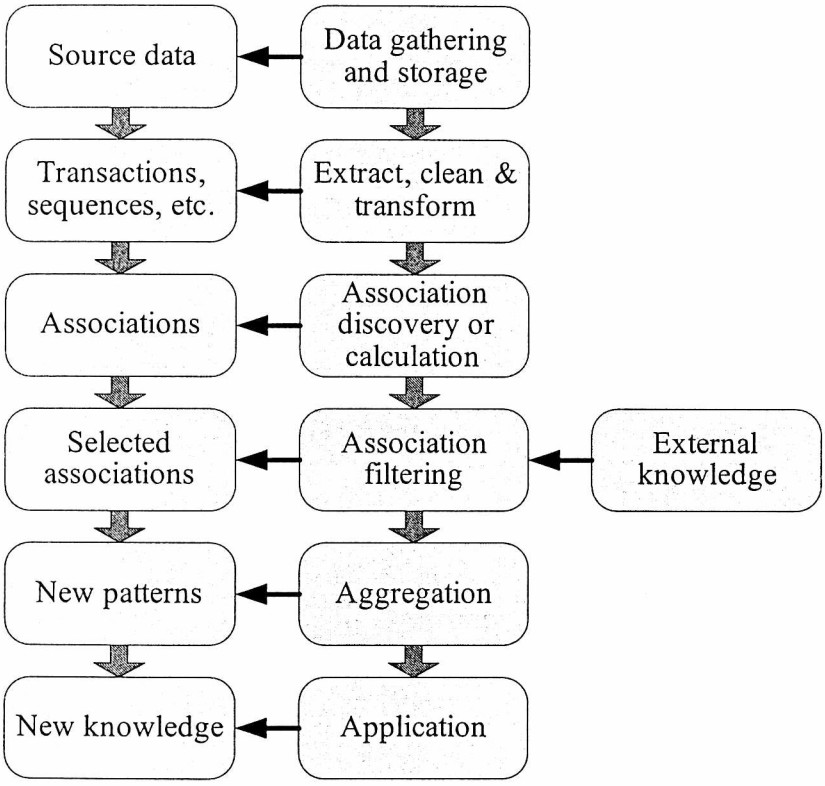

Fig. 2.9. General processes related to associations

This general process has been considered in more detail for some selected environments: in web advertising, see Sec. 2.2.1, in hyperlink verification, see Sec. 2.2.2, in social network analysis, see Sec. 2.2.3.

2.2.1. Association Processing in Personalized Web Advertising

The concept of personalized web advertising is discussed in Sec. 6. The main sources utilized in this process are as follows: textual HTML contents of web pages, web logs containing HTTP requests and pages recently visited by the current user, Fig. 2.10. From the first two some clusters are extracted. The content-based clusters correspond

to different thematic groups (conceptual spaces) existing in the site whereas groups of user sessions reflect typical user behaviours. Each content- and usage-based cluster is also correlated with individual advertisements. User-to-cluster associations are established separately for each current user and at each user request using representatives of the clusters and recent activities of the current user. Based on these associations, ads related to the assigned clusters are filtered according to some user activities, i.e. recently seen or already clicked ads are omitted. The remaining advertisements derived both from the content and usage data are integrated and again filtered in terms of advertising policy (e.g. number of permitted expositions). Finally, some top advertisements are displayed to the user in the personalized and adaptive way.

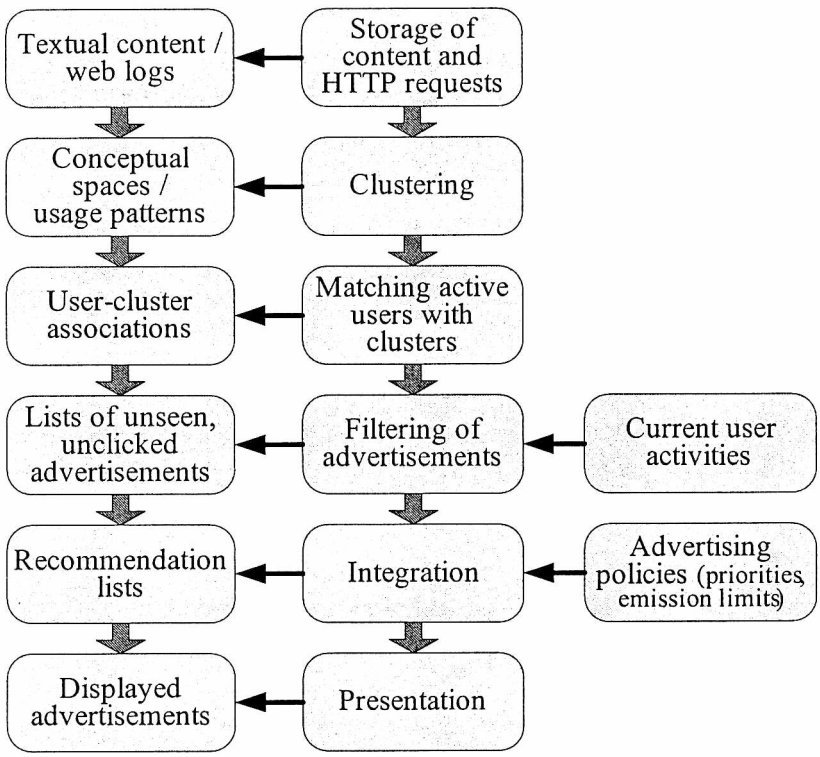

Fig. 2.10. Processes related to associations in personalized web advertising

2.2.2. Process of Hyperlink Verification Based on Associations

The idea of usage-based assessment of hyperlinks based on both positive and negative association patterns (association rules and sequential patterns) is presented in Sec. 4 and 5.5. The entire course of verification starts with pre-processing of web server logs; it regards in particular cleansing procedures, Fig. 2.11. Next, user ses-

sions (for association rules) and navigational paths (for sequential patterns) are identified and exploited for association pattern discovery using some specialized mining algorithms, see Sec. 4.3 and 5.3.2. During or just after the mining stage, the associations are filtered, and only those that match existing hyperlinks are left. Since both association rules and sequential patterns operate on many items, an aggregation procedure is needed to reduce them to 2-page recommendation functions, see Sec. 4.4.2 and 5.5.2. Additionally, different patterns (recommendation functions) can be integrated to provide more comprehensive view onto hyperlink usability. Finally, verified hyperlinks are presented to the content manager.

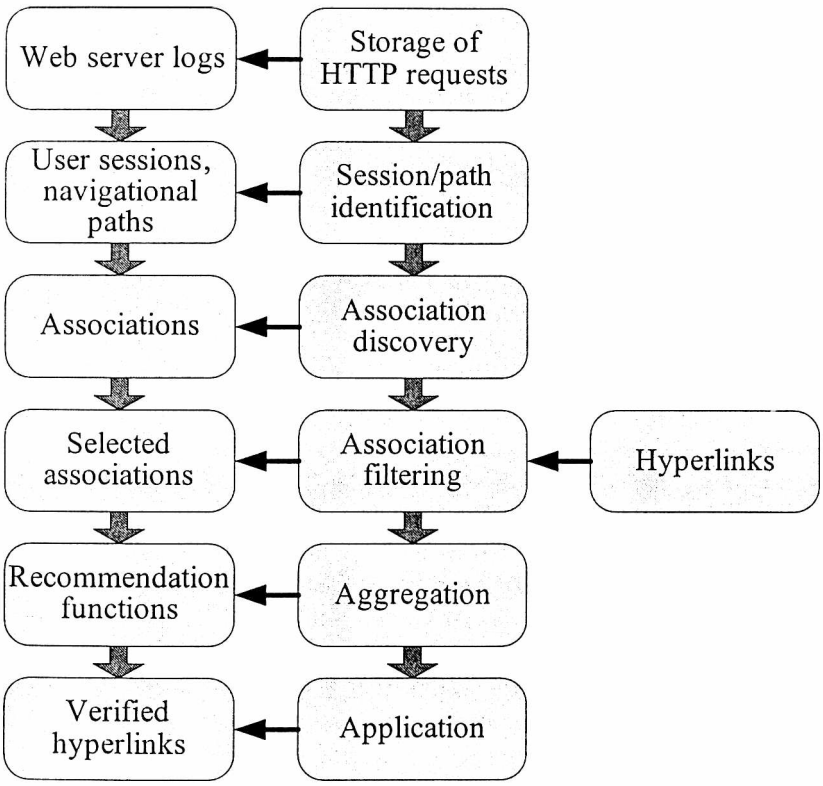

Fig. 2.11. Processes in web hyperlink verification by means of associations

2.2.3. Associations Processed in Social Network Analysis

Associations in social network analysis are extracted from the cleansed communication data, Fig. 2.12; also, some data about user activities like viewed photos in the photo publishing system or comments to blogs exposed by users. In social networks, associations are usually directly calculated from the pre-processed

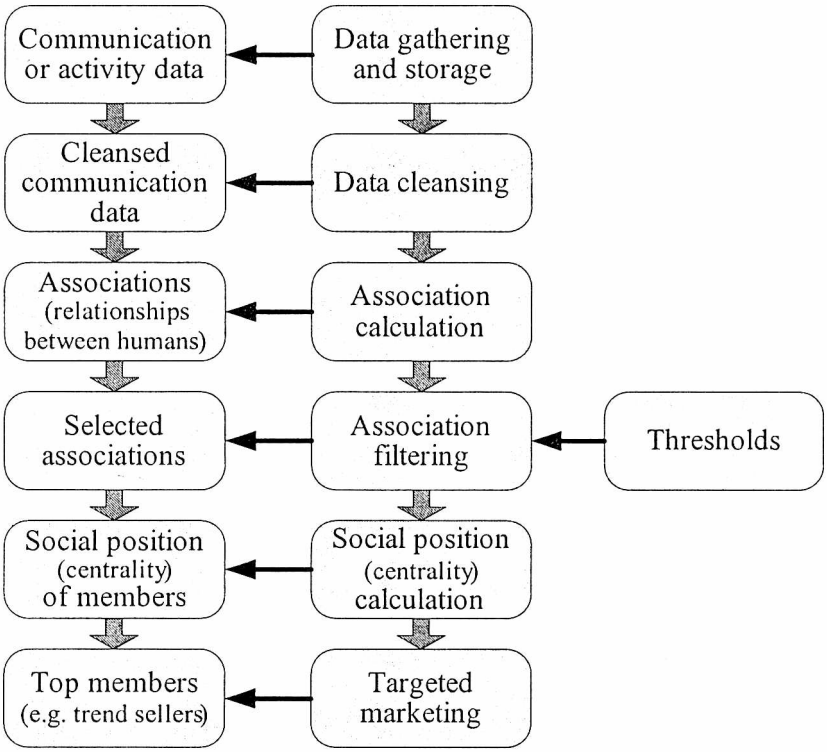

Fig. 2.12. Processes in social network analysis related to associations

communication or activity data. This is opposite to the previous cases; see Sec. 2.2.1 and 2.2.2, where associations were discovered by means of specialized algorithms. Note that associations in social networks reflect relationships between pairs of humans. Anyway, even here they can be filtered to remove some useless connections. Typically, some thresholds are used for this purpose. Filtered associations constitute the social network and are used in social network analysis methods. In one of such methods, associations are exploited to evaluate social position of individual network members - extraction of key persons, see Sec. 7. Humans with the highest social position can be used in targeted marketing as trend sellers, i.e. people who spread new products or services by means of their high influence on others.

3 Indirect Association Rules and Their Application in the Web-based Recommender Systems

Association rules mining is one of the most important and widespread data mining techniques. The association rules reflect regularities in the co-occurrence of the same items within a set of transactions. A classic example of the association rule mining is finding sets of products usually purchased together by many independent buyers. In the web environment, association rules mining is typically applied to HTTP server log data that contains historical user sessions. Hence, items from the classic basket analysis correspond to pages while transactions to user sessions. Web sessions are gathered without any user involvement and additionally, they reliably reflect user behaviour while navigating throughout a web site. For that reason, web sessions can be regarded as an important source of information about users.

Association rules that reveal similarities between web pages derived from user behaviours can be simply utilized in recommender systems.

The main goal of this section is to introduce new kind of association rules namely indirect association rules and examine their application in recommender systems.

Indirect associations exist between pages that rarely occur together but there are other, "third" pages, called transitive, with which they appear relatively frequently. Two types of indirect association rules are described in this section: partial indirect associations and complete ones. The former respect single transitive pages (see Sec. 3.4.1); while the latter cover all existing transitive pages (see Sec. 3.4.2). The IDARM* algorithm presented (see Sec. 3.6) extracts complete indirect association rules with their important measure - confidence, using pre-calculated direct rules. Both direct and indirect rules are joined into one set of complex association rules (see Sec. 3.4.5), which may be used for recommendation of web pages (see Sec. 3.8). The experiments performed revealed the usefulness of indirect rules for extension of typical recommendation lists (see Sec. 3.9). They also deliver new knowledge not available for direct ones. The relation between ranking lists created on the basis of direct association rules as well as hyperlinks existing on web pages has also been examined.

This section has been prepared based on [Kaz *a*].

3.1. Problem Description

Besides many advantages, the association rule method has also some limitations, which can result in the loss of some vital information. Typical association rules focus on the co-occurrence of items (purchased products, visited web pages, etc.) within the transaction set. A single transaction may be a payment for purchased products or services, an order with the list of items as well as a historical user session in the web portal. The mutual independence of items (products, web pages) is one of the most important assumptions of the method but it is not fulfilled in the web environment. Web pages are connected with hyperlinks and they usually determine all possible navigational paths. A user admittedly is able to enter requested page address (URL) in their browser; nevertheless most navigation is done with the help of hyperlinks designed by site authors. Thus, the web structure seriously restricts visited sets of pages (user sessions), which are not so independent of one another as products in a typical store. To reach a page the user is often forced to navigate through other pages, e.g. home page, login page, etc. Additionally, web site content is usually organized by designer into thematic blocks, which are not always suitable for particular users.

For all these reasons, some personalized recommendation mechanisms are very useful in most web portals [Mon03]. However, if they used typical association rules applied to historical user sessions [Ado01, Mob00a, Nak03, Yan03], they would often only confirm "hard" connections that simply result from hyperlinks. Moreover, such rules may avoid some relationships between pages, which do not occur together in the same user sessions. This concerns especially pages not being connected directly with hyperlinks, Fig. 3.1.

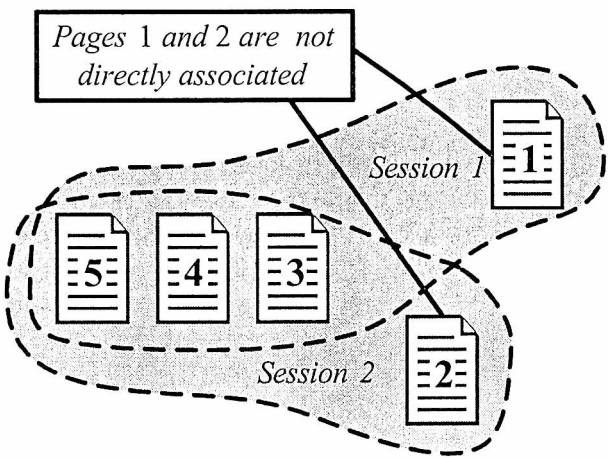

Fig. 3.1. Sessions with two documents (1 and 2), which are associated only indirectly

Original association rules, called in this section *direct*, reflect relationships existing "within" user sessions (transactions). Standard parameters of direct association rules (support and confidence) have usually the greatest value for pages "hard" connected with links due to hypertext nature of the web. To explore significant relationships between pages that rarely occur in common sessions but simultaneously they are close to other pages (Fig. 3.1), the new patterns – *indirect association rules* are suggested in this section. Two pages, each of which relatively frequently co-occurs in sessions with another, third page, can be considered as "indirect associated". Similar idea was investigated in scientific citation analysis [Goo01, Law99] and hyperlink (structure) analysis of the web [Hen01, Wei96]. Two scientific papers or web pages in which the same third document (page) is cited (linked) are supposed to be similar. A similar case occurs while two documents are cited or linked by the third one.

3.2. Background

Association rules are some of the most important and well known data mining techniques [Morz03] also in the web environment. There are many algorithms for mining association rules, see Sec. 2.1.1.

The implementation of data mining into web domain (web mining) has been considered for a couple of years [Bol99, Mad99]. Especially association rules discovered from HTTP server log data or user sessions (web usage mining) have been studied [Ado01, Mob00a, Nak03, Yan03].

Incremental algorithms appear to be most suitable for the extraction of association rules in the web domain, taking into account the nature of web user behaviour and great changeability of content and structure of the web. The problem of diversification between old and new user sessions was considered in [Kaz04b, Kaz04d].

Association rules have been utilized in many web recommendation systems, applied in various domains such as suggestions in personalized distance learning [Wang02], or consecutive steps in web navigation, i.e. hyperlink recommendation [Ger03], [Mob00a, Yan03], personalized shopping adviser [Cho02, Chu04, Ha02], extension of web searching [Bol99]. In the web environment association patterns can be extracted from server logs [Ger03, Mob00a, Wang02], purchased products [Cho02, Chu04, Ha02], or products placed into the basket [Cho02], as well as web content [Bol99]. Association rules outgoing from a certain page (fixed body of the rule) are usually ordered according to their quality measure, confidence, which enables the creation of a ranking list for this page. Chun *et al.* personalized such rankings using rule-user relevancy matrix derived from the data about products purchased by the individual user [Chu04].

Adomavicius and Tuzhilin proposed mining of personal association rules in recommender systems. It means that the individual set of rules is prepared separately for each customer based on historical behaviour of the given user. Additionally, rules are clustered and next filtered manually by the expert - administrator who is responsible for removal of useless rules [Ado01]. This approach strongly suffers from so called cold start problem - we would have nothing to recommend to new users or users with poor history. Besides, the user is presented only with items that have already attracted them. This could be useful in recommendation of options in typical IT systems, e.g. accounting, rather than in recommendation of web pages or products in e-commerce.

In another approach to recommendation, the system retains user profiles or any other kind of historical or recent information related directly to the particular user. Based on this data, we can personalize recommendations according to either past [Ado01, Chu04] or present user activities [Law01, Ger03]. Nevertheless, in this section we focus only on non-user-sensitive recommendations, which enables us to create a static list of preferred pages individually for each web page. In this way all bothersome processes are performed offline. According to legal regulations in some regions of the world, the storage of personal data (also activities) is prohibited without evident user permission [Dir02, Kob02]. In our approach no personal information about user is needed, which helps to fulfil privacy prevention constraints in anonymous web portals.

Early research work on mining indirect associations was carried out by Tan and Kumar [Tan00, Tan02b, Tan03] and next by Wan and An [Wan03, Wan06a, Wan06b]. However, their indirect patterns differ from those presented in this section. We have not assumed that two pages must not be directly correlated like Tan *et al.* did. Thus, their indirect rules reflect rather negative associations existing between items. In the approach presented in this section indirect rules are treated as an extension of direct ones, rather than as that kind of negative associations. Additionally, rules by Tan *et al.* operate on a set of transitive pages (called a mediator set) with the fixed cardinality and this set is treated as one whole. In such an approach, both pages considered have to co-occur with a complete set of other pages instead of a single transitive page. There are no partial rules in that approach either, while in the concept described below there are components of complete rules. Tan *et al.* proposed that one pair of pages may possess many indirect rules with many mediator sets, which may overlap. In web recommendation systems considered in this section we would need only one measure that helps us find out whether the page of interest should or should not be suggested to a user on the given page. There is no simple method of merging many rules to obtain such a single measure. Moreover, it appears to be much more effective to extract indirect rules from direct ones (our approach) instead of deriving them from source data (approach of Tan *et al.*).

Hamano and Sato proposed their own method to mine both negative and positive indirect association rules similar to Tan's *et al.*, using special μ measure [Ham04]. Another algorithm HI-Mine to discover Tan's *et al.* indirect association rules was suggested by Wan and An [Wan03, Wan06a]. Its slightly modified version HI-Mine* based on the compression of transaction database into Super Compact Transaction Database was presented in [Wan06b].

Chen *et al.* extended Tan's *et al.* concept of indirect rules by introducing the lifespan of items. Their temporal indirect association rules mined with MG-Growth algorithm respect temporal dependencies between transactions [Chen06].

Hao *et al.* studied visualization of indirect association rules on a spherical surface especially for marketing purposes [Hao01].

3.3. Direct Association Rules in the Web Environment

Let d_i be an independent *web page* (document) and D be the web site content (web page domain) that consists of independent web pages $d_i \in D$.

Definition 3.1. A set X of pages $d_i \in D$ is called *pageset* X. The number of pages in a *pageset* is called *the size or cardinality of the pageset* and is denoted by $card(X)$. A pageset with the length k is denoted by *k-pageset*.

Definition 3.2. The ith user session S_i is the pageset containing all pages viewed by the user during the ith visit on the web site; $S_i \subseteq D$. S^S is the superset of all user sessions gathered by the system, $S_i \in S^S$. Each session must consist of at least two pages $card(S_i) \geq 2$. A session S_i contains the pageset X if and only if $X \subseteq S_i$.

Sessions correspond to transactions in a typical data mining approach [Agr94, Morz03]. Note that pagesets and user sessions are unordered and without repetitions - we turn navigational sequences (paths) into sets. Additionally, user sessions may also be filtered to omit too short and too long ones, which are not representative enough [Kaz04d].

Definition 3.3. A *direct association rule* is the relationship $X \rightarrow Y$, where $X \subseteq D$, $Y \subseteq D$ and $X \cap Y = \emptyset$. A direct association rule is described by two measures: *support* and *confidence*. The direct association rule $X \rightarrow Y$ has the direct support:

$$sup(X \rightarrow Y) = card(\{S_i \in S^S: X \cup Y \subset S_i\}) / card(S^S). \quad (3.1)$$

The direct confidence *con* for direct association rule $X \rightarrow Y$ is the probability that the session S_i containing X also contains Y:

$$con(X \rightarrow Y) = card(\{S_i \in S^S: X \cup Y \subset S_i\}) / card(\{S_i \in S^S: X \subset S_i\}) \quad (3.2)$$

The pageset X is the *body* (or *antecedent*) and Y is the *head* (or *consequent*) of the rule $X \rightarrow Y$.

Direct association rules represent regularities discovered from a large dataset [Agr93]. The problem of mining association rules is to extract rules that are strong enough and have the support and confidence value greater than given thresholds: minimum direct support - *minsup* and minimum direct confidence - *mincon*.

In this section, we consider dependencies only between 1-pagesets - single web pages (2-pageset for both sides of the rule). For that reason, the 1-pageset X including d_i ($X=\{d_i\}$) will be denoted by d_i and a direct association rule from d_i to d_j is $d_i \rightarrow d_j$. Thus, the rule $d_i \rightarrow d_j$ is described by direct confidence function $con(d_i \rightarrow d_j)$ and direct support function $sup(d_i \rightarrow d_j)$. Similarly, Wang *et al.* restricted heads of their direct association rules in recommender system applied to distance learning domain [Wang02].

In the context of recommender systems the support function is used merely to exclude weak rules, i.e. only rules that exceed the level of minimum direct support *minsup* are considered for recommendation. In other words, support expresses the popularity of the given rule among all others. Direct confidence function $con(d_i \rightarrow d_j)$ indicates with which belief page d_j may be recommended to a user while watching the page d_i. In other words, the direct confidence factor is the conditional probability $P(d_j \mid d_i)$ that a session containing the page d_i also contains page d_j:

$$con(d_i \rightarrow d_j) = P(d_j \mid d_i) = \frac{n_{ij}}{n_i} \quad (3.3)$$

where n_{ij} - the number of sessions with both d_i and d_j; n_i - the number of sessions that contain d_i.

It was assumed that all pages are statistically independent of one another. But this is not the case. Some pages are connected by links (but most pairs are not), some were recommended by the system while other ones were not, and some are placed deeper in the web site structure. Hence, from the statistical point of view the probability value (n_{ij}/n_i) is only an approximation.

3.3.1. Weaking Older Sessions

Some page fads, which have gone a long time ago, still influence confidence value, Eq. (3.3), to the same extent as pages popular nowadays. Since many users tend to change their behaviour, we should not rely on older sessions with the

same confidence as on newer ones. If the given page d_j was visited together with page d_i many times but only in the past, then d_j should not be recommended so much at present. For that reason, the introduction of the time factor is proposed. Numbers of sessions n_{ij} and n_i in Eq. (3.3) are replaced with the time weighted numbers of sessions: n'_{ij} and n'_i, respectively, as follows:

$$con^t(d_i \rightarrow d_j) = \frac{n'_{ij}}{n'_i} = \frac{\sum_{s:s\in S;\, d_i,d_j \in s} (\tau)^{tp(s)}}{\sum_{s:s\in S;\, d_i \in s} (\tau)^{tp(s)}} \tag{3.4}$$

where $con^t(d_i \rightarrow d_j)$ - time weighted direct confidence; τ - the constant time coefficient from the interval [0,1]; $tp(s)$ - the number of time periods since the beginning of session s until the processing time.

In other words, while calculating n'_{ij} and n'_i each session s_k, unlike n_{ij} and n_i, is counted not as 1 but as $(\tau)^{tp(s)}$. The time period length - a unit of measure for $tp(s)$ - depends on how often users enter the web site. The time coefficient τ denotes changeability of the site content and behaviour of users. The more often the site changes, the smaller the τ value should be. In this way, older sessions have less influence on recommendation results.

3.3.2. Case Study

Let us consider an example set of 10 user sessions within the web site that consists of six pages, $D=\{d_1,d_2,d_3,d_4,d_5,d_6\}$ - Table 3.1. The result of mining direct association rules for single web pages $(d_i \rightarrow d_j)$ within the example sessions is a set of rules (Table 3.2) that can be represented as a directed, cyclic graph (Fig. 3.2); *minsup*=20% and *mincon*=40% were assumed. Nodes of the graph correspond to web pages and edges indicate direct associations. An edge weight is equivalent to the value of the appropriate rule confidence. A page can be the body as well as the head of the rule. Each node has two values assigned: v^+_k and v^-_k denoting the number of rules for which d_k is the body $(d_k \rightarrow d_j)$ and head of rules $(d_i \rightarrow d_k)$, respectively.

Table 3.1. Example user sessions

Session id	Pages	Session id	Pages
1	d_1,d_2,d_4	6	d_2,d_4
2	d_1,d_4	7	d_4,d_5,d_6
3	d_1,d_2,d_4	8	d_2,d_4,d_5,d_6
4	d_1,d_3	9	d_1,d_6
5	d_2,d_4,d_5,d_6	10	d_1,d_3

Table 3.2. Values of direct confidence for example sessions from Table 3.1

No.	Rule	con	No.	Rule	con
1	$d_1 \rightarrow d_4$	0.50	9	$d_4 \rightarrow d_5$	0.43
2	$d_2 \rightarrow d_1$	0.40	10	$d_4 \rightarrow d_6$	0.43
3	$d_2 \rightarrow d_4$	1.00	11	$d_5 \rightarrow d_2$	0.67
4	$d_2 \rightarrow d_5$	0.40	12	$d_5 \rightarrow d_4$	1.00
5	$d_2 \rightarrow d_6$	0.40	13	$d_5 \rightarrow d_6$	1.00
6	$d_3 \rightarrow d_1$	1.00	14	$d_6 \rightarrow d_2$	0.50
7	$d_4 \rightarrow d_1$	0.43	15	$d_6 \rightarrow d_4$	0.75
8	$d_4 \rightarrow d_2$	0.71	16	$d_6 \rightarrow d_5$	0.75

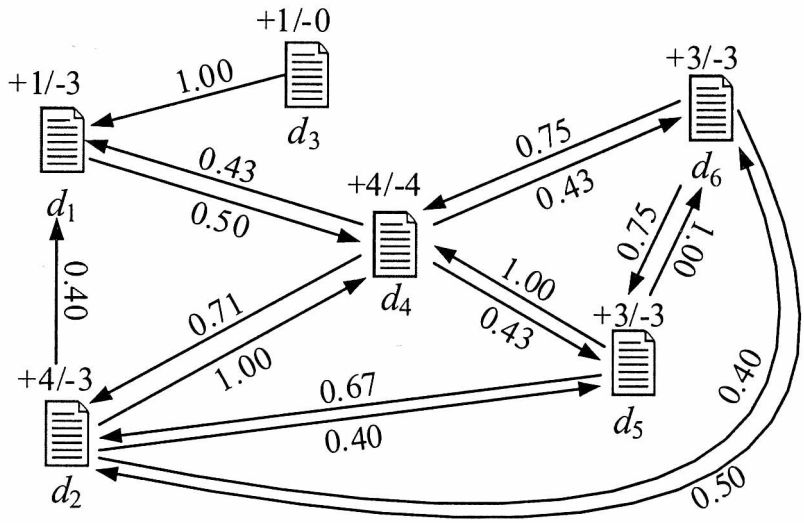

Fig. 3.2. Graph with direct association rules extracted from example sessions (Table 3.1)

3.4. Indirect and Complex Association Rules in the Web

Let us consider another approach to associations: indirect association rules.

3.4.1. Partial Indirect Association Rules

Definition 3.4. *Partial indirect association rule* $d_i \rightarrow^{P\#} d_j, d_k$ is the *indirect* relationship from d_i to d_j with respect to d_k, for which two direct association rules exist: $d_i \rightarrow d_k$

and $d_k \rightarrow d_j$ with $sup(d_i \rightarrow d_k) \geq minsup$, $con(d_i \rightarrow d_k) \geq mincon$ and $sup(d_k \rightarrow d_j) \geq minsup$, $con(d_k \rightarrow d_j) \geq mincon$, where $d_i, d_j, d_k \in D$; $d_i \neq d_j \neq d_k$. The page d_k, in the partial indirect association rule $d_i \rightarrow^{P\#} d_j, d_k$, is called *the transitive page* (Fig. 3.3).

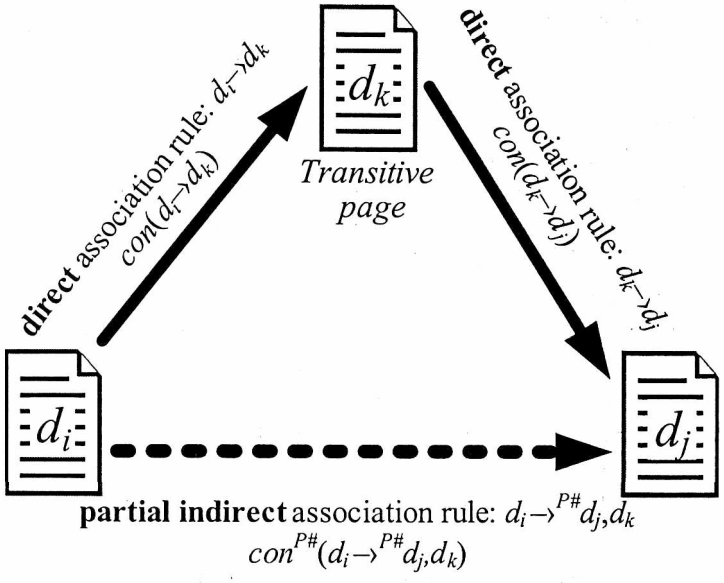

Fig. 3.3. Indirect association between two web pages

Note that there may be many transitive pages d_k for a given pair of pages d_i, d_j and as a result many partial indirect association rules $d_i \rightarrow^{P\#} d_j, d_k$.

Each indirect association rule is described by *partial indirect confidence* $con^{P\#}(d_i \rightarrow^{P\#} d_j, d_k)$, as follows:

$$con^{P\#}(d_i \rightarrow^{P\#} d_j, d_k) = con(d_i \rightarrow d_k) * con(d_k \rightarrow d_j) \tag{3.5}$$

Partial indirect confidence is calculated using direct confidence rather than source user session data. For that reason, the computational complexity of only partial indirect rule mining is much less than for direct ones - see the description of IDARM* algorithm in Sec. 3.6.2. However, note that it refers only to the process of mining indirect rules from direct ones. Obviously, these direct rules need to be previously extracted and the entire process of indirect rule mining consists of two consecutive steps: direct rule mining and indirect rule mining. Nevertheless, the second step is less complex than the first one.

Pages d_i, d_j in $d_i \rightarrow^{P\#} d_j, d_k$ do not need to have any common sessions, but in Eq. (3.5) we respect only "good" direct associations to ensure that indirect associations are based on sensible grounds. From questionable or uncertain direct knowledge we should not derive reasonable indirect knowledge. In consequence,

it was assumed that rules $d_i \rightarrow d_k$ and $d_k \rightarrow d_j$ must be "strong" enough so that $con(d_i \rightarrow d_k)$ and $con(d_k \rightarrow d_j)$ exceed *mincon*.

Some other functions instead of multiplication in Eq. (3.5) like minimum, maximum, arithmetical mean and weighted mean were considered in [Kaz05e]. Multiplication delivers the smallest values (on average even 1/10 compared to values of maximum function) but it has the best discrimination abilities at the same time - the standard deviation doubles the average while for other functions standard deviation is less than the average.

A partial indirect rule $d_i \rightarrow^{P\#} d_j, d_k$ reflects one indirect association existing between d_i and d_j, so no direct association $d_i \rightarrow d_j$ is needed, even though it may exist. The condition of non-existence of direct association is prior assumption in indirect rules proposed by Tan *et al.* in [Tan00, Tan02b, Tan03] and next used by Wan and An in [Wan03, Wan06a, Wan06b].

The rule $d_i \rightarrow^{P\#} d_j, d_k$ also differs from two direct rules: $\{d_i, d_k\} \rightarrow d_j$, and $d_i \rightarrow \{d_j, d_k\}$. Note that these direct rules respect only common user sessions that contain all three pages d_i, d_j, d_k. On the contrary, the partial indirect rule $d_i \rightarrow^{P\#} d_j, d_k$ exploits common sessions of d_i, d_k and separately sessions with d_k, d_j. These two sets of sessions do not even need to overlap.

Since the component direct rules $d_i \rightarrow d_k$ and $d_k \rightarrow d_j$ are directed, also the partial indirect rule $d_i \rightarrow^{P\#} d_j, d_k$ is directed, i.e. $d_i \rightarrow^{P\#} d_j, d_k$ differs from $d_j \rightarrow^{P\#} d_i, d_k$. In consequence, partial indirect confidence function is not symmetric, which means that $con^{P\#}(d_i \rightarrow^{P\#} d_j, d_k)$ does not have to be equal to $con^{P\#}(d_j \rightarrow^{P\#} d_i, d_k)$.

Definition 3.5. The set of all possible transitive pages d_k for which partial indirect association rules from d_i to d_j exist, is called T_{ij}.

Note that T_{ij} is not the same set as T_{ji}.

3.4.2. Aggregation of Partial Rules

Definition 3.6. *Complete indirect association rule* $d_i \rightarrow^{\#} d_j$ aggregates all partial indirect association rules from d_i to d_j with respect to all existing transitive pages $d_k \in T_{ij}$ (Fig. 3.4) and is characterized by *complete indirect confidence* - $con^{\#}(d_i \rightarrow^{\#} d_j)$:

$$con^{\#}(d_i \rightarrow^{\#} d_j) = \frac{\sum_{d_k \in T_{ij}} con^{P\#}\left(d_i \rightarrow^{P\#} d_j, d_k\right)}{max_T} \tag{3.6}$$

where $max_T = \max_{d_i, d_j \in D} (card(T_{ij}))$ - the maximal number of component partial rules for a pair of pages.

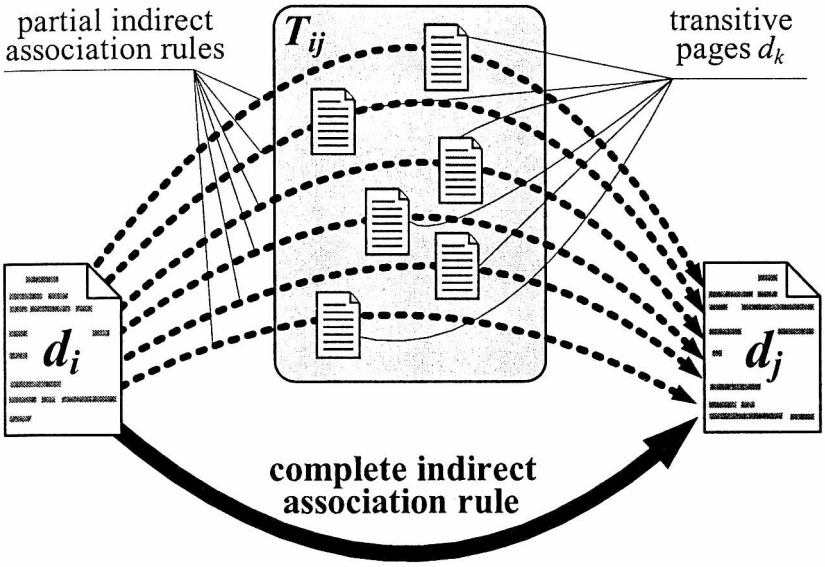

Fig. 3.4. Complete indirect association rule

A complete indirect association rule from d_i to d_j exists if and only if there exists at least one partial indirect association rule from d_i to d_j, i.e. $T_{ij} \neq \emptyset$.

Only indirect rules with complete indirect confidence greater than the given confidence threshold - *iconmin* are accepted. According to Eq. (3.5), there is no point in setting *iconmin* with the value less than the square of the appropriate threshold for direct rules divided by max_T: $iconmin \geq mincon^2 / max_T$.

Complete indirect association rules are not symmetric: the rule $d_i \rightarrow^{\#} d_j$ may exist but the reverse one $d_j \rightarrow^{\#} d_i$ not necessarily. This results from the features of partial indirect associations and direct associations, which are also not symmetric.

The concept of partial indirect rules, Eq. (3.5), enables the introduction of the threshold to partial indirect confidence - *piconmin* to exclude weak partial rules. However, *iconmin* is more general than *piconmin* so the former appears to be a more suitable filtering factor.

The normalization - the denominator max_T in Eq. (3.6) - ensures the range [0,1] to be the domain for complete indirect confidence. However, it also causes the most complete confidence values to be less than equivalent direct ones. max_T represents "global" normalization, while using $card(T_{ij})$ in the denominator we would obtain "local" normalization. Values of complete confidence are on average more than 10 times less at global normalization than at local one. According to experiments performed in the real e-commerce environment (4,242 web pages, 16,127 user sessions) typical value of max_T is about 250 while the average $card(T_{ij})$ is about 10-20, depending on *minsup*.

3.4.3. Transitive Sets

The concept of partial indirect rules with single transitive page can be quite easily extended to indirect rules with the set of transitive elements. In such an approach we need to replace the single page d_k with the K-element set of pages D_K. Thus, we can modify Definition 3.4.

Definition 3.7. *Partial indirect association rule with the set of transitive elements* $d_i \rightarrow^{P\#} d_j, D_K$ is the indirect relationship from d_i to d_j with respect to the set D_K, for which two direct association rules exist: $d_i \rightarrow D_K$ and $D_K \rightarrow d_j$ with $sup(d_i \rightarrow D_K) \geq minsup$, $con(d_i \rightarrow D_K) \geq mincon$ and $sup(D_K \rightarrow d_j) \geq minsup$, $con(D_K \rightarrow d_j) \geq mincon$, where $d_i, d_j \in D$; $D_K \subset D$; $d_i, d_j \notin D_K$; $d_i \neq d_j$.

Note that no change is needed in Eq. (3.5). Nevertheless, the conversion of transitive pages into sets has significant consequences. The way of combination of all partial rules consistent with Definition 3.7 into complete indirect rules (Definition 3.6) is not obvious due to the potential existence of many partial rules with transitive sets of different cardinalities (sizes). Naturally, these sets would often overlap one another and they even cover each other. For every set D_K of cardinality K we have total 2^K-2 proper and non-empty subsets $D_k \subset D_K$ and the same number of different partial rules $d_i \rightarrow^{P\#} d_j, D_k$ that have something in common with $d_i \rightarrow^{P\#} d_j, D_K$.

3.4.4. Case Study

Extracting complete indirect association rules for the example direct rule set (Table 3.2, Fig. 3.2), we obtain a set of complete indirect association rules from Table 3.3.

Table 3.3. Values of complete indirect confidence for example sessions from Table 3.1

No.	Rule	$con^\#$	No.	Rule	$con^\#$
1	$d_1 \rightarrow^\# d_2$	0.12	11	$d_4 \rightarrow^\# d_5$	0.20
2	$d_1 \rightarrow^\# d_5$	0.07	12	$d_4 \rightarrow^\# d_6$	0.24
3	$d_1 \rightarrow^\# d_6$	0.07	13	$d_5 \rightarrow^\# d_1$	0.23
4	$d_2 \rightarrow^\# d_1$	0.14	14	$d_5 \rightarrow^\# d_2$	0.40
5	$d_2 \rightarrow^\# d_4$	0.30	15	$d_5 \rightarrow^\# d_4$	0.47
6	$d_2 \rightarrow^\# d_5$	0.24	16	$d_5 \rightarrow^\# d_6$	0.23
7	$d_2 \rightarrow^\# d_6$	0.28	17	$d_6 \rightarrow^\# d_1$	0.17
8	$d_3 \rightarrow^\# d_4$	0.17	18	$d_6 \rightarrow^\# d_2$	0.35
9	$d_4 \rightarrow^\# d_1$	0.10	19	$d_6 \rightarrow^\# d_4$	0.42
10	$d_4 \rightarrow^\# d_2$	0.17	20	$d_6 \rightarrow^\# d_5$	0.17

Its graph representation is shown in Fig. 3.5. Edge weights indicate appropriate complete indirect confidence values; max_T=3, *iconmin*=6%. Complete indirect rules not having corresponding direct ones, i.e. the new associations, are presented with dotted line, e.g. $d_1 \rightarrow^{\#} d_2$, $d_6 \rightarrow^{\#} d_1$, etc.

Note that also some direct rules do not possess equivalent indirect ones, e.g. $d_1 \rightarrow d_4$, $d_3 \rightarrow d_1$ (compare Fig. 3.2 and 3.5). Hence, as we can see, direct and indirect rules may complement each other.

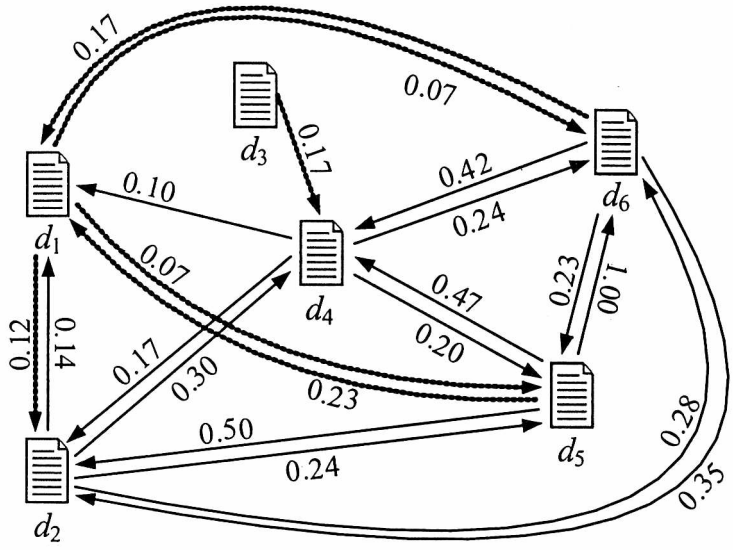

Fig. 3.5. Graph with complete indirect association rules.
Dotted lines represent new associations

3.4.5. Complex Association Rules in the Web

To make use of both direct and indirect association rules for recommendation of web pages, the joined, complex association rules are introduced. A complex association rule exists if at least one of two component rules exists, i.e. either direct (Fig. 3.6a) or complete indirect (Fig. 3.6b) or both of them (Fig. 3.6c). The main quality features of both direct and indirect rules - confidences are combined within complex association rules. The extraction of complex rules is the third stage of the whole process of rule discovery for recommender systems (Fig. 3.7).

Definition 3.8. *Complex association rule* $d_i \rightarrow^{*} d_j$ from d_i to d_j exists if direct $d_i \rightarrow d_j$ or complete indirect $d_i \rightarrow^{\#} d_j$ association rule from d_i to d_j exists. A complex association rule is characterized by *complex confidence* - $con^{*}(d_i \rightarrow^{*} d_j)$, as follows:

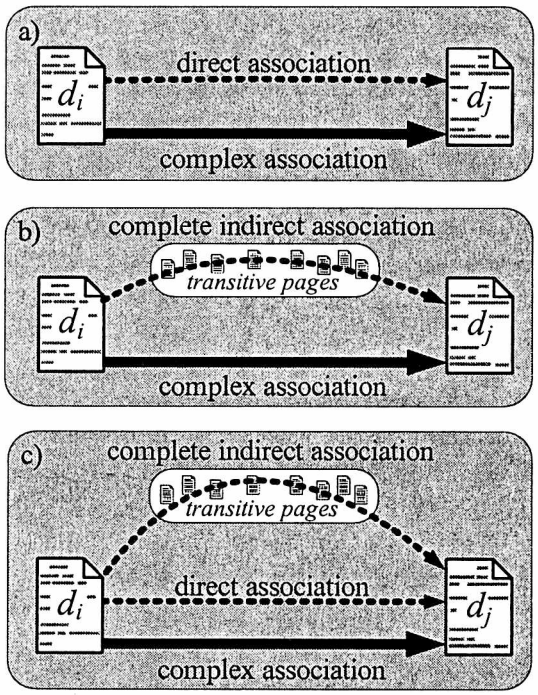

Fig. 3.6. Complex association results from either direct association (a) or complete indirect one (b) or both of them (c)

$$con^*(d_i \rightarrow^* d_j) = \alpha * con(d_i \rightarrow d_j) + (1-\alpha) * con^{\#}(d_i \rightarrow^{\#} d_j) \tag{3.7}$$

where α - direct confidence reinforcing factor, $\alpha \in [0,1]$.

Theorem 3.1. The value of complex confidence is between its component direct and complete indirect confidence, i.e. we have two possible cases:

1) $con \le con^* \le con^{\#}$, if $con \le con^{\#}$.

2) $con^{\#} \le con^* \le con$, if $con > con^{\#}$.

For better transparency arguments $(d_i \rightarrow^* d_j)$, $(d_i \rightarrow d_j)$, and $(d_i \rightarrow^{\#} d_j)$ were omitted in $con^*(d_i \rightarrow^* d_j)$, $con(d_i \rightarrow d_j)$, and $con^{\#}(d_i \rightarrow^{\#} d_j)$ respectively.

Proof

1) $con \le con^{\#} \Rightarrow \exists\, (\delta \in [0,1])\ (con^{\#} = con + \delta \Leftrightarrow con = con^{\#} - \delta)$

$con^* = \alpha * con + (1 - \alpha) * (con + \delta) = (\alpha + 1 - \alpha) * con + (1 - \alpha) * \delta = con + (1 - \alpha) * \delta$

$(1 - \alpha) * \delta \in [0,1] \Rightarrow con^* \ge con$

$con^* = \alpha * (con^{\#} - \delta) + (1 - \alpha) * con^{\#} = (\alpha + 1 - \alpha) * con^{\#} - \alpha * \delta = con^{\#} - \alpha * \delta$

$\alpha * \delta \in [0,1] \Rightarrow con^* \le con^{\#}$

2) Similarly to 1) Q.E.D.

Setting α we can emphasize or damp the direct confidence at the expense of the complete indirect one. The greater the value of α, the closer the complex confidence to the direct one is.

Example values of complex confidence are presented in Table 3.4. They are derived from component values: direct confidences (Table 3.2) and complete indirect confidences (Table 3.3). Since a complex rule exists if any of its two component rules exist, the number of complex rules is greater or equal to the number of both direct and complete indirect rules.

Note that complex association rules do not possess support feature. Only complex confidence, Eq. (3.7), as their quality measure is used. Support values are solely exploited in the filtering of the reasonable direct rules, which are components of both partial indirect association rules (see Sec. 3.4.1) and complex ones.

3.5. Ranking Lists Based on Complex Rules

In the typical item-to-item approach to recommendation based on association rules, ranking lists are created from the entire set of direct rules $d_i \rightarrow d_j$ that exceed minimum confidence and minimum support level, e.g. [Chu04, Ger03]. Pages d_j from all rules $d_i \rightarrow d_j$ outgoing from d_i are considered in creation of recommendation ranking lists for page d_i. These rules, and in consequence their consequents d_j, are ordered according to the appropriate rule quality measure. Complex confidence is utilized as such a ranking function facilitating recommendation proper (Fig. 3.7). In this way, we can make use of both direct and indirect associations. The greater the value of $con^*(d_i \rightarrow^* d_j)$ for page d_j, the higher the position of page d_j in the ranking list for the given page d_i. Usually, M top documents d_j from the ranking list, with the highest value of $con^*(d_i \rightarrow^* d_j)$, are recommended on page d_i.

Since a complex rule exists if either direct or indirect association exists, we can expect that recommendation ranking list based on complex rules often will be longer than typical rankings based on exclusively direct rules. This is also visible in Table 3.5, in which complex rules successfully replenish typical ranking lists created upon direct confidence, e.g. for page d_1 and d_3. This happens in the case of the separate set of indirect rules compared to direct ones. As complex rules join direct and indirect ones the complex rankings unite direct and indirect rankings, e.g. for page d_1, we have: direct ranking - (d_4), indirect one - (d_2,d_5), and complex one - (d_2,d_4,d_5).

The adjustment of α in Eq. (3.7) enables the contribution of both direct and indirect component to be tailored. This can result in the different order of the final ranking for different values of α. For example, in rankings for d_4, d_5, d_6, the small value of α=0.2 stresses indirect rules that change the second position in the rankings.

Table 3.4. Values of complex confidence for example sessions (Table 3.1) with various values of α. "+" and "–" denote the existence and nonexistence of the given rule, respectively

No.	Rule	Direct $d_i \rightarrow d_j$	Indirect $d_i \rightarrow^{\#} d_j$	Complex: $con^{*}(d_i \rightarrow^{*} d_j)$							
				α=0.2	α=0.3	α=0.4	α=0.5	α=0.6	α=0.7	α=0.8	α=0.9
1	$d_1 \rightarrow^{*} d_2$	–	+	0.10	0.08	0.07	0.06	0.05	0.04	0.02	0.01
2	$d_1 \rightarrow^{*} d_4$	+	–	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45
3	$d_1 \rightarrow^{*} d_5$	–	+	0.06	0.05	0.04	0.04	0.03	0.02	0.01	0.01
4	$d_1 \rightarrow^{*} d_6$	–	+	0.06	0.05	0.04	0.04	0.03	0.02	0.01	0.01
5	$d_2 \rightarrow^{*} d_1$	+	+	0.19	0.22	0.25	0.27	0.30	0.32	0.35	0.37
6	$d_2 \rightarrow^{*} d_4$	+	+	0.44	0.51	0.58	0.65	0.72	0.79	0.86	0.93
7	$d_2 \rightarrow^{*} d_5$	+	+	0.27	0.29	0.31	0.32	0.34	0.35	0.37	0.38
8	$d_2 \rightarrow^{*} d_6$	+	+	0.30	0.31	0.33	0.34	0.35	0.36	0.38	0.39
9	$d_3 \rightarrow^{*} d_1$	+	–	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90
10	$d_3 \rightarrow^{*} d_4$	–	+	0.13	0.12	0.10	0.08	0.07	0.05	0.03	0.02
11	$d_4 \rightarrow^{*} d_1$	+	+	0.16	0.20	0.23	0.26	0.30	0.33	0.36	0.40
12	$d_4 \rightarrow^{*} d_2$	+	+	0.28	0.33	0.39	0.44	0.50	0.55	0.60	0.66
13	$d_4 \rightarrow^{*} d_5$	+	+	0.25	0.27	0.29	0.32	0.34	0.36	0.38	0.41
14	$d_4 \rightarrow^{*} d_6$	+	+	0.28	0.30	0.31	0.33	0.35	0.37	0.39	0.41
15	$d_5 \rightarrow^{*} d_1$	–	+	0.19	0.16	0.14	0.12	0.09	0.07	0.05	0.02
16	$d_5 \rightarrow^{*} d_2$	+	+	0.46	0.48	0.51	0.54	0.56	0.59	0.61	0.64
17	$d_5 \rightarrow^{*} d_4$	+	+	0.58	0.63	0.68	0.74	0.79	0.84	0.89	0.95
18	$d_5 \rightarrow^{*} d_6$	+	+	0.39	0.46	0.54	0.62	0.69	0.77	0.85	0.92
19	$d_6 \rightarrow^{*} d_1$	–	+	0.14	0.12	0.10	0.09	0.07	0.05	0.03	0.02
20	$d_6 \rightarrow^{*} d_2$	+	+	0.38	0.39	0.41	0.42	0.44	0.45	0.47	0.48
21	$d_6 \rightarrow^{*} d_4$	+	+	0.48	0.52	0.55	0.58	0.62	0.65	0.68	0.72
22	$d_6 \rightarrow^{*} d_5$	+	+	0.29	0.35	0.40	0.46	0.52	0.58	0.63	0.69

Table 3.5. Ranking lists created upon: direct confidence (Table 3.2), complete indirect confidence (Table 3.3) and complex confidence values (Table 3.4) for various α

Page	Direct	Indirect	Complex							
			0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
d_1	d_4	d_2,d_5	d_2,d_4,d_5	d_4,d_2,d_5						d_4,d_5,d_2
d_2	$d_4,\{d_6,d_5,d_1\}$	d_4,d_6,d_5,d_1								
d_3	d_1	d_4	d_1,d_4							
d_4	$d_2,\{d_1,d_5,d_6\}$	d_6,d_5,d_2,d_1	d_2,d_6,d_5,d_1	d_2,d_6,d_5,d_1						d_2,d_5,d_6,d_1
d_5	$\{d_4,d_6\},d_2,d_1$	$\{d_4,d_2\},\{d_1,d_6\}$	d_4,d_2,d_6,d_1			d_4,d_6,d_2,d_1				
d_6	$\{d_4,d_5\},d_2$	$d_4,d_2,\{d_1,d_5\}$	d_4,d_2,d_1,d_5	d_4,d_2,d_5,d_1			d_4,d_5,d_2,d_1			

Note that ranking lists are static, even though they are periodically recalculated. Their content depends on the behaviour of users visiting the web site in the past (they are extracted from historical user sessions), but they are not adapted to the current user activities. Nevertheless, the obtained candidates for recommendation may be used as the source for further processing, the goal of which would be to receive the individual lists, more suitable for particular users. A pretty simple but very useful approach to personalization is the introduction of rotation mechanism. It excludes from the ranking list those pages that have already been suggested to the active user on the previous page or several pages before.

3.6. Mining Indirect Association Rules, IDARM* Algorithm

3.6.1. Stages of Association Rules Mining for Recommendations

The discovery of indirect rules is performed in two main stages (Fig. 3.7): extraction of direct rules and mining indirect ones. Afterwards, the third stage joins rules of both types into complex association rules, useful for ranking lists.

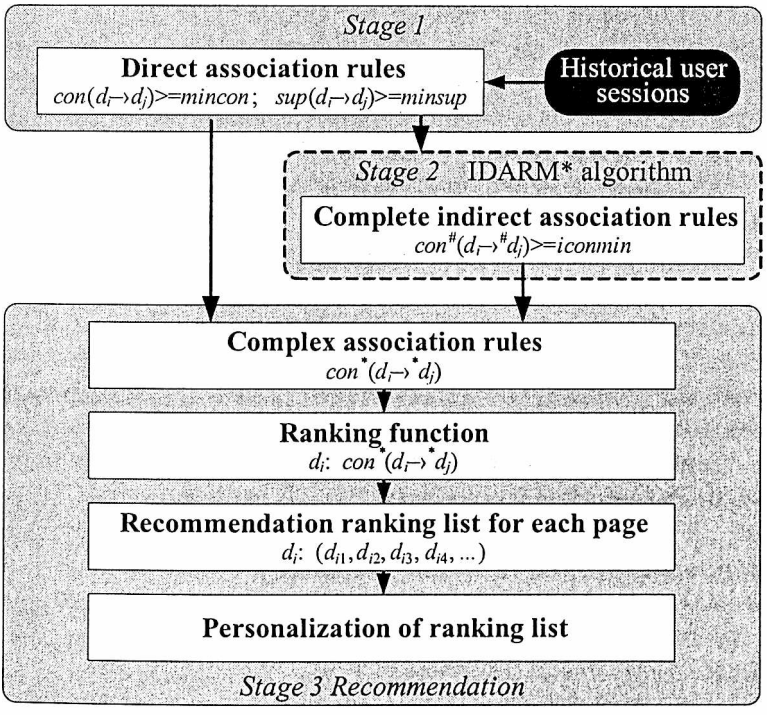

Fig. 3.7. Process of discovering association rules for recommendation

There are many algorithms to mine direct association rules, see Sec. 2.1.1. Overall, two main approaches were distinguished: the horizontal and vertical one [Morz03]. Since in the approach presented, we consider only simple direct rules: between 1-pagesets, i.e. single web pages, the choice between horizontal and vertical mining is not crucial. Nevertheless, we need to apply any algorithm for direct association rule mining at the first stage of the whole process. Taking into account the environment (sessions of web users), it is incremental algorithms that are most suitable [Cheu96, Cheu97, Lee01, Yen96].

Due to frequent modifications of web pages, especially hyperlinks, typical user behaviour, i.e. typical user sessions, tend to change over time. For that reason, the inclusion of time factor into direct rule mining appears to be justified: older sessions are damp during confidence calculation, according to how much time went by between the beginning of session and the processing time (see Sec. 3.3.1).

3.6.2. The IDARM* Algorithm

The IDARM* algorithm (*In-Direct Association Rules Miner*) was introduced to discover complete indirect association rules $d_i \rightarrow^{\#} d_j$ and their complete indirect confidence $con^{\#}(d_i \rightarrow^{\#} d_j)$ from the set of direct rules $d_i \rightarrow d_j$ according to Eq. (3.5) and (3.6). Proper input direct rules, i.e. those that exceed *minsup* and *mincon*, are previously extracted using one of the well known mining algorithms. The IDARM* algorithm makes up the second stage in recommendation process based on association rules (Fig. 3.7). Its general concept is presented in Fig. 3.8.

The IDARM* Algorithm

Input: L_1 - set of all direct rules, $sup(d_i \rightarrow d_j) > minsup$, $con(d_i \rightarrow d_j) > mincon$
$L^{IR}=\emptyset$ - list of complete indirect rules with their confidences
$L^{T}=\emptyset$ - list of numbers of transitive pages $l^{T}_{ij}=card(T_{ij})$ for each complete indirect rule $d_i \rightarrow^{\#} d_j$
Output: full list complete indirect rules L^{IR}
full list of transitive pages L^{T}

1. sort L_1 *by antecedents - create new list* L_2
2. *for each rule* $d_i \rightarrow d_k \in L_1$ do {
3. select *list* L_k *of rules* $d_k \rightarrow d_j$ *from* L_2, $d_j \neq d_i$
4. if $L_k \neq \emptyset$ then {
5. for *each rule* $d_k \rightarrow d_j \in L_k$ do {
6. if *exists complete rule* $d_i \rightarrow^{\#} d_j \in L^{IR}$ then

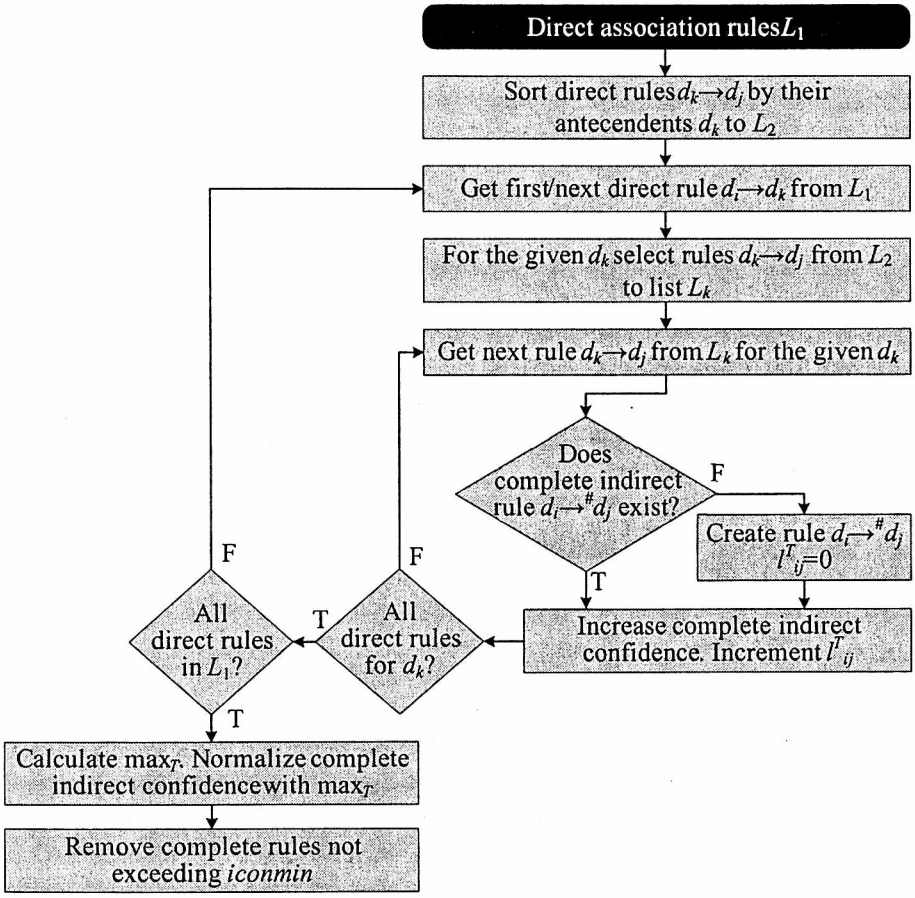

Fig. 3.8. The idea of the IDARM* algorithm

```
 7.        con#(di→#dj) = con#(di→#dj) + con(di→dk) * con(dk→dj)
 8.        lTij = lTij + 1
 9.      else {
10.        create new complete indirect rule di→#dj in LIR with
               con#(di→#dj) = con(di→dk) * con(dk→dj)
11.        create new element (number) in LT: lTij = 1
12.      }
13.    }
14.  }
15. }
16. select maxT = max (lTij∈LT)
17. for each complete indirect rule di→#dj in LIR do {
18.    con#(di→#dj) = con#(di→#dj) / maxT
```

```
19.   remove rules d_i→#d_j from L^IR for which con#(d_i→#d_j)<iconmin;
20.   remove the corresponding l^T_ij from L^T
21. }
```

Sorting in the first line and its outcome (list L_2) are used only to speed up the selection (line 3) and the internal loop (lines 5–13).

L_k is the list of all rules with the fixed d_k as antecedent (line 3). To fulfil precondition $d_i \neq d_j$ from Definition 3.4 we would need to abandon rule $d_k \rightarrow d_i$ from L_k, if such a rule existed in L_2.

The IDARM* algorithm exploits the following property of direct association rules: to extract all partial indirect association rules, in which page d_k^{fixed} is transitive, we only need to take all rules $d_i \rightarrow d_k^{fixed}$ and all rules $d_k^{fixed} \rightarrow d_j$. Joining every direct rule from the former set with every rule from the latter set we obtain all partial indirect rules with respect to d_k.

To speed up the working of IDARM* implementation list L_1 can be previously ordered by rule consequents. In such a case, the selection (line 3) would be performed only as many times as the number of unique consequents.

3.6.3. Example

Let us consider the implementation of the IDARM* algorithm to direct rules from Table 3.2. Value *iconmin*=6% was applied so that none of the rules would be excluded. List L_1 was sorted by their consequents for better clearness and to accelerate processing. In consequence, the same auxiliary list L_k was used with many consecutive rules from list L_1. Note that only four non-overlapping lists L_k were needed to finish discovery of all indirect rules. Value max_T=3 comes from l^T_{24}, i.e. $d_2 \rightarrow {}^{\#}d_4$. The final list of complete indirect rules with their confidences is in Table 3.3. Additionally, the final and auxiliary results of the algorithm are shown in Table 3.6.

3.6.4. Complexity of the IDARM* Algorithm

There are two nested loops in the IDARM* algorithm (lines 2–15 and lines 5–13). They both operate on the list of direct rules. Hence, we can estimate the primary complexity of the IDARM* algorithm as $O(m^2)$, where m is the number of processed direct rules. Note that the maximum value of m is $n(n-1)$, where n is the number of web pages.

Obviously, the quantity of direct rules strongly depends on the minimum thresholds applied (Fig. 3.13). Nevertheless, in practice, the value of m, i.e. the number of direct rules that are good enough and exceed reasonable thresholds, is 1–2 orders of magnitude greater than n (Table 3.7). This is simultaneously nearly three orders of magnitude smaller than the maximum number of direct rules, i.e. $n(n-1)$.

Table 3.6. The run of the IDARM* algorithm; input direct rules are from Table 3.2

L_1	L_2	Transitive page d_k	L_k	Complete indirect rules created (line 10, bold) or increased (line 7) (*in order of processing*)	Excluded partial rules (line 4)	No. of complete rules created / increased / total
$d_2 \rightarrow d_1$	$d_1 \rightarrow d_4$	d_1	$d_1 \rightarrow d_4$	$d_2 \rightarrow^{\#} d_4$, $d_3 \rightarrow^{\#} d_4$	$d_4 \rightarrow^{P\#} d_4, d_1$	2 / 0 / 0
$d_3 \rightarrow d_1$	$d_2 \rightarrow d_1$	d_2	$d_2 \rightarrow d_1$, $d_2 \rightarrow d_4$, $d_2 \rightarrow d_5$, $d_2 \rightarrow d_6$	$d_4 \rightarrow^{\#} d_1$, $d_4 \rightarrow^{\#} d_5$, $d_4 \rightarrow^{\#} d_6$, $d_5 \rightarrow^{\#} d_1$, $d_5 \rightarrow^{\#} d_4$, $d_5 \rightarrow^{\#} d_6$, $d_6 \rightarrow^{\#} d_1$, $d_6 \rightarrow^{\#} d_4$, $d_6 \rightarrow^{\#} d_5$	$d_4 \rightarrow^{P\#} d_4, d_2$, $d_5 \rightarrow^{P\#} d_5, d_1$, $d_6 \rightarrow^{P\#} d_6, d_1$	9 / 0 / 9
$d_4 \rightarrow d_1$	$d_2 \rightarrow d_4$					
$d_4 \rightarrow d_2$	$d_2 \rightarrow d_5$					
$d_5 \rightarrow d_2$	$d_2 \rightarrow d_6$					
$d_6 \rightarrow d_2$	$d_3 \rightarrow d_1$	d_3				0 / 0 / 0
$d_1 \rightarrow d_4$	$d_4 \rightarrow d_1$	d_4	$d_4 \rightarrow d_1$, $d_4 \rightarrow d_2$, $d_4 \rightarrow d_5$, $d_4 \rightarrow d_6$	$d_1 \rightarrow^{\#} d_2$, $d_1 \rightarrow^{\#} d_5$, $d_1 \rightarrow^{\#} d_6$, $d_2 \rightarrow^{\#} d_1$, $d_2 \rightarrow^{\#} d_5$, $d_2 \rightarrow^{\#} d_6$, $d_5 \rightarrow^{\#} d_1$, $d_5 \rightarrow^{\#} d_2$, $d_5 \rightarrow^{\#} d_6$, $d_6 \rightarrow^{\#} d_1$, $d_6 \rightarrow^{\#} d_2$, $d_6 \rightarrow^{\#} d_5$	$d_1 \rightarrow^{P\#} d_1, d_4$, $d_2 \rightarrow^{P\#} d_2, d_4$, $d_5 \rightarrow^{P\#} d_5, d_4$, $d_6 \rightarrow^{P\#} d_6, d_4$	8 / 4 / 12
$d_2 \rightarrow d_4$	$d_4 \rightarrow d_2$					
$d_5 \rightarrow d_4$	$d_4 \rightarrow d_5$					
$d_6 \rightarrow d_4$	$d_4 \rightarrow d_6$					
$d_2 \rightarrow d_5$	$d_5 \rightarrow d_2$	d_5	$d_5 \rightarrow d_2$, $d_5 \rightarrow d_4$, $d_5 \rightarrow d_6$	$d_4 \rightarrow^{\#} d_2$, $d_6 \rightarrow^{\#} d_2$, $d_2 \rightarrow^{\#} d_4$, $d_6 \rightarrow^{\#} d_4$, $d_2 \rightarrow^{\#} d_6$, $d_4 \rightarrow^{\#} d_6$	$d_2 \rightarrow^{P\#} d_2, d_5$, $d_4 \rightarrow^{P\#} d_4, d_5$, $d_6 \rightarrow^{P\#} d_6, d_5$,	1 / 5 / 6
$d_4 \rightarrow d_5$	$d_5 \rightarrow d_4$					
$d_6 \rightarrow d_5$	$d_5 \rightarrow d_6$					
$d_2 \rightarrow d_6$	$d_6 \rightarrow d_2$	d_6	$d_6 \rightarrow d_2$, $d_6 \rightarrow d_4$, $d_6 \rightarrow d_5$	$d_2 \rightarrow^{\#} d_4$, $d_2 \rightarrow^{\#} d_5$, $d_4 \rightarrow^{\#} d_2$, $d_4 \rightarrow^{\#} d_5$, $d_5 \rightarrow^{\#} d_2$, $d_5 \rightarrow^{\#} d_4$	$d_2 \rightarrow^{P\#} d_2, d_6$, $d_4 \rightarrow^{P\#} d_4, d_6$, $d_5 \rightarrow^{P\#} d_5, d_6$	0 / 6 / 6
$d_4 \rightarrow d_6$	$d_6 \rightarrow d_4$					
$d_5 \rightarrow d_6$	$d_6 \rightarrow d_5$					
Total:						20 / 15 / 35

3.7. Indirect Rules Influence Direct Ones - Motif Analysis

Direct rules can be treated as the directed edges in the network. Topology of complex networks, both biological and engineered, were analyzed with respect to so-called *network motifs* [Mil02]. They are small (usually 3 to 7 nodes in size) subgraphs, which can occur in the given network far more (or less) often than in the equivalent random networks, in terms of the number of nodes, node degree distribution, average path length, clustering, etc [Jus08a, Jus08b, Mus08c, Mil02].

To study the influence of indirect rules on the complex ones it is reasonable to consider only triads, i.e. subgraphs with three nodes. Overall, there are thirteen possible triad types in the network (Fig. 3.9). Starting with the triads extracted from the network built upon direct rules (triads with the grey background in Fig. 3.9), we can analyze links reflecting both indirect and complex rules. Hence, dotted arrows correspond to new connections derived from indirect rules that enrich the final network based on complex rules.

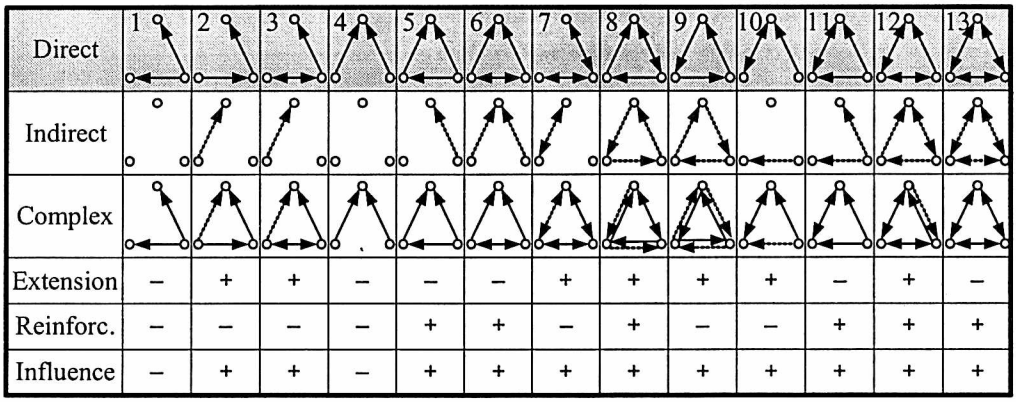

Fig. 3.9. Possible triads that can exist within the network. Upper triad row (grey background) is based on direct rules, middle row - on indirect rules and the lower row - complex rules. Indirect rules can influence (extend and/or reinforce) connections that result from direct rules

Note that indirect rules do not provide any new links in the case of six types of direct triads (1, 4, 5, 6, 11 and 13) whereas the other seven types benefit from indirect rules, i.e. 2, 3, 7, 8, 9, 10 and 12 (see also Table 3.8). Simultaneously, triads number 5, 6, 8, 11, 12, 13 are reinforced by indirect rules. Nevertheless, triad 13 for direct rules coincides with triad for indirect rules and the influence of indirect rules depends only on weights (confidences) assigned to the connections considered. As a result, only two kinds of triads: 1 and 4 do not gain at all from indirect rules, neither in new nor strengthened links.

Thus, indirect rules can provide new knowledge in some cases, while in the other, they can confirm existing connections. The positive contribution of indirect rules depends on the distribution of individual triad kinds. In particular, the more triads of type 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, and 13, the bigger the influence of indirect rules on the recommendation lists based on complex rules.

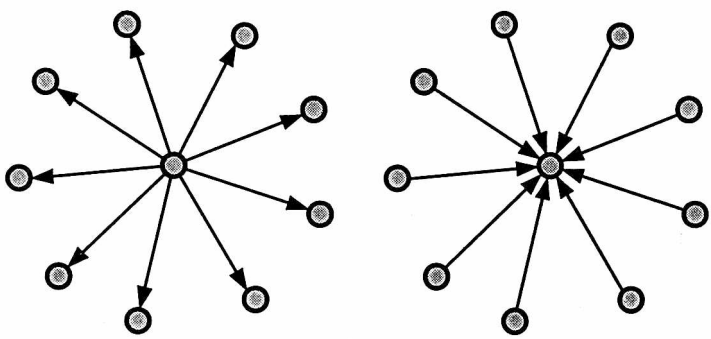

Fig. 3.10. Networks based on direct rules with no corresponding indirect rules

It may theoretically happen that the network built on direct rules consists of only triads of kind 1 or 4, i.e. only incoming or outgoing stars with one node in the centre (Fig. 3.10). In such a case, there would not be any indirect rules. In consequence, they would not influence final complex rules. Nevertheless, such specific, degenerated case is hardly possible in the real environments. In all other cases, indirect rules deliver new knowledge about relationships between web pages.

3.8. Architecture of the Recommender System

The implementation of recommender system based on association rules was realized with distributed architecture. Every system's module may be treated as the software expert-agent that possesses its own characteristic depending on its role in the recommendation process (Fig. 3.11).

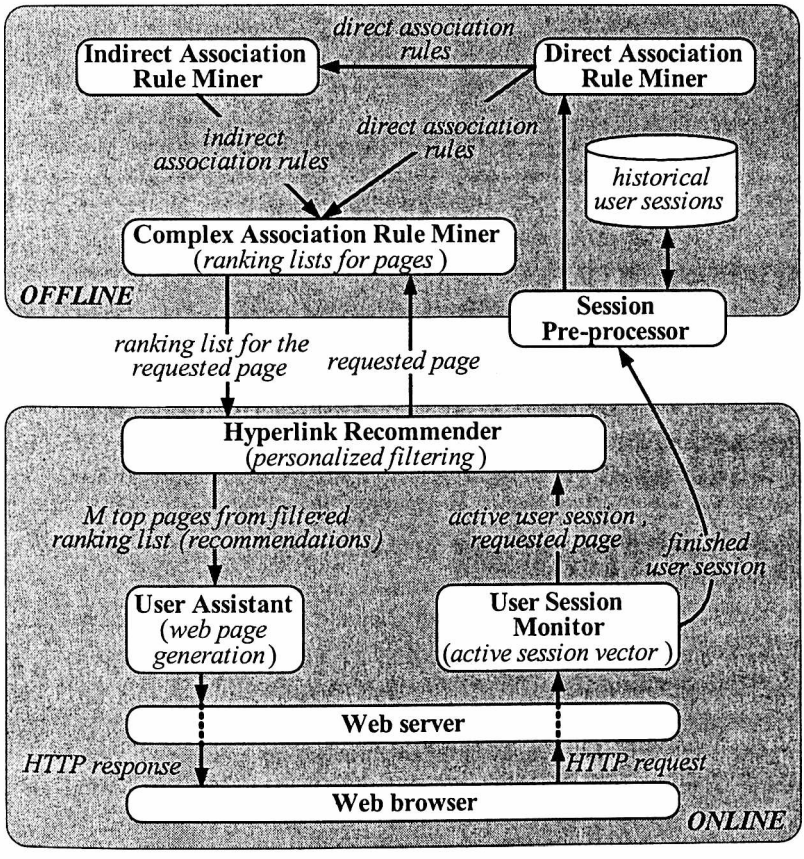

Fig. 3.11. System architecture

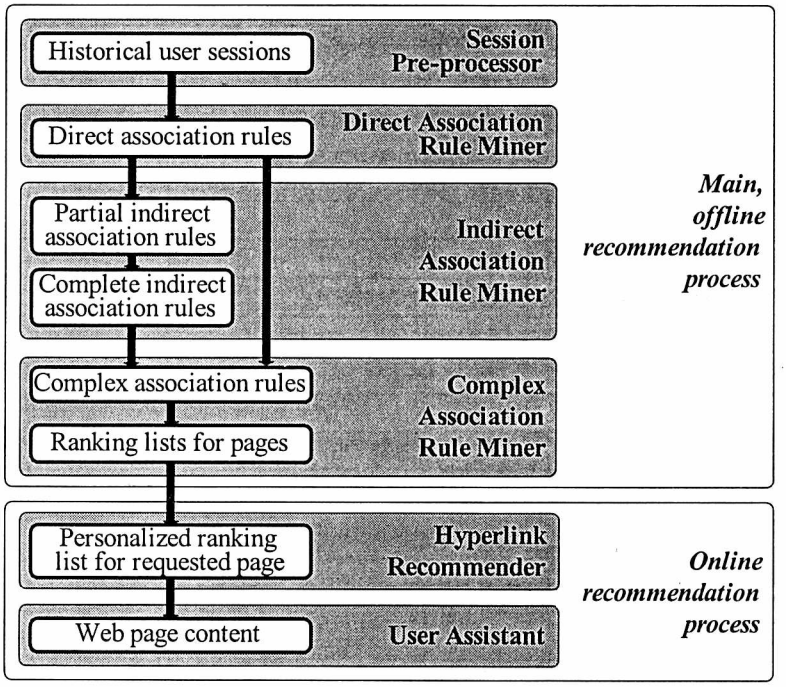

Fig. 3.12. Recommendation process based on the mining of association rules

User Session Monitor captures user HTTP requests and groups them into sessions using JSP servlet session mechanism. It preserves data about active user session and sends it (the set of pages visited during the session) to Session Pre-processor just after the session has finished.

Session Pre-processor filters and gathers in its own database finished sessions obtained from User Session Monitor. It also excludes sessions that are too short, e.g. containing less than two HTTP requests. Storing and filtering is performed online. However, Session Pre-processor makes historical user sessions accessible for offline association rules mining. Thus, this module works both online and offline.

The main recommendation process is performed offline and involves four modules (Fig. 3.12): Session Pre-processor, Direct Association Rule Miner, Indirect Association Rule Miner, and Complex Association Rule Miner. The only task for Session Pre-processor is to deliver historical user sessions to Direct Association Rule Miner.

Direct Association Rule Miner extracts proper direct association rules from user sessions using any of the well known mining algorithms (see Sec. 3.2). The appropriate parameters: *minsup* and *mincon* are used to include merely rules that seem to be useful (see Sec. 3.3).

Indirect Association Rule Miner receives direct association rules from Direct Association Rule Miner and calculates indirect association rules using the IDRAM* algorithm (see Sec. 3.6.2). Similarly to direct association rules, complete indirect association rules are filtered using separate minimum confidence threshold *iconmin* (see Sec. 3.4.2).

Complex Association Rule Miner combines into complex association rules both direct and indirect rules delivered by Direct and Indirect Association Rule Miner, respectively (see Sec. 3.4.5). It creates a separate ranking list for each web page based on the complex rules obtained (see Sec. 3.5). Complex Association Rule Miner operates offline.

Hyperlink Recommender is responsible for creation of the appropriate ranking list for each page requested by the active user. It receives the active user session data and the requested page (URL) from User Session Monitor. The requested page is relayed to Complex Association Rule Miner in order to obtain the static ranking list for this page based on complex association rules. Next, this ranking list is filtered by Hyperlink Recommender to exclude pages lately visited by the user according to active user session data. M top pages from the filtered ranking list are presented to the user (by means of User Assistant).

User Assistant generates the final web page content for the active user. The HTML content includes M hyperlinks (recommendations) provided by Hyperlink Recommender.

Since the web usage patterns tend to change over time, association rules obtained offline should be periodically recalculated. The knowledge update problem was tackled by the introduction of the special update method into the architecture [Kaz03b].

3.9. Experiments

A series of experiments have been conducted in order to discover the influence of direct and indirect association rules on recommendation ranking lists.

3.9.1. Test Environment

The data used for the experiments came from web log files of two big Polish sites, one significant e-commerce that offers hardware, and the other the main portal of the Wrocław University of Technology (WUT), *www.pwr.wroc.pl*. The influence of indirect rules on recommendation lists was partially studied in [Kaz05d].

First the log data were cleansed. All multimedia requests and those generated by search engine spiders, which constituted over 90% of all entries, were removed. Then sessions were identified on the basis of the same user hostname, the

same user agent and the time interval between two consecutive requests within 25.5 minutes [Lu03]. After removing one-page sessions and too long ones (more than 80 pages), which do not reflect actual user behaviour, 173,896 sessions were left for WUT log data, for the period of 9 weeks. For e-commerce this number was 16,085 sessions for the 4-month period. Statistical data for the two sites are presented in Table 3.7.

Table 3.7. Statistical data for two test environments

Item	E-commerce	WUT
Total pages	2,799	10,661
Total cleansed sessions	16,085	173,896
Average session length	7.3	4.7
Total direct rules	547,338	409,318
mincon	1%	1%
minsup	0.02%	0.001%
Filtered direct rules	64,716	124,236
Average *con* for filtered direct rules	19.99%	34.38%
Partial indirect rules	8,292,224	7,563,070
Total complete indirect rules	1,160,786	1,169,477
iconmin	0.01%	0.01%
Filtered complete indirect rules	327,859	631,908
Average *con* for filtered, indirect rules	0.09%	0.34%
α	2%	5%
Complex rules	330,948	637,744
Average *con* for complex rules	0.17%	0.65%
Pages with any rules	1,865	4,733

The parameter α used to reinforce or dump direct rules at the expense of indirect ones was set to a very small value for both sites: 2% for e-commerce and 5% for WUT, since the mean confidence for filtered direct rules was significantly greater than the mean confidence for filtered indirect rules for both sites; 213 and 102 times greater for e-commerce and WUT, respectively.

3.9.2. Thresholds

Values of basic rule thresholds, namely minimum confidence *mincon* and minimum support *minsup* have significant influence on the number of direct rules (Fig. 3.13) and in consequence, also on the number of indirect rules (see also Sec. 3.3). To ensure that indirect rules were formed only from strong direct ones, reasonable values of both *mincon* and *minsup* were applied in further experiments

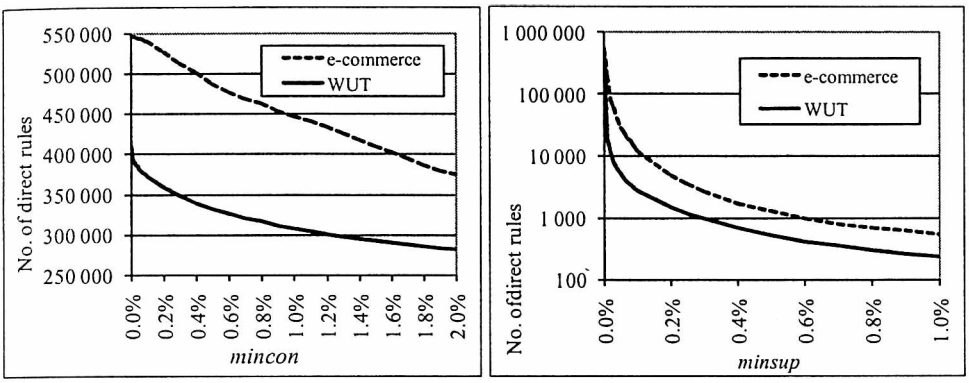

Fig. 3.13. The number of direct rules in relation to *mincon* (*minsup*=0) and *minsup* (*mincon*=0)

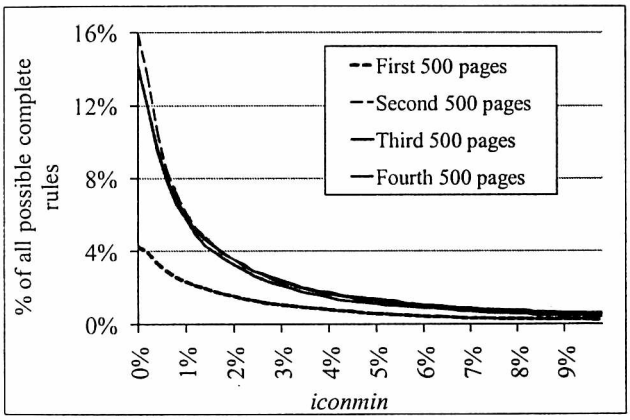

Fig. 3.14. The number of complete indirect rules as the percentage of all possible rules (249,500) in relation to *iconmin* for consecutive 500-page sets from e-commerce

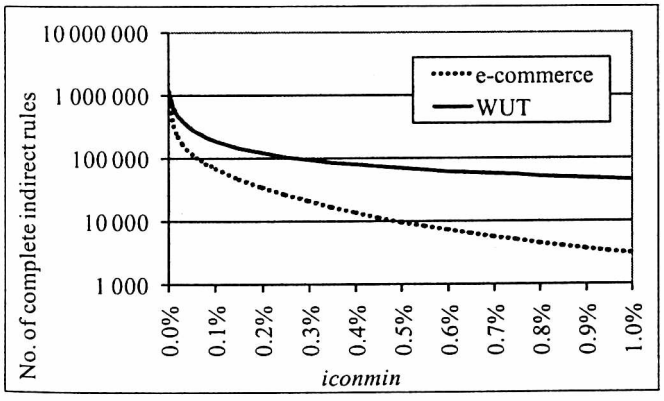

Fig. 3.15. Number of complete indirect rules in relation to *iconmin*

(see Sec. 3.9.4 to 3.9.9). Hence, *mincon*=1% was the same for both sites, whereas *minsup* had to be smaller for the WUT site (*minsup*=0.001%) due to the fact that the number of pages on this site was considerably bigger and at the same time the average session length was smaller than on the e-commerce site (*minsup*=0.02%), Fig. 3.13, Table 3.7.

Similarly, *iconmin* was introduced for complete indirect rules (Fig. 3.14 and 3.15). Its value was set to the square of *mincon* for both of the sites (Table 3.7).

3.9.3. Kendall's and Spearman's Rank Correlation Coefficients

Ranking lists containing suggested pages d_j were created separately for each web page d_i based on confidence values of appropriate association rules, i.e. either direct $con(d_i \rightarrow d_j)$ or indirect confidence $con^{\#}(d_i \rightarrow^{\#} d_j)$ or complex one $con^{*}(d_i \rightarrow^{*} d_j)$, see Sec. 3.5.

However, we would need a method to compare rankings somehow. For this purpose, Kendall's coefficient of concordance as well as Spearman's rank correlation coefficient were used to determine the similarity between two ranking lists.

Let A and B be any n-item rankings, e.g. two lists of n most similar pages d_j for the given page d_i created using different approaches. Note that each page d_j can possess in both rankings A and B different positions: a_i and b_i, respectively. For example, page d_5 can occupy the second place in ranking A (position a_5=2) and the seventh place in ranking B (b_5=7).

Kendall's coefficient of concordance $\tau(A,B)$ can be evaluated from the following formula [Dan01, Fag03, Kaz05d, Ken48]:

$$\tau(A,B) = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} \operatorname{sgn}(a_j - a_i) \operatorname{sgn}(b_j - b_i), \tag{3.8}$$

where a_i and b_i are the positions of the same ith item in rankings A and B, respectively; they range from 1 to n; $\operatorname{sgn}(a_j - a_i)$ is the sign of the difference $a_j - a_i$. It means that if item j follows item i in ranking A, then $\operatorname{sgn}(a_j - a_i)$=–1; if they are at the same position $\operatorname{sgn}(a_j - a_i)$=0; otherwise $\operatorname{sgn}(a_j - a_i)$=+1.

When two rankings have the same items at every position, Kendall's coefficient for them is equal to +1. However, when two rankings have all items the same but they occur in reverse order, their Kendall's coefficient equals –1.

For the same n-item rankings A and B Spearman's coefficient $\sigma(A,B)$ is expressed in the following way [Dan01, Fag03, Spe87]:

$$\sigma(A,B) = 1 - \frac{6}{n(n^2-1)} \sum_{i=1}^{n} (a_i - b_i)^2 \tag{3.9}$$

Similarly to Kendall's coefficient, Spearman's coefficient amounts to +1 for the same rankings and -1 for rankings in reverse order. Generally, Spearman's coefficient can be treated as a special case of the Pearson product-moment coefficient in which the data are converted to rankings before calculation.

As neither Eq. (3.8) nor (3.9) can be used for 1-item rankings, it was assumed that when the only item in both rankings was the same, Kendall's and Spearman's coefficients were assigned the value of 1, otherwise the value of -1 was established.

The formulas (3.8) and (3.9) work fine for rankings with the same number of items. However, as far as recommendation ranking lists derived from association rules are concerned, it is rarely the case. The length of ranking lists ranges from 1 to several hundred sometimes. Therefore a method for handling different length rankings had to be devised. We suggest appending all items from ranking B, which do not occur in list A, after the last item in ranking A. All appended items attain the same position: the origin length of A plus 1. As a result while comparing two n-item rankings A and B we may obtain up to $2*n$-element rankings after conversion. Afterwards, it is only those extended rankings that Eq. (3.8) or (3.9) is applied to.

3.9.4. Correlation of Recommendation Ranking Lists

Having direct, indirect and complex association rules, the recommendation ranking lists by means of Kendall's and Spearman's coefficient were compared for e-commerce and the WUT site. In particular, similarities between direct and complex ranking lists were examined. The lists were cut at various lengths: 1, 2, 3, 4 as well as 5, and Kendall's and Spearman's coefficients were calculated for such rankings.

The results of the experiment show that the median for 1-item rankings was 1 for both sites (and for 2-item rankings for e-commerce), which means that most of the pages recommended in the first position were the same for direct and complex rankings. This refers more frequently to e-commerce, as the average is greater and the standard deviation is smaller. Nevertheless, 12.8% of rankings for e-commerce and 46.6% for WUT had the first item that was different. The results for 1-item rankings do not follow a pattern set by a bit longer lists. For lists with up to 5 items, the mean and the average appear to increase gradually and the standard deviation appears to fall as the length of the list rises (Fig. 3.14–3.19). On average, both Kendall's and Spearman's coefficients for all ranking lengths examined were higher for the e-commerce site. This may have resulted from the value of parameter α that reinforces more direct rules, making direct and complex recommendation ranking lists more similar. In general, ranking lists based on direct and complex association rules are rather correlated and do not essentially differ from each other. Nevertheless, they are not the same and complex rules order items in rankings in a slightly dissimilar way.

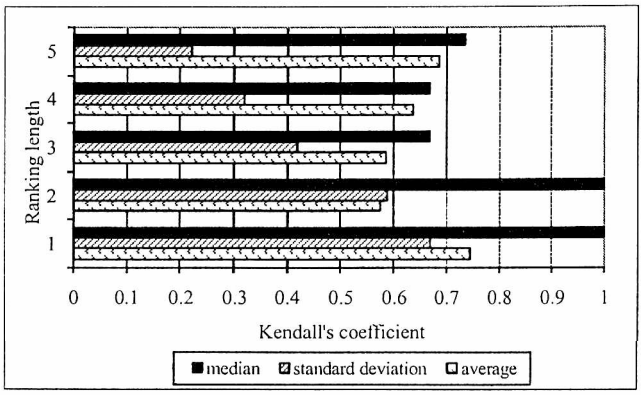

Fig. 3.16. Kendall's coefficient for direct and complex ranking lists for e-commerce site

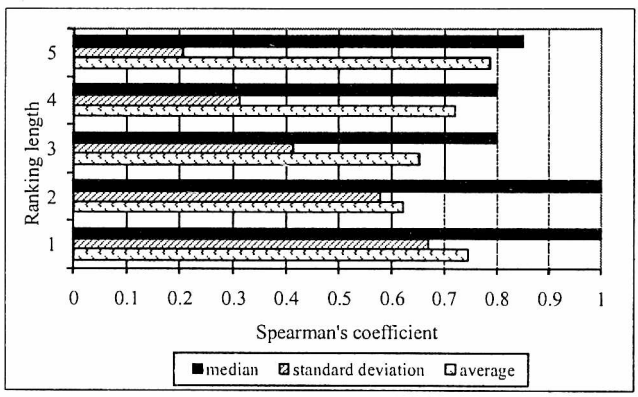

Fig. 3.17. Spearman's coefficient for direct and complex ranking lists for e-commerce site

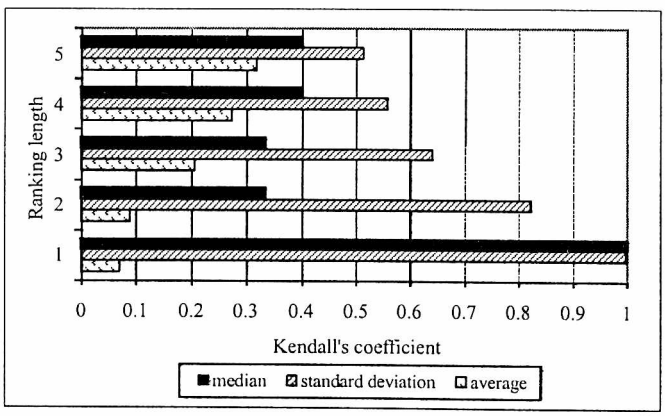

Fig. 3.18. Kendall's coefficient for direct and complex ranking lists for WUT

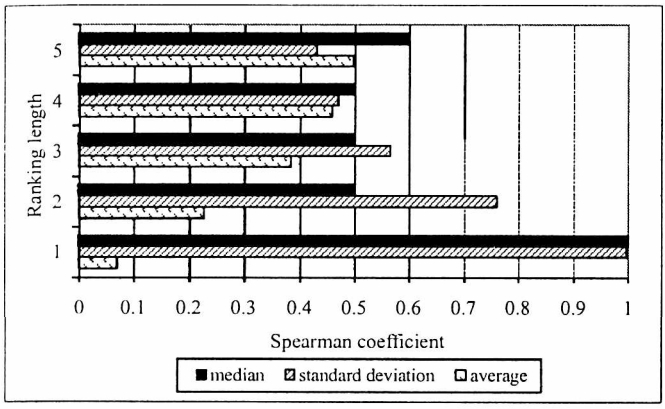

Fig. 3.19. Spearman's coefficient for direct and complex ranking lists for WUT

Note that Kendall's and Spearman's coefficients deliver similar knowledge about rank correlation, even though the average values of Kendall's coefficient are noticeably smaller and standard deviations are greater for WUT compared to Spearman's coefficient. A similar conclusion regarding the rank correlation coefficients was presented in [Fag03].

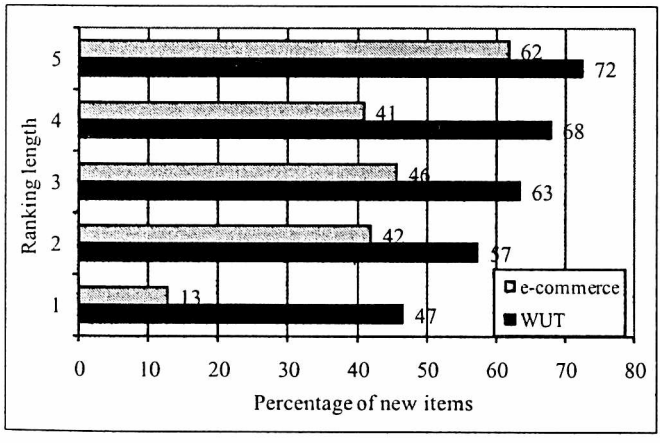

Fig. 3.20. The percentage of pages for which new items were added to rankings based on complex rules compared to the direct ones

The greatest changes between direct and complex ranking lists can be observed in the first and second position. Thus, we gain most from introducing indirect and complex rules in very first positions, as they offer completely new knowledge, which direct rankings did not possess. Moreover, even for very short rankings the percentage of pages for which new items were added in rankings

based on complex rules compared to the direct ones was quite high, which indicates that direct rankings were certainly enriched with new suggestions (Fig. 3.20). Concluding, the rankings based on complex rules are always able to provide some new knowledge to the recommender system.

3.9.5. Complex Rules Extend Direct Ranking Lists

The main reason for using indirect association rules is the fact that they provide substantially more suggestions for recommendations compared with direct rules. The experiments performed revealed that there was a great number of pages with very few recommendations (5 items or less) - 25.5% of all 1,865 pages with any ranking for e-commerce and 27.7% from all 4,733 pages for WUT (Fig. 3.21 and 3.23). This is the case where indirect association rules may become very useful, since they can considerably lengthen short direct ranking lists. The percentage of rankings with long lists in indirect rankings nearly agrees with that of complex rankings and is much greater than in direct ones (Fig. 3.21).

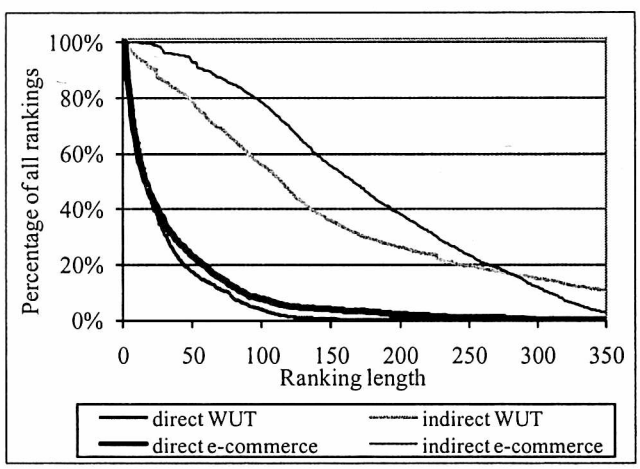

Fig. 3.21. Contribution of rankings within all rankings that accomplish at least the given length separately for direct and indirect ranking types; complex rankings visually agree with indirect ones

The average length of direct rankings was 34.7 for e-commerce and 26.2 for WUT. These values for indirect rankings were 175.8 and 133.5, respectively, while for complex rankings 177.5 and 134.7, respectively (Fig. 3.22). The increase in length of indirect rankings compared to complex ones was only 1%, whereas complex rankings were on average 5.11 times longer than direct ones for e-commerce and 5.13 times longer for WUT (Fig. 3.22).

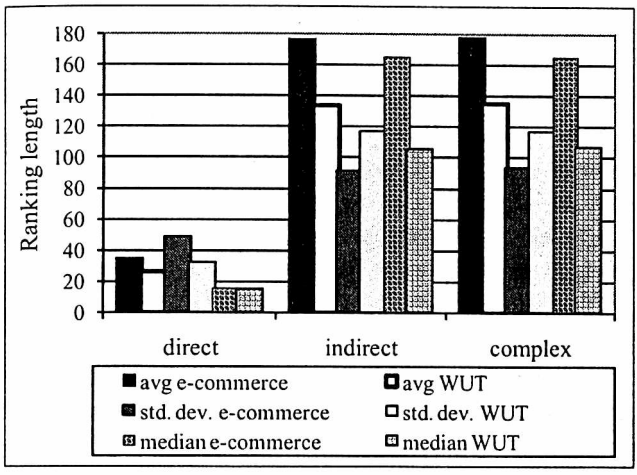

Fig. 3.22. Average ranking lengths for different ranking types

The contribution of pages with too short direct ranking lists to all pages was examined and results are presented in Fig. 3.23. For the smallest length required, i.e. 2, the percentage of too short rankings was similar for e-commerce and WUT: 4.2% and 11%, respectively.

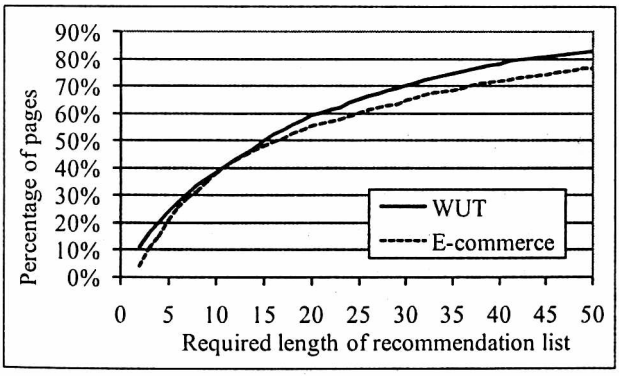

Fig. 3.23. Contribution of pages with too short ranking lists based on direct rules within pages with any ranking

In general, the average percentage of pages with too short rankings was quite prominent. The proposed solution to this problem is the extension of direct ranking lists by means of indirect and complex ones. Thus, there has been tested the contribution of pages with too short rankings which were successfully extended with complex rules within all pages with short rankings. To this end, the number of pages has been taken with too short direct rankings for which complex

ranking lists were longer than direct ones, separately for each list length. Next, this number was divided by the total number of pages with too short direct rankings. The obtained results were very similar for indirect and complex rules. They show how much short direct ranking lists can be extended with indirect or complex ones (Fig. 3.24). For complex rankings the percentage started from 97.5% for e-commerce and 70.1% for WUT (for 2-item rankings) and reached 99.89% for e-commerce and 95.1% for WUT.

This definitely emphasizes the usefulness of complex and in consequence indirect association rules.

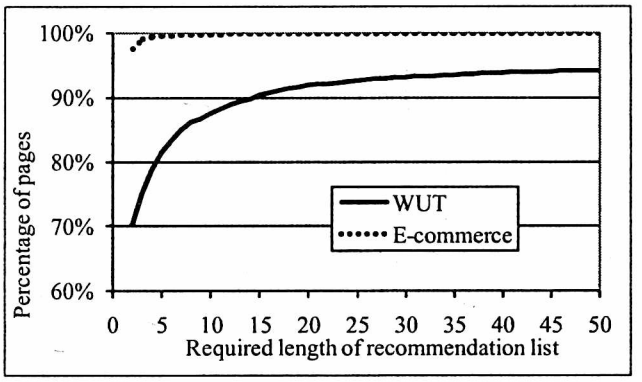

Fig. 3.24. Contribution of pages with too short direct ranking lists extended with complex rules within all pages with too short rankings

3.9.6. Coverage of User Sessions by Recommendation Lists

The average coverage of user sessions by recommendation lists was examined in the next experiments carried out for the WUT web site. The recommendation list for each page in the user session - as the unordered set - was compared to the content of the session, i.e. visited pages. The greater the part of the session covered by the recommendation set, the better. This reflects the ability of the recommender system to suggest suitable next steps which is confirmed by the pages that the user really visited. In the case of direct rankings, the percentage coverage was significantly lower than for indirect and complex rankings for all sessions longer than 2 (Fig. 3.25). The difference amounted from 3.1% for 3-item rankings, up to 15.3% for 28-item rankings. In general, the coverage decreased for longer sessions since they required more items to be matched. This undoubtedly results from the substantially longer indirect and complex rankings compared to direct ones for every length of sessions (Fig. 3.26). The greater the dissimilarity in ranking length is, e.g. for session of length 21–50, the greater also the difference in session coverage is (13.6%–15.3%).

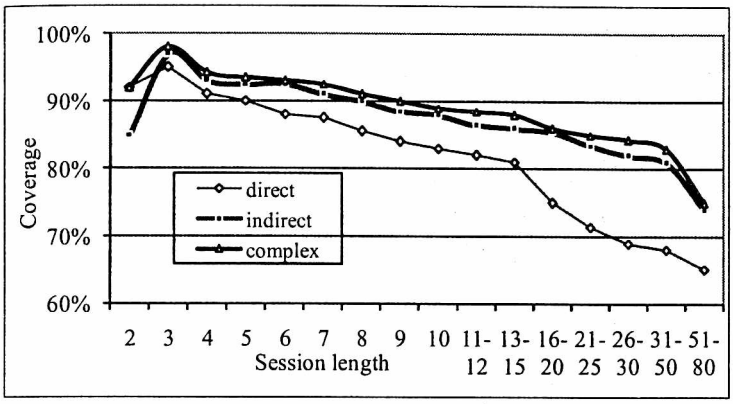

Fig. 3.25. Coverage of user sessions by ranking lists in relation to the session length

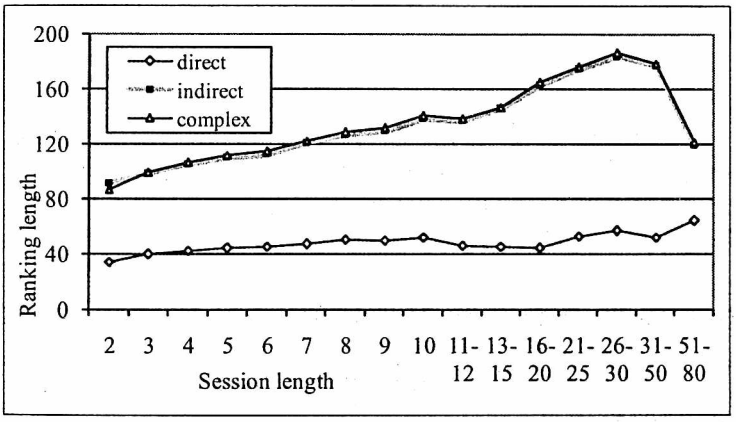

Fig. 3.26. Average ranking lengths for different lengths of sessions

3.9.7. Recommendation Ranking vs. Existing Hyperlinks

A vital question one can ask is whether recommendation ranking lists only confirm existing hyperlinks or maybe they add new knowledge as well. If all they did was to confirm the existing structure of hyperlinks on a site, the recommendations offered to a user might not be interesting to them.

In order to test if ranking lists add anything new, the content of the WUT site was downloaded. From it information about all hyperlinks on each page was extracted. Having the structure of the site, the assessment of recommendation lists became possible in the following way: the number of common items in hyperlink sets and ranking lists cut at various lengths was divided by the ranking length, i.e. the required length or the actual length if it was smaller than the

length required. Such calculations were performed for direct, indirect and complex recommendation lists, Fig. 3.27.

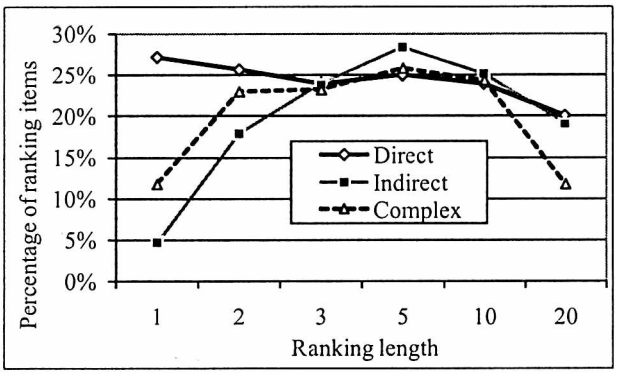

Fig. 3.27. The average percentage of ranking items covered by hyperlinks at the WUT site

The results for direct, indirect and complex rankings for very short lists (1 and 2 items) differ substantially: 27.1%, 4.6% and 11.7%, respectively for 1 item. The same is true for 20-item long lists: 19.9%, 19% and 11.7%, respectively. This indicates that for very short indirect and complex rankings (up to 2 pages) and long ones (20 items and more) recommendation lists go beyond confirming existing hyperlinks and add new potentially useful knowledge.

3.9.8. Usage of Association Rules for Hyperlink Assessment

Another application of direct, indirect and complex rules may be the assessment of hyperlinks on a site. Hence, we can test whether the hyperlinks on a page have been placed appropriately by analyzing significant navigational patterns (association rules) derived from user behaviour.

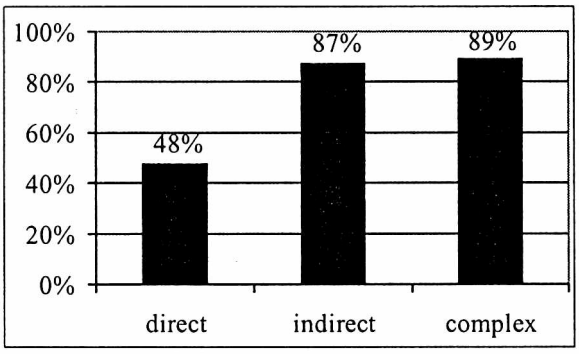

Fig. 3.28. The average percentage of all hyperlinks confirmed by rules at the WUT site

In the experiments carried out for the WUT web site, the percentage of hyperlinks confirmed by rules was calculated by dividing the number of common items in hyperlink sets and whole ranking lists for a given page, by the number of hyperlinks on the page, separately for direct, indirect and complex rules (Fig. 3.27). Note that the number of hyperlinks was put in the denominator as opposed to calculations in Sec. 3.9.7 where it was ranking length.

The average percentage of hyperlinks confirmed by direct rules amounted to only 48%, probably because there were too few of them. Indirect and complex rules, on the other hand, confirmed many more hyperlinks - 87% and 89%, respectively, due to their larger quantity. These relatively great values that have been obtained may have resulted from the enormous differences between the average number of hyperlinks on a page - 10 and the average ranking length: 51, 177, and 180 for direct, indirect and complex rules, respectively. Concluding, indirect and complex rules appear to be better for assessing the usefulness of hyperlinks compared to direct rules.

Note that in any case at least 11% of hyperlinks were not confirmed by any rule, so they may be recommended to be removed from the content of pages. The usage of another kind of patterns - negative association rules for the same purpose was presented in [Kaz08f].

3.9.9. Motif Distribution

There is a discussion related to small subgraphs called motifs in Sec. 3.7. Most motifs created upon direct rules result in indirect rules that can influence the source direct rules either by new connections or by reinforcing the existing ones. Only motifs of type 1 or 4 provide no indirect rules.

Table 3.8. Distribution of motifs in the network built from direct rules. Motif IDs correspond to indexes in Fig. 3.9

Motif ID	Extension	Reinforcement	E-commerce	WUT
Motif 1	-	-	0%	2.9%
Motif 2	+	-	0.003%	3.0%
Motif 3	+	-	1.0%	6.4%
Motif 4	-	-	55.9%	62.9%
Motif 5	-	+	0%	1.2%
Motif 6	-	+	1.0%	1.9%
Motif 7	+	-	15.8%	4.4%
Motif 8	+	+	0%	0.1%
Motif 9	+	-	0%	0%

(Continued on the next page)

Table 3.8.

Motif ID	Extension	Reinforcement	E-commerce	WUT
Motif 10	+	-	20.8%	11.2%
Motif 11	-	+	0.8%	1.9%
Motif 12	+	+	1.7%	1.3%
Motif 13	-	+	2.9%	2.8%
Summary				
Total motifs			12,551,518	10,163,553
Extension	+		39.4%	26.4%
No extension	-		60.6%	73.6%
Reinforcement		+	6.4%	9.2%
No reinforcement		-	93.6%	90.8%
Influence	+		44.1%	34.2%
No influence	-		55.9%	65.8%

Table 3.8 contains distribution of motifs in the network based on direct rules with the thresholds from Table 3.7 applied. Over one third of motifs in the case of e-commerce and over one fourth in the case of WUT facilitate new connections while less than 10% of motifs provide reinforcement. In total, in almost a half (e-commerce) and $^1/_3$ (WUT) of motifs direct rules are influenced by indirect ones.

3.10. Conclusions

Indirect association rules reflect relationships existing both between and within web user sessions. Complex rules combining both direct and indirect rules usually increase the length of rankings compared to those based on direct associations. This helps overcome the problem of a multitude of pages with too short rankings (Fig. 3.23) and makes it possible for them to fulfil the requested ranking length (Fig. 3.24). Additionally, indirect rules substantially change the order of ranking lists (Fig. 3.16–3.19). Moreover, they provide new knowledge to the rankings since they introduce new items not available for direct association rules (Fig. 3.20).

Recommendation lists based on direct rules to a greater extent only confirm hyperlinks existing on web pages compared to lists extracted from complex rules, for short and long ranking lengths (Fig. 3.27). Besides, all kinds of rules, especially indirect and complex ones, can be useful for the assessment of hyperlinks.

Indirect rules may not only confirm and strengthen direct relationships but they also often link objects not related with direct rules. In the web environment, they can help to go outside of typical user navigational paths that result from static hyperlinks, so they reveal many associations out of reach for direct rule

mining. For all these reasons, indirect rules are useful in recommender systems: they extend ranking lists and add to them non-trivial information.

Owing to the IDARM* algorithm presented, we obtain complete indirect rules with their complete indirect confidence. The algorithm exploits pre-calculated direct rules rather than raw user session data.

The recommender engine based on association rules is built in the distributed architecture that facilitates system expansion and redistribution between hosts.

4 Positive and Negative Association Rules in the Assessment of Web Hyperlink Usability

Hyperlinks incorporated into web pages determine user navigational paths and are one of the crucial factors on which portal usability depends. Since the quality of the portal content, layout and structure is an important element of its competitiveness, the site designers and content managers exploit their knowledge, experience and even automatic support tools to identify and remove the least valuable hyperlinks and replace them with more suitable ones. Nevertheless, users often have their own habits, needs and abilities, so they take advantage of some hyperlinks while the others are left unused.

The main goal of this section is to present the HRS method (the Hyperlink Recommender System) for both positive and negative hyperlink assessment, which can be very useful in continuous adaptation of web site structure to user preferences and behaviour. The method evaluates usability of the existing hyperlinks and suggests the new ones based on web usage mining techniques, i.e. analysis of web server logs.

For this purpose, positive and negative association rules extracted from web user sessions are used. They indicate either positive or negative relationships between web pages and can be treated as patterns of general user behaviour. Positive association rules both confirm existing hyperlinks and express the user needs for new ones, whereas negative rules point to the low usefulness of hyperlinks placed on web pages. Positively and negatively verified as well as new hyperlinks are presented to the web site content manager and can considerably facilitate the maintenance of the web site structure and its adjustment to user behaviour. Hence, the interactive appraisal of hyperlinks can be seen as a kind of specific recommender system whose users are web administrators rather than regular web site visitors.

The experiments carried out, in particular with expert contribution, on four web sites confirmed the usefulness of the HRS method, see Sec. 4.5.

This section has been prepared based on [Kaz07a].

4.1. Background

The main aim of most web recommender systems is to suggest the next steps in the navigational path to web users. The most typical data mining methods used for this purpose are association rules [Chu04, Ger03, Gey02, Mob01, Wang02], clustering [Kaz07a, Mob00b], sequential patterns [Ger03, Kaz07c], and their varied combinations [Law01, Mob02, Wang04]. Suggestions can be created based upon patterns derived from web logs, i.e. user behaviour - web usage mining [Ger03, Pie03, Wang04], content of web pages - web content mining [Dum00], hyperlinks - web structure mining [Chak99, Lau00] as well as diverse hybrid combinations [Kaz07a].

Baraglia and Silvestri utilized their own measure of web page usability. It is based on the analysis of web logs and is independent of the content of pages. The strength of the correlation between two pages is symmetric and its idea is similar to the confidence function in association rules; see Eq. (4.2). The main difference is that the authors used in the denominator the greater value of the two: the number of user sessions that contain the first web page and the number of sessions containing the second page [Bar07].

Another relevant domain, besides recommendations for end users, is web assessment. This assessment may include many different factors. Almost every part of a web portal design may be assessed. There are many different approaches to improve site structure usability and the basic one is link validation. Currently, every web design application has the ability to verify the correctness of hyperlink destination. However, the problem of missing links still occurs in the case of very big sites. An innovative mobile agent solution, which can be used even with very limited access to the Internet connection, has been presented in [Chan01].

The other aspect of navigation structure improvement is proposals of new hyperlinks. One of such methods exploits case based reasoning (CBR) as a possibility for the automatic generation of hyperlinks for hypertexts in order to extend traditional textual methods [Haf00].

Another method of hyperlink assessment is querying visitors using forms [McG01]. However, it is very difficult to evaluate the replies since users tend to present subjective opinions.

The next method is a statistical log analysis. It may deliver information about most common paths, average session length, pages where visitors leave the site, etc. An example of improvements based on this approach has been presented in [Sul97].

There are also some mathematical methods to assess the navigation structure of an information system, e.g. the analysis of the graph complexity [Wey88]. The most reliable graph measure appears to be the number of independent paths in the graph.

Srikant and Yang proposed algorithms to discover incorrect locations of web pages in the hierarchical structure of the site based on the backtrack analysis of navigational paths extracted from web logs [Sri01].

In many cases an automated assessment is the best way to discover incorrect site structure. Based on the archive of navigational patterns, some automatic simplifications of paths existing in the system can be recommended [Rod02, Sri01]. Typical association rules and their indirect version were used for the creation of recommendation ranking lists and as less important research for the assessment of hyperlinks. It has been experimentally proved that almost half of all hyperlinks can be confirmed by typical association rules while almost 90% by indirect ones [Kaz05d].

Spiliopoulou and Pohle have defined the success of a site as the efficiency of its component pages in attracting users to exploit the supported services and buy the goods offered, especially in e-commerce sites. For this purpose, they proposed three basic measures: the contact efficiency, relative contact efficiency and conversion efficiency of a page. All of these are evaluated with statistical analysis of data about page requests, and both customer and non-customer user sessions extracted from web logs. Finally, they suggest pages that needed to be improved [Spi01].

Cowderoy analyzed web site complexity from the developer perspective and compared it to the typical metrics useful for software development projects [Cow00].

Effectiveness of the web site can also be studied from the organizational point of view as a measure of service quality provided especially by the state dependent agencies [Wel07].

A short overview of algorithms for mining negative association rules can be found in Sec. 2.1.3.

4.2. Association Rules in the Web

Definition 4.1. Let $P=\{p_1, p_2, ..., p_K\}$ be a set of web pages in a single web site. Let S, called a session, be a tuple $<S^+, S^->$. Each session S consists of a set of pages $S^+ \subset P$ visited during one user visit and all other pages that have not been visited $S^- \subset P$, such as $S^+ \cup S^- = P$ and $S^+ \cap S^- = \emptyset$. Let D be a multiset of all sessions available for analysis.

In other words, a session is in a sense the partition of set P.

The multiset D can possess repetitions, but the order of its members (sessions) does not matter. Thus, there may exist two or more sessions in D with the same component elements S^+ and S^-, i.e. the same pages visited. Note that a user can

request and watch the given page p_i many times during one visit but the page p_i will occur in the session S only once.

Note that a session is here slightly different than in Sec. 3.3, Definition 3.2. According to Definition 4.1, a session also contains unseen web pages (set S^-). This has been introduced to facilitate the description of negative patterns, in particular confined negative association rules, see Sec. 4.2.2.

4.2.1. Positive Association Rules

Definition 4.2. A positive association rule is an expression of the form $X \rightarrow Y$, where $X \subset P$, $Y \subset P$, $X \cap Y = \emptyset$, for which there are $N>0$ user sessions $S_i = \langle S_i^+, S_i^- \rangle$, $i=1,2,\ldots N$; $S_i^+ \subset D$, $S_i^- \subset D$ such that $X \cup Y \subset S_i^+$.

A positive association rule $X \rightarrow Y$ indicates whether set X of web pages co-occurs in user sessions with another set Y.

Positive association rules can be extracted directly from the session multiset D using any of the specialized association rule mining algorithms, see Sec. 2.1.1.

Each rule has two associated measures that denote its significance and strength, called support and confidence, respectively. The support $sup(X \rightarrow Y)$ of the positive rule $X \rightarrow Y$ in the multiset D specifies the popularity of the rule and is described with the following formula:

$$sup(X \rightarrow Y) = \frac{card(\{S = \langle S^+, S^- \rangle \in D : X \cup Y \subset S^+\})}{card(D)}. \tag{4.1}$$

The confidence $con(X \rightarrow Y)$ of a positive rule $X \rightarrow Y$ in the multiset D is:

$$con(X \rightarrow Y) = \frac{card(\{S = \langle S^+, S^- \rangle \in D : X \cup Y \subset S^+\})}{card(\{S = \langle S^+, S^- \rangle \in D : X \subset S^+\})}. \tag{4.2}$$

4.2.2. Confined Negative Association Rules

Negative association rules are another type of associations that reflect negative relationships between objects.

Definition 4.3. A confined negative association rule $X \rightarrow \sim Y$, where $X \subset P$, $Y \subset P$, $X \cap Y = \emptyset$, is the association, for which there are $N>0$ user sessions $S_i = \langle S_i^+, S_i^- \rangle$, $i=1,2,\ldots N$; $S_i^+ \subset D$, $S_i^- \subset D$, such that $X \subset S_i^+$ and $Y \subset S_i^-$.

Overall, a confined negative association indicates the negative relationship between X and Y, i.e. if set X occurs in some user sessions, another separate set Y does not co-occur or co-occurs very rarely in these sessions.

In further sections, confined negative association rules are also called shortly negative rules or negative association rules.

The support $sup(X \rightarrow \sim Y)$ of a confined negative rule $X \rightarrow \sim Y$ in the multiset D is:

$$sup(X \rightarrow \sim Y) = \frac{card(\{S = \langle S^+, S^- \rangle \in D : X \subset S^+ \wedge Y \subset S^-\})}{card(D)}. \tag{4.3}$$

The confidence $con(X \rightarrow \sim Y)$ of a confined negative rule $X \rightarrow \sim Y$ in the multiset D is evaluated with the following equation:

$$con(X \rightarrow \sim Y) = \frac{card(\{S = \langle S^+, S^- \rangle \in D : X \subset S^+ \wedge Y \subset S^-\})}{card(\{S = \langle S^+, S^- \rangle \in D : X \subset S^+\})}. \tag{4.4}$$

In practice, only positive and negative rules with support that reaches *minsup* threshold, and with confidence of at least *minconpos* for positive associations and *minconneg* for the negative ones, are really considered. In other words, $sup(X \rightarrow Y)$, $sup(X \rightarrow \sim Y) \in [minsup;1]$, $con(X \rightarrow Y) \in [minconpos;1]$, and $con(X \rightarrow \sim Y) \in [minconneg;1]$. The separation of confidence thresholds into *minconpos* and *minconneg* for positive and negative associations respectively, results from the usually different typical values of both kinds of rules. Negative associations have mostly greater confidence than the positive ones, see Sec. 4.5.4.

A rule $X \rightarrow Y$ or $X \rightarrow \sim Y$ where $card(X)=card(Y)=1$ is called a simple rule; otherwise it is the complex one.

Similarly, we can define two other types of confined negative rules: $\sim X \rightarrow Y$ and $\sim X \rightarrow \sim Y$. However, their interpretation for hyperlink recommendation is questionable. Symbol $\sim X$ denotes that the rule concerns the elements of X not being visited during user sessions. If page p was not visited frequently enough, we should not assess usability of its content including its outgoing hyperlinks. Rules of type $\sim X \rightarrow Y$ solely indicate that pages (elements) of Y were presented to users but with no navigation through the elements from X. There is only one reasonable conclusion that can be drawn from $\sim X \rightarrow Y$ and $\sim X \rightarrow \sim Y$: the legitimacy of the existence of the entire page $p \in X$ in the web site is problematic or the page p is difficult to reach. However, it is hard to address such knowledge to a particular component of p, including its outgoing hyperlinks. For all these reasons, rules of the type $\sim X \rightarrow Y$ and $\sim X \rightarrow \sim Y$ are omitted in the PANAMA algorithm and the HRS method described in Sec. 4.3 and 4.4.

4.3. Mining Positive and Negative Association Rules from Web Logs for Hyperlink Assessment

To extract association rules from a web log, its records have to be adequately prepared and afterwards the appropriate algorithm can be applied.

4.3.1. Data Preparation

The most demanding step of hyperlink recommendation based on web usage mining is data preparation, and it regards both server logs as well as web content.

Log data is cleaned up by removing all requests that have finished in an error code. Afterwards, non-web page requests such as JavaScript, styles, pictures, etc. are excluded. The final step of log processing is filtering by agent field using the positive list of browsers – in consequence all external crawler requests are removed. The next phase is the joining of requests into sessions. A session is a set of pages that have been downloaded by one user during a single visit to the site. It is identified by the same IP address and agent field provided that the time gap between two following requests is no longer than 30 min [Coo99]. Additionally, a session has to consist of at least 2, and no more than 200, pages. The upper restriction is very useful in filtering sessions that have been provided by crawlers that identify themselves as regular web browsers. Note that the knowledge about the order of visiting pages is lost. Since a session is defined as a regular set of pages, it is not possible to keep information about multiple requests for the same page in a single session as well.

The final step of preparing logs is matching them with corresponding HTML pages. It is based on the comparison of the official page address with URLs requested by users and extracted from web logs. For this purpose, the entire site content has to be either downloaded using a separate web crawler or retrieved directly from the web server database. Note that the site content is downloaded only once, so it is a snapshot of a certain point in time. It may cause many problems upon matching this content with logs gathered from the longer period. Completeness of this operation depends on changeability of site structure. In fact, it is impossible to match all log requests with pages, especially in the case of very dynamic portals for which new pages are added and some out of date ones are removed very frequently. The more changes in structure within the log collection period, the lower the accuracy of matching.

4.3.2. Shortcomings of Existing Algorithms

Negative patterns, e.g. negative association rules, can be used to verify positive associations delivered by other methods or already existing in the system. How-

ever, none of the known algorithms used to mine negative association rules, see Sec. 2.1.2, is able to exclude from verification, i.e. checking whether the candidates for frequent itemsets really exceed given thresholds, those candidates that do not correspond to the positive knowledge gathered previously. Such exclusion would obviously reduce processing. Besides, the reasonable thresholds for positive rules usually differ from the rational thresholds for negative ones; the latter are much higher. This difference was not respected by the algorithms mentioned above. These disadvantages of known algorithms were overcome in the PANAMA algorithm proposed in Sec. 4.3.3.

4.3.3. The PANAMA Algorithm

Having extracted user sessions, see Sec. 4.3.1, both positive and confined negative association rules can be mined. There are some algorithms delivering both positive and negative association rules, see Sec. 2.1.3. Overall, any of them are suitable, provided that the resulting set is, or can be limited to rules of the form $X \rightarrow Y$ or $X \rightarrow \sim Y$, see Sec. 4.2.2. However, to defeat some disadvantages of existing solutions, see Sec. 4.3.2, the new PANAMA algorithm is proposed.

The HRS method described in Sec. 4.4 is generally inspired by the algorithm presented in [Ant04b]. However, three significant improvements have also been incorporated to adjust the algorithm for hyperlink classification. Firstly, the mechanism for matching the rule candidates with the set of hyperlinks has been introduced. Secondly, there are two separate thresholds for confined negative and positive rules. Finally, useless confined negative rules of types $\sim X \rightarrow Y$ and $\sim X \rightarrow \sim Y$ (see Sec. 4.2.2) are excluded, even though they were considered by Antonie and Zaïane in [Ant04b]. In effect, we obtained the positive and negative association rule mining algorithm, called PANAMA, which is appropriate for hyperlink recommendation, Fig. 4.1.

There is an important measure useful for simultaneous mining of both negative and positive association rule mining: correlation coefficient - $correlation(X,Y)$. It denotes the strength of the linear relationship between two independent variables X and Y, i.e. the potential left side X and right side Y of a rule. It has been discussed in the context of association rules in [Ant04a, Ant04b, Tan02c, Xio04]. In practice, the well known Pearson's formula and contingency tables are used to calculate this correlation measure. If $correlation(X,Y)$ between X and Y is positive, then only the positive rule $X \rightarrow Y$ is considered. With a negative value of $correlation(X,Y)$ we expect a negative rule $X \rightarrow \sim Y$.

A few significant changes have been proposed compared to the solution presented in [Ant04b]. The key one is the introduction of candidate $X \rightarrow Y$ matching with the hyperlink set to the PANAMA algorithm, lines 12 and 17. Owing to the above, unnecessary, expensive calculation of support and confidence for candi-

The PANAMA Algorithm

```
 Input:  D - multiset of user sessions, transactional
             database,
         links - set of links from the entire web site,
         ρmin - minimum Pearson's correlation coefficient,
         minsup - minimum support,
         minconpos - minimum positive confidence,
         minconneg - minimum negative confidence.
 Output: posAR - set of all extracted positive rules,
         negAR - set of all extracted negative rules.

1posAR = ∅; negAR = ∅;
2Generate frequent 1-itemsets F1   /* F1 is the initial set */
3for (k = 2; Fk-1 <> ∅; k++){
4Ck = Fk-1 |><| F1;      /* |><| joins all items from Fk-1 with
                            items from F1 */
 5 foreach i ∈ Ck {
 6  s = sup(i);          /* support of itemset i within D */
 7  if s > minsup then
 8   Fk = Fk ∪ {i};    /* itemset i is added to Fk for the next
                          iteration */
 9  foreach X,Y (i=X∪Y, X∩Y=∅) {    /* all binary partitions
                                       of i */
10   ρ = correlation(X,Y);
11   if ρ ≥ ρmin then     /* correlation is positive => consider
                            a positive rule */
12    if (X,Y)∉links or k>2 then /* X,Y are 1-item; link X=>Y*/
13     if s ≥ minsup then           /* sup(X,Y)=s=sup(i) */
14      if con(X,Y) ≥ minconpos then
15       posAR = posAR ∪ {X→Y}
16   if ρ ≤ -ρmin then    /* correlation is negative => consider
                            a negative rule */
17    if (X,Y)∈links or k>2 then /* X,Y are 1-item; link X=>Y*/
18     if sup(X,~Y) ≥ minsup then     /* support for X∈S+ and
                                         Y∈S- in D */
19      if con(X,~Y) ≥ minconneg then
20       negAR = negAR ∪ {X→~Y}
21}}}
```

Fig. 4.1. The PANAMA algorithm for mining both positive and negative association rules in the web environment

date rules can be avoided. A rule that does not correspond to a hyperlink is useless, and if we want to use the PANAMA algorithm only to classify existing hyperlinks, this improvement reduces computations. However, in the case of a full recommendation system, which also includes suggestions of new links (see Sec. 4.4), line 12 should be removed. This excludes positive rules that are the basis for potential recommendations of new hyperlinks. Note that the matching with hyperlinks is necessary only for 1-itemsets (k=2). For larger itemsets (k>2) all their components have previously been validated for k=2. The correctness of this statement comes from the reasoning similar to candidate pruning - the apriori principle of frequent itemset mining: if an itemset is frequent, then all of its subsets must also be frequent.

The next change is the introduction of support calculation for negative rules - line 18. The algorithm originally presented does not contain this important operation.

Additionally, two separate confidence thresholds were introduced in the PANAMA algorithm. Negative association rules tend to have much higher confidence values than positive ones. This mainly stems from the typical length of the session: its positive component (S^+) is much smaller than its complement S^- in P, i.e. $card(S^+)<<card(S^-)$. According to the experiments from Sec. 4.5.2, the average ratio $card(S^-)/card(S^+)$ is from 40 to 1000. This was also indirectly confirmed by the distributions presented in Sec. 4.5.4. Thus, it appears that the negative threshold *minconneg* should usually be greater than the positive *minconpos*.

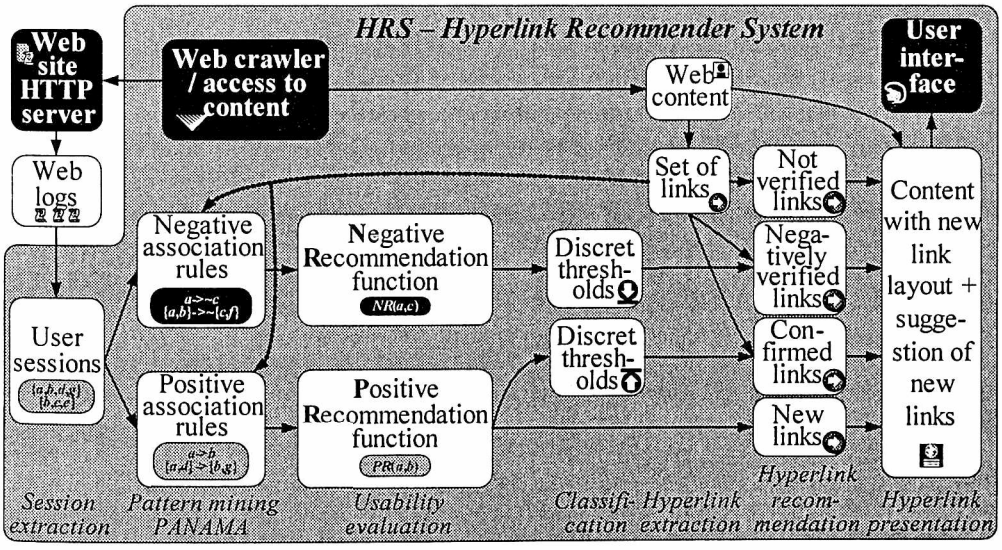

Fig. 4.2. The concept of the Hyperlink Recommender System (HRS) based on positive and confined negative association rules extracted from web logs

4.4. HRS - The Hyperlink Recommender System Based on Positive and Negative Association Rules

4.4.1. General Concept

Positive and negative association rules, which are extracted from data concerning user behaviour in the web site, provide important information about both the usefulness of hyperlinks existing in the site and the lack of some connections that could potentially be helpful for users. Strong positive rules $X \rightarrow Y$ outgoing from page $p_i \in X$ can be used to confirm hyperlinks leading from page p_i to pages $p_j \in Y$ if any such hyperlinks exist on the page p_i. In the case of non-existence of hyperlinks $p_i \Rightarrow p_j$, these positive patterns can be a hint for introduction of new hyperlinks $p_i \Rightarrow p_j$ on page p_i. A strong rule is the rule that has relatively high value of its confidence. By analogy, confined negative association rules $X \rightarrow \sim Y$ are signs of uselessness of hyperlinks eventually existing on page $p_i \in X$ and pointing to pages $p_j \in Y$.

Therefore, based on both positive and negative association rules as patterns of typical user behaviour and indirectly user needs, we are able to build the Hyperlink Recommender System (HRS). It can help content managers to adjust the structure of their sites to the preferences of their customers. HRS utilizes association rules to classify existing links into categories of good or bad as well as to identify new, potentially useful connections. Afterwards it recommends all these links to the content manager as positively verified, negatively verified or new ones (Fig. 4.2).

The entire process of recommendation consists of several steps: data preprocessing with session identification, content processing (hyperlink extraction), association rule mining, recommendation function calculation (rule merging), classification, and recommendation of hyperlinks to the user (Fig. 4.2).

In the first step, HRS recognizes user sessions from the log files that contain consecutive HTTP requests and are accumulated by almost every web server. Since it is assumed that the web site is anonymous and no user identification is available, session extraction can be performed with lower or higher accuracy [Chen03, Pie03, Tan02a], see Sec. 4.3.1.

Based on the identified sessions, both positive and negative association rules are discovered using the PANAMA algorithm, see Sec. 4.3.3. Only rules that exceed the given thresholds of minimum support and minimum confidence are passed for further processing.

Both positive and confined negative rules operate on sets of pages (Definition 4.2 and 4.3) whereas hyperlinks join single pages. For that reason, we need to interpret the sets in the context of their individual elements. Moreover, there may exist many separate rules that refer to a pair of pages, i.e. a single hyperlink $p_i \Rightarrow p_j$. To take into

consideration all the rules that are related to the single pair $p_i \Rightarrow p_j$, the rule clustering mechanism has been introduced, see Sec. 4.4.2. Its main idea is to gather all related complex rules and treat them as simple ones, as shown in Fig. 4.3. A complex rule is the rule for which cardinality of either its left or right side is greater than 1. A simple rule involves only single pages, e.g. $\{p_2\} \rightarrow \{p_4\}$, see also Sec. 4.2.2.

There are two positive complex rules in Fig. 4.3 that can be reduced to three simple rules. A complex rule may be related to many simple ones, e.g. $\{a\} \rightarrow \{b,c\}$ influences both $\{a\} \rightarrow \{b\}$ and $\{a\} \rightarrow \{c\}$. Simultaneously, a simple rule $\{a\} \rightarrow \{b\}$ is directly related to two complex rules. A similar case occurs in negative rules: one complex rule refers to four simple rules. After aggregation of all related rules into either a Positive Recommendation function (*PR*) or a Negative Recommendation function (*NR*), we obtain a single value for a pair of pages. For example, three rules: $\{a\} \rightarrow \{b\}$, $\{a\} \rightarrow \{b,c\}$, and $\{a,d\} \rightarrow \{b\}$ are clustered into one value addressed to pair $a \Rightarrow b$.

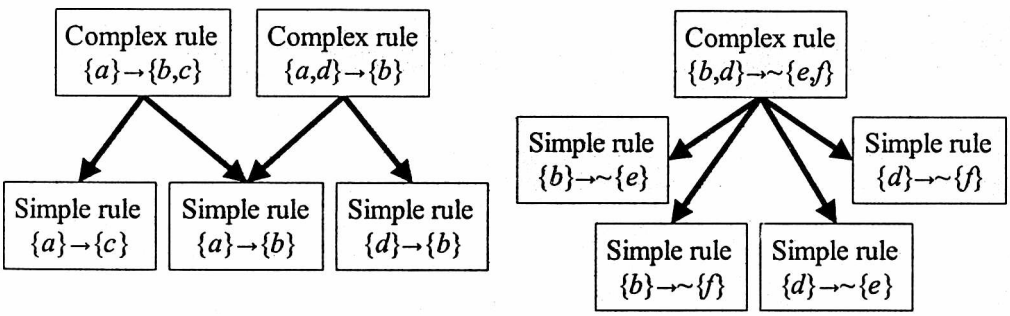

Fig. 4.3. Relationships between complex and simple rules

Values of recommendation functions can be discretized by application of appropriate thresholds. In this way, pairs of pages that have corresponding recommendation functions can be classified into several classes of higher or lower usability, see Sec. 4.4.3. Suppose that there are two pages *a* and *b* that belong to the same site and there is a hyperlink from page *a* to *b*. A high value of the Positive Recommendation function *PR*(*a*,*b*), which has been evaluated upon user behaviour, supports the correctness and usefulness of link $a \Rightarrow b$. Similarly, the existence of a Negative Recommendation function *NR*(*a*,*b*) suggests that the link $a \Rightarrow b$ is incorrect or useless. If the hyperlink $a \Rightarrow b$ does not yet exist on page *a*, significant value of Positive Recommendation function *PR*(*a*,*b*) means that the insertion of the new hyperlink $a \Rightarrow b$ should be recommended to the content manager.

However, to provide recommendation based on values of recommendation functions, we need to match these values with hyperlinks extracted from the web HTML content. Due to the dynamic character of web services, this matching is

never 100% effective, see Sec. 4.3.1. Moreover, the set of hyperlinks can make the rule mining algorithm more efficient (see lines 12 and 17 in the PANAMA algorithm, Sec. 4.2.2). This is also marked with dotted arrows in Fig. 4.2. On the other hand, the introduction of this improvement results in a lack of positive association rules that do not correspond to hyperlinks. In consequence, recommendation of new links cannot be discovered. For that reason, if we want to provide suggestions of new connections, line 12 of the algorithm should be removed. The remaining line 17 still reduces calculations, especially that the number of negative rules is usually greater than the number of positive ones, see Table 4.5.

With matching existing hyperlinks with discretized values of Positive and Negative Recommendation functions, we finish the hyperlink verification process, see Sec. 4.4.4. In its output, some links have been verified more or less positively, more or less negatively or they have not been assessed at all. Additionally, some new hyperlinks can be suggested based on high values of the Positive Recommendation function that do not match any existing links.

All these positively or negatively verified hyperlinks, together with the new ones, are presented to the content manager who, based on this knowledge, can appropriately modify the content of the web site by removal of some useless links, promotion of the most useful ones (e.g. by moving them to the top in the list), or insertion of some new links that do not yet exist but appear to be useful for users.

Since both the site content and behaviour of users change in the course of time, the entire process should be periodically repeated to make the site more adaptable to user needs.

4.4.2. Positive and Negative Recommendation Functions - Rule Merging

There exists one difficulty with hyperlink recommendation based on association rules. In general, rules of the form $X \rightarrow Y$ or $X \rightarrow \sim Y$ operate on sets of elements, i.e. both X and Y can consist of many web pages (see Definition 4.2 and 4.3). Moreover, there may be many rules $X_k \rightarrow Y_l$ for a pair of pages p_i and p_j such that $p_i \in X_k$ and $p_j \in Y_l$. Since a hyperlink $p_i \Rightarrow p_j$ in the web joins only two single pages: from p_i to p_j, according to the HTML standard, we expect only one simple measure corresponding to each such pair. Hence, we have to introduce an integration mechanism applied to all association rules extracted from web logs if we want to use them for classification of hyperlinks.

All positive rules $X_k \rightarrow Y_l$ that contain p_i on their left side ($p_i \in X_k$) and p_j on their right side ($p_j \in Y_l$) are exploited in the Positive Recommendation function for a pair of pages p_i and p_j. In particular, the Positive Recommendation function $PR(p_i,p_j)$ is based on the quality measure of its component positive association rules, i.e. confidence (see Sec. 4.2.1), as follows:

$$PR(p_i,p_j)=\frac{\sum\{con(X\rightarrow Y):p_i\in X,p_j\in Y\}}{card(\{X\rightarrow Y:p_i\in X,p_j\in Y\})}. \tag{4.5}$$

Similarly, the Negative Recommendation function $NR(p_i,p_j)$ for a pair of pages p_i, p_j makes use of confidence values assigned to confined negative association rules $X_k\rightarrow\sim Y_l$ (see Sec. 4.2.2) related to both p_i, p_j. It means that $p_i\in X_k$ and $p_j\in Y_l$. Negative Recommendation function $NR(p_i,p_j)$ is defined in the following way:

$$NR(p_i,p_j)=\frac{\sum\{con(X\rightarrow\sim Y):p_i\in X,p_j\in Y\}}{card(\{X\rightarrow\sim Y:p_i\in X,p_j\in Y\})}. \tag{4.6}$$

Each complex rule $X_k\rightarrow Y_l$ is involved in many pairs of pages p_i, p_j and in consequence influences many values of $PR(p_i,p_j)$. The same is valid for negative rules. For partitioning X, Y of the frequent itemset (line 9 in the PANAMA algorithm, Sec. 4.3.3) only one of three cases is possible:

1. there exists a positive association rule (lines 11–15);
2. there exists a negative rule (lines 16–20);
3. or there is no rule at all.

Therefore, for the given pair p_i, p_j we have either a non-zero value of $PR(p_i,p_j)$, a non-zero value of $NR(p_i,p_j)$ or we have not got any premisses for recommendation.

Since the confidences of all the component rules have to accomplish minimum confidence threshold, i.e. *minconpos* for positive and *minconneg* for negative rules, the values of $PR(p_i,p_j)$ and $NR(p_i,p_j)$ are always greater than *minconpos* or *minconneg*, respectively. In other words, $PR(p_i,p_j)\in[minconpos;1]$ and $NR(p_i,p_j)\in[minconneg;1]$.

It follows from the experiments performed that the number of negative recommendations usually exceeds the number of positive ones by severed times, see Sec. 4.5.4, Fig. 4.7 and Table 4.5. Moreover, the values of Negative Recommendation function *NR* are on average about 2.5 times greater than values of Positive Recommendation function *PR* (Table 4.5).

4.4.3. Classification of Recommendation Values

Positive and Negative Recommendation functions provide continuous values that reflect the usefulness or uselessness of relations between pairs of web pages. However, from a practical point of view these values are rather incomprehensible for the content manager, the user of HRS. Therefore, it appears to be helpful to create a few fixed positive and negative classes (intervals) separately for *PR* and *NR* and assign each of the *PR* and *NR* values to one of them. The positive intervals can be obtained by simple partitioning of the *PR* domain. Each *k*th positive

class is represented by the kth threshold τ^p_k, i.e. the limit inferior of the kth positive interval, whereas the limit superior comes from the upper, k+1th class, that is, τ^p_{k+1}. In other words, PR value belongs to the kth class if $\tau^p_k \le PR < \tau^p_{k+1}$. The limit superior of the top class is 1. The threshold for the first class is at least the minimum value of PR, e.g. $\tau^p_1 = minconpos$ but it can be somewhat greater.

The same discretization process is applied to the Negative Recommendation function, but both the number of classes and particularly individual thresholds can be different (Fig. 4.4). This separate treatment of negative classes compared to positive ones stems from their different distributions, see Sec. 4.5.4, Fig. 4.7.

To enable easy interpretation by HRS users, the number of classes should be between 1 and 3 both for positive and negative classes. Two positive and two negative intervals have been used in the implementation of HRS, see Sec. 4.5.4.

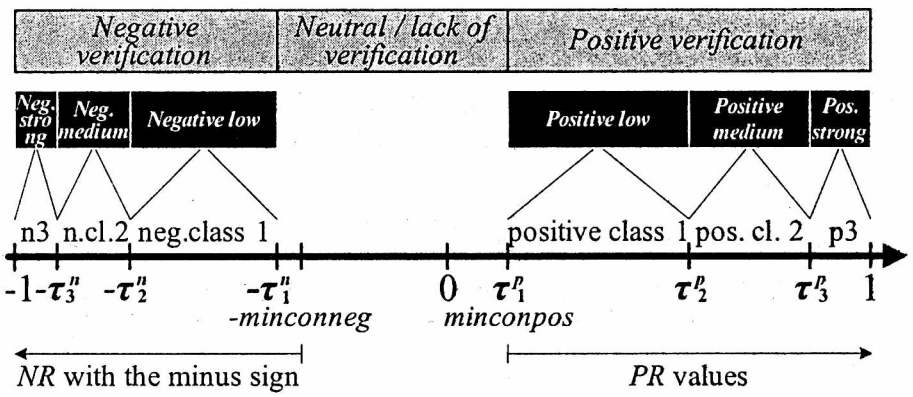

Fig. 4.4. Discretization with the use of three positive and three negative classes; $\tau^p_1 = minconpos$, $\tau''_1 > minconneg$

4.4.4. Hyperlink Recommendation

A Positive Recommendation function denotes how much a typical user who visits page p_i is also likely to visit page p_j during one session. Therefore, we can suppose that the hyperlink from p_i to p_j is useful if the value of $PR(p_i,p_j)$ is high enough, i.e. pair $p_i \Rightarrow p_j$ belongs to one of the positive classes. Moreover, the higher the class, the greater the usefulness. Such hyperlink $p_i \Rightarrow p_j$ should be preserved, if it already exists, or inserted into the HTML content of page p_i if it does not exist. On the contrary, the high value of Negative Recommendation function $NR(p_i,p_j)$ indicates that users visiting page p_i usually do not come to page p_j. This happens when $NR(p_i,p_j)$ belongs to any negative class, i.e. at least $NR(p_i,p_j) \ge \tau^n_1$. In consequence, if there exists a hyperlink $p_i \Rightarrow p_j$, it should be considered for removal. Note that according to line 17 in the PANAMA algorithm, only negative rules that corre-

spond to existing hyperlinks are mined. Therefore, every $NR(p_i,p_j)$ always possesses an equivalent hyperlink.

To extract hyperlinks from web pages, their HTML content needs to be processed. In order to obtain this content either a web crawler or direct access to the web server database or content management system (CMS) is necessary. Hyperlinks extracted from pages can be used to make the PANAMA algorithm more efficient (lines 12 and 17, the dotted line in Fig 4.2). However, to have new item recommendations, line 12 should be removed.

The list of hyperlinks has to be matched with both types of rules, i.e. with Positive or Negative Recommendation functions, or more precisely, with their discretized versions: positive and negative classes, see Sec. 4.4.3. Thus, one of the three cases is valid for every hyperlink $p_i \Rightarrow p_j$:

1. Positive recommendation. Hyperlink $p_i \Rightarrow p_j$ was assigned to one of the positive classes, $PR(p_i,p_j) \geq \tau^p_1$. This indicates that $p_i \Rightarrow p_j$ was confirmed (positively verified) and the certainty about this verification is greater for higher positive classes. Hyperlink $p_i \Rightarrow p_j$ should be preserved from removal.

2. Negative recommendation. Hyperlink $p_i \Rightarrow p_j$ was assigned to one of the negative classes, $NR(p_i,p_j) \geq \tau^n_1$. This means that $p_i \Rightarrow p_j$ was negatively verified. Hyperlink $p_i \Rightarrow p_j$ is recommended as useless and it probably should be removed from page p_i. This conviction is greater for higher negative classes.

3. No recommendation. Hyperlink $p_i \Rightarrow p_j$ was not verified. There is neither a positive class, $PR(p_i,p_j) < \tau^p_1$ nor a negative class, $NR(p_i,p_j) < \tau^n_1$, and for that reason $p_i \Rightarrow p_j$ cannot be assessed.

According to the discretization process (Sec. 4.4.3), the positive and negative recommendation can have several levels (classes); for example, "strong positive", "medium positive", and "low positive".

Note that not all existing hyperlinks can be classified (case 3) since the rule set does not have to cover all possible pairs of pages. This regards especially hyperlinks from pages which were not visited at all or were visited so rarely that rules did not reach minimum support. This feature of the HRS method even appears to be an advantage: we should not decide about usability of hyperlinks if the web pages on which they occur are not requested by the users. These pages themselves ought to be considered for removal.

Additionally, all positive $PR(p_i,p_j)$ values that do not have corresponding hyperlink $p_i \Rightarrow p_j$, can be used for recommendation of new items. In other words, HRS utilizes a fourth type of suggestion. The top n pages p_j for which values of $PR(p_i,p_j)$ are the greatest can be recommended by HRS on page p_i.

Having collated all four kinds of recommendations for the given page p_i (positively verified, negatively verified, not verified and new hyperlinks), the layout of page p_i is modified by means of the appropriate adaptation of HTML content

(Fig. 4.5). This enables the administrator or content manager to see them in their context and helps them to make the decision whether to delete or retain individual hyperlinks.

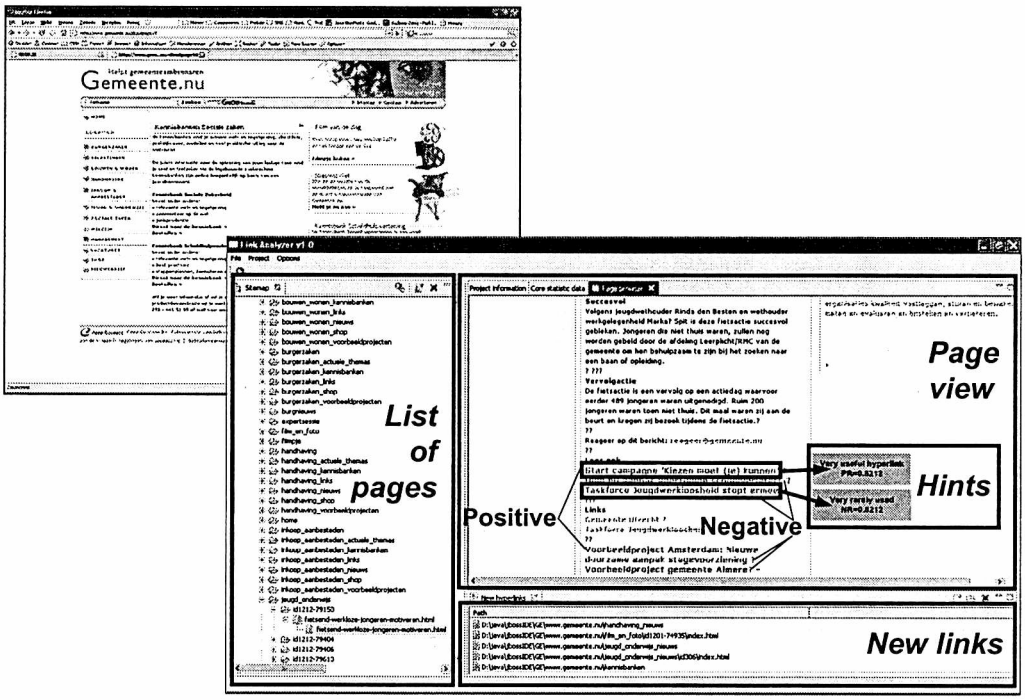

Fig. 4.5. The original layout of a page in the Gemeente (GE) web site and the same page with the layout modified by the application of HRS (*Link Analyzer*); actually, positively verified hyperlinks are in green whereas negative recommendations are in red; additional hints display values of either the Positive or the Negative Recommendation function; left bottom window contains suggestions of new hyperlinks

4.4.5. Disscusion - HRS Profile

The Hyperlink Recommender System makes use of data about user behaviour (web logs) and for that reason it facilitates the adaptation of the structure of the web site to typical user needs. It estimates the usefulness of existing hyperlinks both in a positive and negative way.

Verification does not depend on the real usage of individual hyperlinks, although in practice positively verified links are frequently used whereas negatively verified links are used very rarely for the experimental evidence, see Sec. 4.5.5. This results from the session profile: a session is an unordered set of pages with no regard for the sequence of navigation. Let us consider page p_1 that was

frequently visited together with page p_2, but between them another page p_3 was usually requested. If the appropriate rule $p_1 \rightarrow p_2$ has been discovered from web logs then rule $p_1 \rightarrow p_2$ and Positive Recommendation function $PR(p_1,p_2)$ confirms the potential usefulness of hyperlink $p_1 \Rightarrow p_2$ even though it has never been exploited by users. Moreover, the great value of $PR(p_1,p_2)$ and its high positive class suggest that hyperlink $p_1 \rightarrow p_2$ was probably wrongly located on page p_1 and for that reason users have not used it.

While assessing hyperlinks outgoing from page p_i, HRS takes into account the popularity of page p_i, the denominator in Eq. (4.2) and (4.4). Note that a 100-time usage of hyperlink $p_i \Rightarrow p_j$ when page p_i has been visited 100,000 times can be insignificant, whereas the same usage for page p_i that has been visited 100 times is the perfect indicator of $p_i \Rightarrow p_j$ usefulness.

Apart from the assessment of existing connections, HRS provides suggestions of new hyperlinks based on high values of $PR(p_i,p_j)$. The recommendation granularity can be tailored through the quantity of classes applied to classification, see Sec. 4.4.3. Thus, we can only have "good" - "not evaluated" - "bad" hyperlinks or "very good" - "medium good" - "good" - "not evaluated" - "bad" - "medium bad" - "very bad" ones.

HRS operates on both complex and simple rules, see Sec. 4.2.2, 4.4.1, and 4.4.2. Consequently, recommendations provided by HRS respect the broader context of navigation, that is, correlation between sets of pages visited together. A single hyperlink $p_i \Rightarrow p_j$ can be the bridge between the entire sections of the web site, and the Positive Recommendation function also partly reflects such a case by means of complex rule merging.

4.5. Experiments

To analyse features and check usability of the HRS method some experimental analyses have been carried out on real data. The experiments focused on negative and positive recommendations matched with existing hyperlinks and used for their assessment.

Positive association rules can also be utilized to suggest new hyperlinks, not yet existing on web pages. However, this kind of application has been analysed in numerous papers, e.g. [Chu04, Ger03, Mob01, Mob02, Wang02, Wang04] and therefore it has been passed over.

4.5.1. HRS Implementation

For testing purposes, the HRS method was implemented as the *Link Analyzer* application (Fig. 4.5). It was written in Java 5 as a standalone application that runs

in the Eclipse runtime environment. *Link Analyzer* executes all tasks of HRS (Fig. 4.2): log preparation with session identification, web site content and structure analysis (hyperlink extraction), positive and negative association rule mining, rule merging and classification, and finally, hyperlink layout modification by introduction of recommendations that could be used by a content manager to amend site structure.

Link Analyzer modifies the layout of all hyperlinks that have been either positively or negatively verified using green or red font colours, respectively, and it leaves non-verified links unchanged. Also, it maintains an additional window with the list of new hyperlinks that do not yet exist on the page (Fig. 4.5). According to the high values of the Positive Recommendation function, these new hyperlinks appear to be useful and the content manager should consider their addition to the currently viewed page. This last feature of *Link Analyzer* is not innovative, and therefore it was switched off in further experiments.

Table 4.1. General information about web sites and parameters used in experiments

	WUT	GE	DI	ZO
Total number of pages	892	2,668	150	6,562
Number of visited pages	847	2,250	131	5,803
Period investigated	6 weeks	6 weeks	6 weeks	6 weeks
Total number of HTTP requests	8,962,968	12,293,747	2,876,823	10,001,843
Number of requests for HTML resources	741,485	638,925	174,487	611,822
Number of HTML requests with corresponding pages	460,634	520,319	149,120	498,551
Number of HTML requests with corresponding pages including only requests from correct sessions	299,462	331,912	93,022	303,455
Number of sessions	139,484	170,589	91,731	167,903
Number of correct sessions	39,752	56,220	29,565	51,930
Average length of sessions	7.53	5.90	3.15	5.84
Number of hyperlinks	43,765	176,641	5,152	294,773
Number of hyperlinks per page	49.06	66,21	34,35	44,92
Number of hyperlinks on visited pages	39,528	166,849	4,184	218,963
Number of hyperlinks on visited pages per page	46.67	74.15	31.93	37.73
minsup	4 sessions	4 sessions	4 sessions	4 sessions
minconpos	20%	20%	20%	20%
minconneg	70%	70%	70%	70%

4.5.2. Test Environment

All experiments that are described in the further sections were performed on web sites of one of the universities in Poland and three Dutch online journals (Table 4.1). These sites were: Wrocław University of Technology, *www.pwr.wroc.pl* (WUT), Gemeente, *www.gemeente.nu* (GE), Distrifood, *www.distrifood.nl* (DI), and Zorg en welzijn, *www.zorgwelzijn.nl* (ZO). Some preliminary experiments on WUT data have been published in [Kaz06e]. Each of the sites changes its content relatively often. The profiles of the sites are presented in Table 4.1.

The total number of HTTP requests differed considerably from the number of requests for HTML resources. The reason was that the web pages usually contained many corellated elements like CSS, images, downloadable resources, etc. Requests for such components were not the subject of this research. An additional set of requests was omitted during matching requests with real pages of the system. Although the data had been tentatively cleaned up a lot of requests were left that could not be matched. There were several reasons for this. Firstly, it was the result of the dynamic nature of the site content: new pages appeared while the others were deleted. The logs considered contained entries covering 6 weeks. The second reason was virtual web servers that existed on the same physical server together with the analyzed one. Requests to these virtual servers cluttered up the logs. This case occurred especially for the WUT site, where many university departments shared the same server. Therefore, all requests for non-existing resources had to be removed.

It has been assumed that a correct user session can contain between 2 and 200 pages and the idle time between two consecutive requests does not exceed 30 minutes [Coo99]. Note that 1-page sessions are useless for analyses based on association rules.

The total number of hyperlinks has been arrived at based on the content analysis of the entire site. The number of hyperlinks on visited pages includes only those links that have been located on pages visited at least once within correct sessions during the period under investigation. About 10% of hyperlinks belonged to unvisited pages.

Figure 4.6 contains example screenshots from two sites being tested: GE and DI. The region marked with the dashed line is common for all pages in the site; it is the static core part of pages and this part is not considered in further studies. It contains menu items, search fields and buttons, links to contact pages.

For all the experiments, a common set of parameters has been used. To have a comprehensive view on association rules, the value of the minimum correlation coefficient ρ_{min} used to classify positive and negative rules was set to 0 (see lines 11 and 16 in the PANAMA algorithm, Sec. 4.3.3). Thus, all rules with positive value of coefficient ρ were classified as positive; otherwise they were negative. Note that zero values could not have occurred. The value of 4 sessions was assigned to the minimum support *minsup*. It meant that rules that occurred in at least 4 sessions were considered. Minimum confidence values were different for

positive and negative rules, i.e. *minconpos*=20% and *minconneg*=70%. The values of parameters were adjusted based on analysis of quantitative rule distribution.

Fig. 4.6. Overview of DI and GE sites. The dashed lines highlight the restricted areas, in particular menu components

4.5.3. Rule Lengths

As rule extraction is a very time-consuming process, it would be very useful to determine a real influence of complex rules on the final hyperlink classification. Note that classification comes from recommendation functions, i.e. either from *PR*, Eq. (4.5), or from *NR*, Eq. (4.6). For that reason, values of recommendation functions were calculated for the WUT site separately for nine data periods (six 1-week sets, two 3-week sets, and one 6-week set) based either on: only 2-element rules, 2- and 3-element rules (up to 3-), up to 4-, up to 5- or up to 6-element rules. Thus, there were five different kinds k of recommendation sets 2, 2-3, 2-4, 2-5, and 2-6, separately for each of the nine datasets. For example, k=6 meant that all rules containing from 2 to 6 elements were considered.

Next, two recommendation sets G_1 and G_2 based on the same k length of rules but extracted from two different datasets (session sets D_1 and D_2) were compared to each other using the extended Jaccard measure for weighted values $XJ^k(G_1,G_2)$, see also Eq. (5.11) and (6.3):

$$XJ^k(G^k_1,G^k_2)= \frac{\sum_{p_i,p_j\in P} R^k_1(p_i,p_j) * R^k_2(p_i,p_j)}{\sum_{p_i,p_j\in P}(R^k_1(p_i,p_j))^2 + \sum_{p_i,p_j\in P}(R^k_2(p_i,p_j))^2 - \sum_{p_i,p_j\in P} R^k_1(p_i,p_j) * R^k_2(p_i,p_j)}, \quad (4.7)$$

where k – maximum length of the rules processed, i.e. all rules of length up to k are considered, k=2, …, 6.

Table 4.2. Recommendation function comparison based on rule length (k=2, 3, 4, 5 and 6) using extended Jaccard measure for the WUT site

Dataset	1W1	1W2	1W3	1W4	1W5	1W6	3W1	3W2	6W
6W	0.5882 0.6429 0.6861 0.7098 0.7249	0.5678 0.6801 0.7326 0.7389 0.7420	0.5592 0.6360 0.7113 0.7138 0.7190	0.5866 0.6482 0.7460 0.7495 0.7504	0.6014 0.6411 0.6952 0.7044 0.7086	0.5839 0.6591 0.7022 0.7086 0.7104	0.7815 0.8420 0.8845 0.8910 0.9009	0.7746 0.8335 0.9010 0.9066 0.9099	1
3W2	0.6583 0.7496 0.8401 0.8422 0.8439	0.6469 0.7503 0.8283 0.8299 0.8305	0.6692 0.7520 0.8436 0.8459 0.8460	0.6033 0.7093 0.7948 0.7960 0.7982	0.6502 0.7333 0.7992 0.8002 0.8020	0.6488 0.7256 0.8124 0.8141 0.8170	0.7809 0.8547 0.9196 0.9205 0.9230	1	
3W1	0.6074 0.7373 0.8077 0.8120 0.8149	0.6194 0.7293 0.8334 0.8378 0.8400	0.6005 0.7893 0.8533 0.8540 0.8570	0.6139 0.7404 0.8146 0.8160 0.8172	0.6592 0.7294 0.8108 0.8130 0.8133	0.6375 0.7031 0.8310 0.8360 0.8367	1		
1W6	0.6790 0.7355 0.8999 0.9009 0.9030	0.7031 0.8532 0.9218 0.9236 0.9245	0.6955 0.7612 0.8727 0.8727 0.8780	0.7245 0.7709 0.8814 0.8830 0.8842	0.7011 0.7204 0.8936 0.8961 0.8980	1			
1W5	0.8290 0.8774 0.9508 0.9521 0.9533	0.7487 0.8340 0.9003 0.9017 0.9030	0.7033 0.8103 0.8904 0.8920 0.8929	0.6665 0.7404 0.8464 0.8477 0.8503	1				
1W4	0.7285 0.7942 0.8731 0.8735 0.8749	0.6813 0.7906 0.8720 0.8720 0.8733	0.6956 0.7526 0.8692 0.8710 0.8722	1					
1W3	0.7145 0.7907 0.8290 0.8307 0.8326	0.6636 0.7320 0.8593 0.8603 0.8608	1						
1W2	0.7435 0.8113 0.8936 0.8966 0.8993	1							
1W1	1								

$R^k_1(p_i,p_j)$, $R^k_2(p_i,p_j)$ - the value of either the Positive Recommendation function $PR(p_i,p_j)$ or the Negative Recommendation $NR(p_i,p_j)$ from page p_i to page p_j calculated from rules of length up to k for set G_1 or G_2, respectively. When there was neither a Positive Recommendation $PR(p_i,p_j)$ nor a Negative Recommendation $NR(p_i,p_j)$ for the given two pages p_i p_j within set G_1, then $R^k_1(p_i,p_j)=0$ was assumed. The thresholds presented in Table 4.1 were used in the rule extraction process.

The results of the experiment for the WUT site are presented in Table 4.2. Each table cell contains five values of extended Jaccard similarity coefficient that correspond to separate k values: k=2, 3, 4, 5 and 6, in vertical order. Values for k=2 and k=3 were omitted to improve table legibility.

The results obtained showed that the greater the length of the rules, the greater the similarity between corresponding recommendations. However, the longer rules do not affect recommendation values as much since their quantity is very small compared to the number of shorter rules (Table 4.3). The difference in similarities was smaller when including longer rules rather than shorter ones. Moreover, the inclusion of 6- or even 5-element rules is insignificant. For example, adding 5-element rules in XJ^5 increases similarity of XJ^4 by 0.1% for XJ^4(3W1, 3W2), up to 3.5% for XJ^4(1W1,6W).

Table 4.3. Quantity of rule sets for different rule lengths. Percentages refer to the maximum rule set (up to 6 pages, the last column)

Data-set	No. of 2 page rules	No. of 3 page rules	No. of 4 page rules	Total up to 4 page	No. of 5 page rules	Total up to 5 page	No. of 6 page rules	Total rules (up to 6 page)
6W	44,457 (25.2%)	97,012 (54.9%)	30,822 (17.5%)	172,291 (97.6%)	3,925 (2.2%)	176,216 (99.8%)	390 (0.2%)	176,606 (100%)
3W2	29,921 (25.2%)	68,294 (57.5%)	18,101 (15.2%)	116,316 (97.9%)	2,267 (1.9%)	118,583 (99.8%)	228 (0.2%)	118,811 (100%)
3W1	31,439 (24.8%)	72,329 (57.0%)	20,252 (16.0%)	124,020 (97.8%)	2,588 (2.0%)	126,608 (99.8%)	199 (0.2%)	126,807 (100%)
1W6	19,292 (29.3%)	33,120 (50.3%)	12,011 (18.2%)	64,423 (97.9%)	1,266 (1.9%)	65,689 (99.8%)	146 (0.2%)	65,835 (100%)
1W5	23,043 (32.1%)	35,463 (49.5%)	11,922 (16.6%)	70,428 (98.3%)	1,143 (1.6%)	71,571 (99.9%)	104 (0.1%)	71,675 (100%)
1W4	23,928 (36.8%)	29,021 (44.7%)	10,853 (16.7%)	63,802 (98.2%)	1,056 (1.6%)	64,858 (99.8%)	101 (0.2%)	64,959 (100%)
1W3	18,198 (31.9%)	27,780 (48.7%)	9,991 (17.5%)	55,969 (98.1%)	998 (1.7%)	56,967 (99.9%)	83 (0.1%)	57,050 (100%)
1W2	21,134 (32.7%)	30,943 (47.9%)	11,211 (17.4%)	63,288 (98.0%)	1,187 (1.8%)	64,475 (99.9%)	89 (0.1%)	64,564 (100%)
1W1	20,096 (29.5%)	33,918 (49.7%)	12,876 (18.9%)	66,890 (98.1%)	1,203 (1.8%)	68,093 (99.9%)	92 (0.1%)	68,185 (100%)

The percentage of 4-element rules within all rules (with up to 6 elements) was only from 15.2% to 18.9%, for 5-element rules this indicator was less than 2.2% and for 6-element rules only up to 0.22%. Therefore, it was decided that only rules with up to 4 elements were to be used in further experiments.

4.5.4. Hyperlink Classification

To enable hyperlink recommendation, the content links recognized on web pages need to be classified using positive and negative rules, see Sec. 4.4.3 and Fig. 4.4. Rules extracted from user sessions were merged (see Sec. 4.4.2) into rule groups using the Positive Recommendation *PR*, Eq. (4.5) or the Negative Recommendation function *NR*, Eq. (4.6). However, to classify hyperlinks the appropriate thresholds had to be fixed and applied, see Sec. 4.4.3. For that reason, the distribution of *NR* and *PR* function was analyzed experimentally for 3W1 sets from WUT, GE, and DI sites (Fig. 4.7). The thresholds were not used at all, i.e. *minconpos* = *minconneg* = 0.

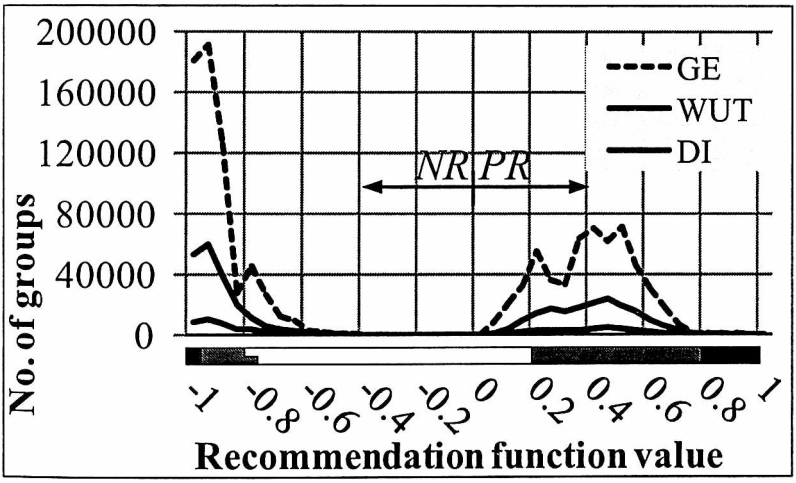

Fig. 4.7. Distribution of *PR* and *NR* (with the minus sign) for WUT, GE and DI sites

The total number of negative groups essentially differs from positive ones; there are from 2.5 (for GE) to 3.7 (for DI) times more negative groups than positive, depending on the site (Fig. 4.7, Table 4.5). This is chiefly due to the relationship between session length and the number of pages in the web site – there were about two orders of magnitude more pages than the average length of sessions. This difference in quantity and distribution justifies the usage of separate thresholds for positive and negative recommendations, see Sec. 4.4.3.

Table 4.4. Threshold values used for hyperlink classification assigned according to distribution (Fig. 4.7)

Recommendation class	WUT	GE / DI / ZO	Class no.
Very frequently used link	0.8	0.8	2
Good link	0.2	0.2	1
Rarely used link	0.8	0.75	-1
Very rarely used link	0.95	0.95	-2
Not evaluated	$PR < 0.2$ or $NR < 0.8$	$PR < 0.2$ or $NR < 0.75$	0

To simplify the experiments for experts (see Sec. 4.5.6) only five classes were defined for the existing hyperlinks. The two positive classes were *good links* and *very frequently used links* (the best), the two negative ones were *rarely used links* and *very rarely used links* (the worst); hyperlinks for which there was no rule group were classified as *not evaluated*, Table 4.4.

Table 4.5. Quantitative summary of hyperlink recommendations

	WUT	GE	DI	ZO
Sets of *PR* and *NR* values				
PR	82,092 (22.0%)	333,374 (28.3%)	14,866 (21.2%)	998,024 (30%)
NR	290,924 (78.0%)	843,324 (71.7%)	55,273 (78.8%)	2,329,061 (70%)
Total $NR \cup PR$ values	373,016 (100%)	1,176,698 (100%)	70,139 (100%)	3,327,085 (100%)
$PR<0.2$ or $NR<0.8$ (WUT) or $NR<0.75$ (GE/DI/ZO)	14,447 (3.9%)	44,834 (3.8%)	2,987 (4.3%)	107,998 (3,1%)
$PR\geq0.2$ or $NR\geq0.75$ or 0.8	358,569 (96.1%)	1,131,864 (96.2%)	67,152 (95.7%)	3,327,085 (96,9%)
Hyperlink recommendations				
Very frequently used links	2,937 (7.4%)	19,593 (11.7%)	845 (20.2%)	29,674 (13.6%)
Good links	9,257 (23.4%)	35,770 (21.4%)	1,033 (24.7%)	42,888 (19.6%)
Rarely used links	2,437 (6.2%)	2,437 (1.5%)	429 (10.3%)	9,532 (4.4%)
Very rarely used links	8,477 (21.4%)	52,438 (31.4%)	1,233 (29.5%)	82,932 (37.9%)
Not evaluated links	16,420 (41.5%)	56,611 (33.9%)	644 (15.4%)	53,937 (24.6%)
Total links	39,528 (100%)	166,849 (100%)	4,184 (100%)	218,963 (100%)

Based on the distributions from Fig. 4.7, almost the same thresholds applied to recommendation functions were used for all the web sites, i.e. 80% and 20% for positively classified links (*PR*) as well as 95% and 75% (GE, DI) or 80% (WUT) for negative recommendations (*NR*), as shown in Table 4.4. Since ZO data had similar distribution to GE data, GE's thresholds were utilized for recommendation in the ZO site.

Having fixed the appropriate intervals for each class, the system was able to classify (recommend to the content manager) hyperlinks extracted from the HTML content by matching them with the previously obtained rule groups.

The general results of hyperlink classification are gathered in Table 4.5. The most common group for positive recommendations were *good links*, while *very frequently used links* were two times less numerous. The differences between the two negative recommendation classes are even more considerable. *Very rarely used links* were the most common group of links; up to 96% of all negative recommendations for the GE site.

There was also a significant number of *not evaluated* hyperlinks. These ranged from 15% for the DI site to 41% for the WUT site, which was in particular due to the thresholds used in the experiment, especially *minsup*. In the case of WUT there were many pages that were either visited only several times within the period investigated or not visited at all. Since there were no association rules that would match hyperlinks outgoing from such pages or the rules had too little support, all such links were automatically treated as *not evaluated*. The other minor reasons were both the confidence thresholds *minconpos*=20% and *minconneg*=70% and the thresholds applied to recommendation function *PR* and *NR* (Table 4.4). Nevertheless, the thresholds are responsible only for a few links not being evaluated (see, for example, the fourth row in Table 4.5).

4.5.5. HRS vs. Referer Field

The optional referer field in web logs contains the URL of the page that was visited before the requested one [HTTP92]. However, this field may be left empty, for example when users provide URLs of pages they want to access directly in their browsers. Typically, the web browser fills in the referer field automatically with the URL of the page that contains a just-clicked hyperlink. If page p_1 is included in the referer field inside the request for page p_2, then such an entry in the log supports the usability of the hyperlink from p_1 to p_2.

The experiments with the use of a referer field were conducted on two consecutive 3-week datasets separately for each of the four web sites: WUT, GE, ZO, and DI. The first dataset was used to extract association rules and to recommend hyperlinks in the positive (very frequently used, good) or negative way (rarely or very rarely used), (see Table 4.4). The second dataset was utilized to verify the classified hyperlinks against the referer field. Each request in the logs that has filled in the referer field must correspond to the existing hyperlink. Hence, a hyperlink may be assessed twice: firstly, by classification function derived from positive and negative association rules and secondly, by frequency analysis of referer→requested page entities in the log data. Positively recommended hyperlinks used more frequently than four times, or negatively recommended ones used at most four times are considered as successfully verified by the referer field.

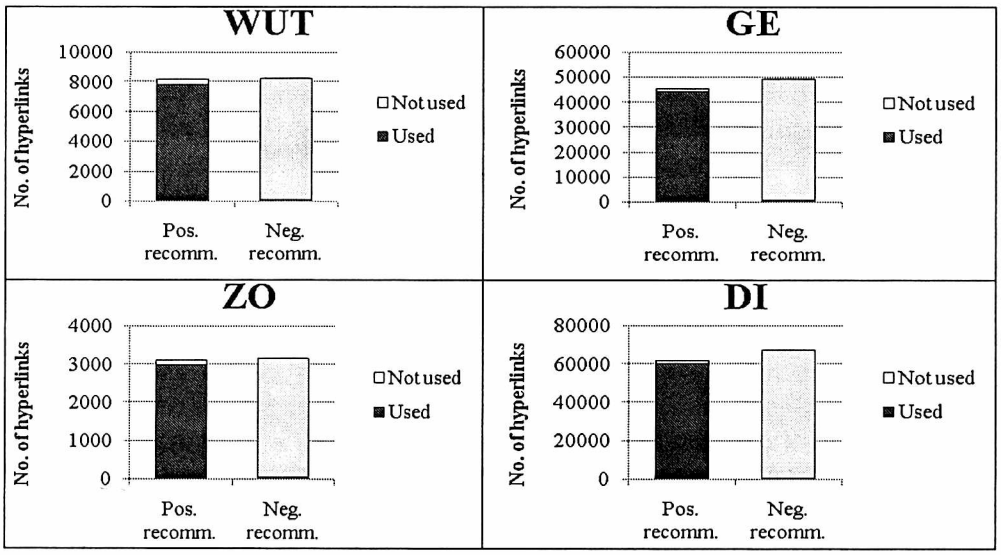

Fig. 4.8. The usage of hyperlinks positively or negatively recommended verified by referer field analysis for four web sites

Table 4.6. Positive and negative recommendations verified with the referer field

	WUT	GE	DI	ZO
Correct links used frequently enough (>4 times in the referer field)	7,762 (95%)	44,437 (97%)	60,310 (97%)	2,969 (96%)
Correct links never or rarely used (≤ 4 times)	376 (5%)	1,182 (3%)	1,663 (3%)	129 (4%)
Incorrect links never or rarely used (≤ 4 times)	8,081 (98%)	48,812 (99%)	66,651 (99%)	3,107 (99%)
Incorrect links used frequently (>4 times)	143 (2%)	582 (1%)	612 (1%)	46 (1%)

As we can observe, the positively recommended hyperlinks are also used more than 4 times relatively frequently according to the referer field and this refers to over 95% of them (Table 4.6 and Fig. 4.8). Similarly, over 98% of negatively recommended hyperlinks are either used very rare, at most 4 times, or not used at all.

Note that referer field analysis does not replace the HRS approach.

4.5.6. Expert Verification

The main goal of the Hyperlink Recommender System is to suggest new hyperlinks and to provide positive as well as negative assessment of hyperlinks already existing on web pages. Since these negative recommendations are the most inno-

vative component of the method, they were verified by independent content managers responsible for the web sites used in the experiments. There were two experts from Reed Business Information and one from WUT. The experiment was conducted on all the four sites but due to organizational purposes only on hyperlinks derived from a small set of content pages. With the help of *Link Analyzer*, (see Sec. 4.5.1) the experts had the possibility to provide their opinions about negatively verified hyperlinks located on selected content pages. They were supposed to set one of the two notes about the recommendation: 'I agree' meaning that the link really was incorrect and should probably be removed or 'I disagree' meaning that an expert would leave the link unchanged. During the experiment, it was necessary to add a third category - 'Core part'. This category meant that even though an expert did agree with the evaluation, it was not possible to remove the link as it belonged to general graphical design of the page or even the entire site. The experiment was carried out with the following parameters for all sites: *minconpos*=0.2, *minconneg*=0.7, *minsup* covered 4 sessions, period=3 weeks. Table 4.7 and Fig. 4.9 gather results of the experiment.

Table 4.7. Experts' opinion on negative hyperlink recommendations

Expert's opinion	WUT	GE	ZO	DI
'I agree'	59 (77.6 %)	76 (69.7%)	27 (77.1%)	108 (81.2%)
'I disagree'	14 (18.4 %)	12 (11.0%)	3 (8.6%)	11 (8.3%)
'Core part'	3 (3.9 %)	21 (19.3%)	5 (14.3%)	14 (10.5%)
Total no. of links	76 (100 %)	109 (100%)	35 (100%)	133 (100%)

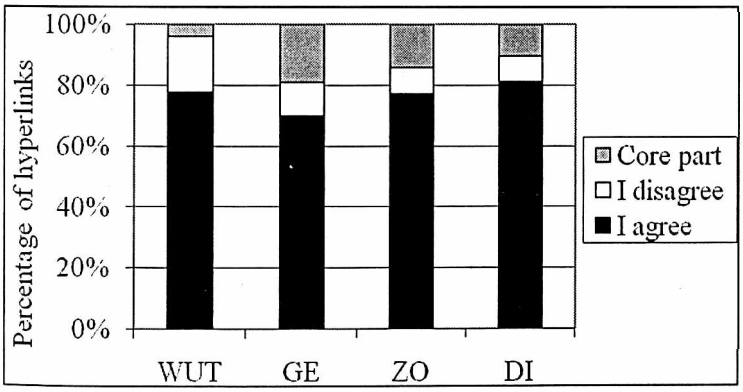

Fig. 4.9. Negative hyperlink recommendations verified by web site content managers

Note that the experts mostly agree with the HRS evaluation. The amount of 'I agree' choices ranged from almost 70% in the case of the GE site up to 81% for the DI site. This confirms the general effectiveness and usability of HRS.

Up to 20% of hyperlinks belonged to the core part of the site, e.g. a static menu. An additional outcome of the experiments on the WUT site were some new considerations related to the general concept of the site organization, including the common menu shared by all pages. The content manager recognized that student and employee parts should have separate navigational conceptions.

4.6. Conclusions

The concept presented in this section is a method for automatic positive and negative recommendation that provides evaluation of existing hyperlinks and suggestion of new ones. The Hyperlink Recommender System (HRS) is especially useful for web content managers. It takes advantage of historical user behaviour by using both positive and confined negative association rules extracted from web server logs (web usage mining). The Positive Recommendation function merges all positive rules and their confidences related to the given pair of web pages. Similarly, confidences of confined negative rules are components of the Negative Recommendation function. The Positive Recommendation function enables the system to estimate the usefulness of existing hyperlinks and to suggest new connections that are potentially useful for users whereas the high value of Negative Recommendation function is a significant sign of redundancy of the given hyperlinks. Since this knowledge is, in a sense, the condensed pattern of typical navigational behaviour and corresponds to user needs, the content managers can modify the structure of the web site by displacement of the most valuable hyperlinks to more prominent place on the web page, adding new ones or removing the most ineffective links.

HRS respects the relative popularity of the page that contains the hyperlinks considered, see Sec. 4.4.5. Additionally, recommendations provided by HRS can be more or less precise, by application of the appropriate number of classes at the stage of classification, see Sec. 4.4.3 and 4.4.5.

Experiments carried out on real web logs revealed that positive and negative rules and recommendation functions need to have separate parameters. Besides, the effectiveness of HRS has been borne out by real usage and non-usage of hyperlinks: the referer field (see Sec. 4.5.5) and verification performed by experts (see Sec. 4.5.6).

Nevertheless, it should be emphasized that the HRS recommendations provide only suggestions which have to be approved by the web site content manager. Moreover, some negatively verified and potentially useless hyperlinks have to be left on the page due to the general interaction concept, such as menu items or some policy restrictions: links to privacy remarks, authors or the contact page.

5 Sequential Patterns with Negative Conclusions

Sequential patterns are one of the typical methods within the data mining research domain. Apart from association rules mining (see Sec. 2 and 4), sequential pattern discovery can be classified as a technique for association (link) discovery. However, association rules operate on unordered items (sets), whereas sequential patterns respect the time assignment for events correlated with the items, see Sec. 2.

5.1. Background and Problem Description

Sequential patterns have been studied in many scientific papers for over ten years [Agr95, Ayr02, Pei01, Pei04, Zak01]. As a result, many algorithms for mining regular sequential patterns have been developed, including incremental and parallel ones, see Sec. 2.1.4. The more unique solutions include mining sequential patterns in streams [Ho06, Hua08, Li07], documents [Gar06], spatial-temporal databases [Hua08], as well as discovery of hierarchical [Pla06] or compressed [Chan06] sequential patterns. All these patterns are positive frequent subsequences included and discovered within all sequences from the source multiset.

All association rules and sequential patterns reflect positive patterns. Nevertheless, there exists another type of patterns – negative ones. Hence, in the case of association rules, we have negative association rules that indicate the negative relationship between two sets of objects, see Sec. 4. If the first set occurs in source transactions, another set does not co-occur or co-occurs very rarely in these transactions. Note that both the source transactions and output patterns operate on sets whereas sequential patterns refer to sequences in time.

This lack of negative patterns related to sequences resulted in the working out of the new kind of negative patterns. Thus, novel positive patterns that possess negative conclusions are introduced in this section. They, in a sense, combine frequent positive sequences (sequential patterns) and negative association rules.

This can be shortly described in the following way: if there is a frequent sequence q, then elements of set X usually do not occur after the sequence q.

This type of patterns enables the previously expected sequences to be verified in the negative way. For example, the management of the e-commerce web site supposes that their users usually terminate their visits with payments. Sequential patterns with negative conclusions extracted from web logs (navigational paths) can negatively verify this expectation. If users of the given e-commerce web site put certain products into their basket, afterwards they enter the first step of the purchase - personal information delivery (sequence q), and next, they are not likely to finish their buy with none of the payments (set X - the negative conclusion), then such a pattern debunks the beliefs of the e-commerce management.

Another application of sequential patterns with negative conclusions is verification of existing links between objects. These can be hyperlinks placed by authors or content managers on the web pages they manage, see Sec. 5.5. This, in its concept, is similar to verification based on negative association rules, see Sec. 4.4. Moreover, both negative association rules and sequential patterns with negative conclusions can be combined to provide more comprehensive negative knowledge about hyperlinks, see Sec. 5.5.3.

One more example of existing links, which can be negatively verified by sequential patterns with negative conclusions, are correlations extracted from web contents based on their textual analysis. This appears to be useful especially in recommender systems, in which content-based recommendation lists are verified by negative usage patterns, see Sec. 5.6.

5.2. Regular Sequential Patterns

Before we introduce the new kind of sequential patterns, let us get better acquainted with regular sequential patterns.

Definition 5.1. A sequence $t_i=<p_{i1},p_{i2},\ldots,p_{im_i}>$ in the domain set D is a time ordered list (tuple) of m_i items from domain D: $\forall 1\leq j\leq m_i\ p_{ij}\in D$. The number of elements in the sequence m_i is called the length of sequence t_i. Source sequences T in domain D are the multiset of sequences collected in the system.

Ordering according to time means that the time assigned to items occurring in the sequence is non-decreasing, i.e. $\forall 1\leq j\leq m_i\ time_{ij+1}\geq time_{ij}$. Note that the length of two sequences t_i and t_j in T may differ, i.e. $m_i\neq m_j$. Since T is the multiset there may be many the same sequences in T.

For instance, if $D=\{p_1,p_2,\ldots,p_N\}$ is the set of N web pages existing in the single web site, then source sequences are the multiset of all navigational paths accumulated

by the web server within the certain period. Hence, a navigational path is a single source sequence t_i=<p_{i1},p_{i2},..., p_{im_i}> whose all m_i items (pages) p_{ij} belong to the domain D of all web site pages. The example navigational path t=<p_4,p_6,p_1,p_6> means that the user first visited page p_4, next page p_6, p_1, and finished on page p_6.

The simple but quite effective path extraction from web logs consists in matching IP addresses and user agent fields from the HTTP requests gathered in the web logs. Additionally, some time constraints are applied. A user navigational path comprises all time-ordered requests that come from the same address IP_i and the same user agent with the idle time between two following requests of less than 30 minutes. Besides, the length of the path can be restricted to a few hundred.

In the multiset, repetitions are allowed, i.e. there may exist two separate source sequences with the same component elements equally ordered. For example, two different users a and b may navigate through the web site in the same way: t_a=t_b=<p_6,p_6,p_1,p_4,p_6>. Moreover, every source sequence t may contain repetitions in any places. In t_a, we have p_{a1}=p_{a2}=p_{a5}.

5.2.1. Subsequences and Their Complements

Definition 5.2. A sequence t_i=<p_{i1},p_{i2},..., p_{im_i}> contains another sequence q_j=<p_{j1},p_{j2},...,p_{jn}>, if there exist n integers k_1<k_2<...<k_n, called index K of q_j in t_i, such that p_{ik_1} = p_{j1}, p_{ik_2} = p_{j2}, ..., p_{ik_n} = p_{jn}. Sequence q_j is also called the subsequence of t_i. Item p_{ik_n} is called the end or the last item of q_j in t_i with respect to index K whereas its position in t_i, that is, k_n, is called the end position and it is denoted by k^{end}.

Overall, index K denotes positions of the subsequence q_j's items within the given sequence t_i.

Each sequence t_i may contain up to (2^{m_i} -m_i-1) different sequences q_j that consist of at least 2 items. For example, a navigational path t=<p_4,p_6,p_1,p_6>, i.e. the sequence, contains the following 2-, 3- and 4-item subsequences: <p_4,p_6>, <p_4,p_1>, <p_4,p_6,p_1>, <p_4,p_1,p_6>, <p_4,p_6,p_6>, <p_4,p_6,p_1,p_6>, <p_6,p_1>, <p_6,p_6>, <p_6,p_1,p_6>, <p_1,p_6>. Their number equals 10 and is less than the maximum (11) due to repetition of p_6. Note that subsequence <p_4,p_6> has two separate indexes K_1=(1,2) and K_2=(1,4); the end positions are k_1^{end}=2 and k_2^{end}=4, respectively.

Definition 5.3. A complement $C(q,t,K)$ of subsequence q in sequence t with respect to index K is the set of all items of t that follows the last item of subsequence q in t. The complement of subsequence q in t, which has the largest number of items, is called the maximum complement of q in t and is denoted by $C^{max}(q,t)$.

Obviously, q must be a subsequence of t according to Definition 5.2. For the example navigational path $t=<p_4,p_6,p_1,p_6,p_1,p_6>$ and the subsequence $q=<p_4,p_6>$, we have three end positions $k_1^{end}=2$, $k_2^{end}=4$, and $k_3^{end}=6$ for three corresponding indexes $K_1=(1,2)$, $K_2=(1,4)$, and $K_3=(1,6)$, respectively. Hence, complement $C_1(q,t,K_1)=C_2(q,t,K_2)=\{p_1,p_6\}$ and $C_3(q,t,K_3)=\varnothing$. The complement is not a multiset - repetitions are not allowed. The first two complements are the greatest so they are simultaneously the maximum complement $C^{max}(q,t)=C_1(q,t,K_1)=C_2(q,t,K_2)$.

Overall, the maximum complement corresponds to index K with the smallest value of end position $k_1^{end}=2$. Apparently, the number of different complements is less than or equal to the length of sequence t minus the length of subsequence q.

Note that for the given sequence t and its subsequence q, the maximum complement contains all other complements of q in t: $C_1 \subseteq C^{max}$, $C_2 \subseteq C^{max}$, and $C_3 \subseteq C^{max}$. Based on this feature, we can easily prove that if item p does not belong to the maximum complement $C^{max}(q,t)$, then item p does not belong to any of its subsets either, i.e. to any of the complements $C_i(q,t,K_i)$.

5.2.2. Support - a Measure for the Frequent Sequence, Sequential Patterns

Definition 5.4. Support $sup(q)$ of sequence q in T is the number of all source sequences from T that contain q (Definition 5.2):

$$sup(q)=\frac{card(\{t \in T : q \text{ is a subsequence of } t\})}{card(T)}. \tag{5.1}$$

Support can be expressed either as the regular number of source sequences or the percentage support in T's.

Definition 5.5. A sequence q is called a sequential pattern in T if its support is big enough, i.e. $sup(q) \geq minsup$, where $minsup$ is the minimum threshold.

The algorithms used to extract sequential patterns (frequent sequences) are enumerated in Sec. 2.1.4.

5.3. Sequential Patterns with Negative Conclusions

Definition 5.6. A sequential pattern q in T and set $X \subset D$ constitute a sequential pattern with the negative conclusion $s^-(q \rightarrow \sim X)$ if there are some source sequences $t_i \in T$ that contain q, for which set X does not intersect their complements $C(q,t_i,K_j)$ for any K_j.

Note that the empty intersection of set X with any of the complements $C(q,t_i,K_j)$ is equivalent to $X \cap C^{max}(q,t_i)=\varnothing$.

A sequential pattern with the negative conclusion $s^-(q \rightarrow \sim X)$, which has 1-item sequence $q=<p>$ on its left-hand side, is simply equivalent to the negative association rule $\{p\} \rightarrow \sim X$, see Sec. 2.1.3.

For the case of web user navigational paths, a sequential pattern with the negative conclusion denotes: if users have visited sequence q of web pages, afterwards, they usually visit none of the pages from set X. In other words, we do not expect any of the elements from X after sequence q. For example, the pattern $s^-(<p_4,p_6> \rightarrow \sim \{p_4,p_5,p_8\})$ means that if users visit page p_4 and then p_6, they visit neither p_4 nor p_5 nor p_8. The following source sequences support the above sequential pattern with the negative conclusion: $<p_1,p_4,p_3,p_6,p_1,p_2>$, $<p_1,p_4,p_5,p_4,p_5,p_8,p_6,p_1,p_3,p_1,p_6>$ whereas the sequences: $<p_4,p_3,p_6,p_4,p_2>$, $<p_4,p_6,p_4,p_6,p_2>$ (both due to the second p_4), and $<p_4,p_6,p_8>$ (due to p_8) do not match the pattern.

5.3.1. Measures of Sequential Patterns with Negative Conclusions

Similarly to association rules, each sequential pattern $s^-(q \rightarrow \sim X)$ with negative conclusion possesses two basic measures: support $sup^{s^-}(q \rightarrow \sim X)$ and confidence $con^{s^-}(q \rightarrow \sim X)$. The former is expressed as follows:

$$sup^{s^-}(q \rightarrow \sim X)=\frac{card(\{t \in T : q \text{ is a subsequence of } t \wedge C^{max}(q,t) \cap X = \varnothing\})}{card(T)}. \tag{5.2}$$

Confidence $con^{s^-}(q \rightarrow \sim X)$ is calculated in the following way:

$$con^{s^-}(q \rightarrow \sim X)=\frac{card(\{t \in T : q \text{ is a subsequence of } t \wedge C^{max}(q,t) \cap X = \varnothing\})}{card(\{t \in T : q \text{ is a subsequence of } t\})}. \tag{5.3}$$

Note that, see also Eq. (5.1):

$$con^{s^-}(q \rightarrow \sim X) = sup^{s^-}(q \rightarrow \sim X)/sup(q). \tag{5.4}$$

Only patterns $s^-(q \rightarrow \sim X)$ that exceed minimum thresholds are really considered, i.e. $sup^{s^-}(q \rightarrow \sim X) \geq minsup^{s^-}$ and $con^{s^-}(q \rightarrow \sim X) \geq mincon^{s^-}$. Since the quantity of the domain $card(D)$ is usually several orders of magnitude greater than the average length of source sequences, then typically $mincon^{s^-}$ has the value close to 1.

5.3.2. SPAWN - Mining Sequential Patterns with Negative Conclusions

Sequential patterns with negative conclusions can be discovered using the previously obtained set of regular sequential patterns q - any algorithm can be

The SPAWN algorithm

```
Input: T - the multiset of source sequences
       D - the domain, e.g. the set of web pages existing in
           the site
       minsup^s- - minimum support for sequential patterns
           with negative conclusions; expressed in number of
           sequences
       mincon^s- - minimum confidence for sequential patterns
           with negative conclusions; expressed in %
Output: Q - the set of regular, positive sequential patterns
        SUPQ - the set of support values for the appropriate
            patterns q∈Q
        S- - DB containing sequential patterns with negative
            conclusions
 1. extract regular sequential patterns q (with support);
    fill Q and SUPQ
 2. S- = empty
 3. for each q∈Q and sup_q∈SUPQ {
 4.   threshold_q = max (minsup^s-, sup_q*mincon^s-)
 5.   CDB_q = empty
 6.   for each t∈T
 7.     if q is the subsequence of t then
 8.       if C^max(q,t)≠∅ then
 9.         append C^max(q,t) to CDB_q
10.   CAND_1 = distinct_p (CDB_q)
11.   M_q = D / CAND_1
12.   check_and_add (CAND_1, sup_q, threshold_q, CDB_q, S-, R_1)
13.   k = 1
14.   while R_k is not empty {
15.     k = k+1
```

Fig. 5.1. The SPAWN algorithm for mining sequential patterns with negative conclusions

used for this purpose, see Sec. 2.1.4. Hence, having regular sequential patterns q previously mined, the frequent set of maximum complement $C^{max}(q,t_i)$ is extracted from source sequences t_i that contain each such regular sequential pattern q. Items from domain D (see Definition 5.1) that do not belong to any complement of these source sequences automatically become members of the negative pattern conclusion - set X. All other items that frequent maximum complement are treated as candidate members for conclusion set X. These candidates and their combinations X_i that exceed $minsup^{s-}$ and $mincon^{s-}$ thresholds form sequential patterns

```
16.      CAND_k = generate_candidates (R_k-1)
17.      check_and_add (CAND_k, sup_q, threshold_q, CDB_q, S⁻, R_k)
18.    }
19.    ABSENT_q = generate_candidates (M_q)
20.    for each X∈ABSENT_q {
21.      append s⁻(q→~X) to S⁻
22.      sup^s⁻ (q→~X) = sup_q
23.      con^s⁻(q→~X) = 100%
24.    }
25.    for each X∈ABSENT_q and each Y∈R_i { /* for all
                                              i=1,2,...,k*/
26.      append s⁻(q→~(X∪Y)) to S⁻
27.      sup^s⁻ (q→~(X∪Y)) = sup^s⁻ (q→~Y)
28.      con^s⁻(q→~(X∪Y)) = con^s⁻(q→~Y)
29.  }}            /* for each q∈Q and the entire algorithm*/

procedure check_and_add
Input: CAND_k, sup_q, threshold_q, CDB_q, S⁻
Output: R_k - rare itemsets, S⁻
30. for each X∈CAND_k {       /* for CAND_1, X is a 1-itemset */
31.    sup^s⁻ (q→~X) = sup_q
32.    for each C^max from CDB_q
33.       if X ∩ C^max ≠ ∅ then
34.          sup^s⁻ (q→~X) = sup^s⁻ (q→~X) - 1
35.    if sup^s⁻ (q→~X) ≥ threshold_q then {
36.      append X to R_k
37.      append s⁻(q→~X) to S⁻
38.      preserve sup^s⁻ (q→~X)
39.      con^s⁻(q→~X) = sup^s⁻(q→~X) / sup_q
40. }} /*for each X∈CAND_k and entire proc.check_and_add*/
```

$s^-(q \rightarrow \sim X_i)$ with negative conclusions. The process is repeated separately for each positive sequential pattern q. The above concept is used in SPAWN - the algorithm for discovering Sequential PAttern With Negative conclusions, Fig. 5.1.

Note that M_q is the set containing items from the domain D that do not occur in any complement of pattern q. For that reason, subsets of M_q can be utilized to create new negative conclusions and to supplement output patterns. This is achieved using either only subsets of M_q (lines 20-24) or by the extension of conclusions for the patterns that have been previously obtained in procedure *check_and_add* (lines 25-29).

The database scan (procedure *check_and_add*, lines 32–34) is performed only for the temporal CDB_q that contains maximum complements for the given regular pattern q, see Definition 5.4. Since the value of $mincon^{s^-}$ is usually closer to 1 rather than to 0, the concept of CDB_q scan consists in decreasing the initial, maximum support. This enables the loop (lines 32–34) to be interrupted after the support value falls below the minimal thresholds.

The meaning of selected lines in the SPAWN algorithm is as follows:

- Line 1: Use any algorithm to discover regular sequential patterns q.
- Lines 3–29: One run of the loop creates all sequential patterns with negative conclusion for a single regular sequential pattern q extracted in line 1.
- Line 3: sup_q is the support in T for the corresponding q; it is expressed in number of sequences.
- Line 4: $threshold_q$ is the min. number of sequences $t \in T$ that must contain searched patterns $s^-(q \rightarrow \sim X)$; $sup_q * mincon^{s^-}$ is rounded up to integers.
- Line 5: CDB_q is the database (list) that contains sets with non-empty maximum complements $C^{max}(q,t)$ of subsequence q for all t in T that contain q.
- Lines 6–9 fill CDB_q. The database CDB_q is in a sense similar to the α-projected database used in the PrefixSpan algorithm [Pei01].
- Line 10: find all distinct $p \in D$ that occur in CDB_q. $CAND_k$ is the set of candidates of length k.
- Line 11: M_q is the complement of $CAND_1$ in D. M_q contains items that never follow q in source sequences. For that reason, the elements of M_q can by default extend negative conclusions, see lines 19-29.
- Line 12: Procedure *check_and_add* (lines 30-40) tests the support for candidates from $CAND_1$ to find rare itemsets. From the rare enough sets, the new output patters $sup^{s^-}(q \rightarrow \sim X)$ are created together with their measures.
- Line 16: Function *generate_candidates* (R_{k-1}) generates new candidates with the length (k) increased by 1. Each new candidate is the sum of two sets from R_{k-1}.
- Line 17: Check k-item candidates. If they are rare create new sequential patterns with negative conclusions.
- Line 19: $ABSENT_q$ contains all possible subsets of M_q. For $M_q=\{p_1,p_3,p_4\}$, $ABSENT_q=\{\{p_1\},\{p_3\},\{p_4\},\{p_1,p_3\},\{p_1,p_4\},\{p_3,p_4\},\{p_1,p_3,p_4\}\}$.
- Lines 20–24: Generate new patterns using elements X from $ABSENT_q$ (i.e. negative conclusions $\sim X$) and the regular pattern q considered. Add the patterns $s^-(q \rightarrow \sim X)$ obtained to the output S^-. Elements of $ABSENT_q$ do not occur in any source sequence t after the regular pattern q. For that reason, $sup^{s^-}(q \rightarrow \sim X)=sup_q$ and $con^{s^-}(q \rightarrow \sim X)=100\%$.
- Lines 25–29: For each subset X of $ABSENT_q$ and each verified candidate Y from any set R_i (see line 36) create a new pattern for their unions $X \cup Y$, i.e. $s^-(q \rightarrow \sim(X \cup Y))$. Since X does not occur in any t containing q ($\sim X$ "occurs" in all these t), the support

and confidence of each $s^-(q \rightarrow \sim(X \cup Y))$ are the same as for the appropriate $s^-(q \rightarrow \sim Y)$, i.e. $sup^{s^-}(q \rightarrow \sim(X \cup Y)) = sup^{s^-}(q \rightarrow \sim Y)$ and $con^{s^-}(q \rightarrow \sim(X \cup Y)) = con^{s^-}(q \rightarrow \sim Y)$.

- Lines 30–40: One run of the loop checks the frequency of each candidate $X \in CAND_k$. If X is rare enough ($\sim Y$ is frequent enough) then a new pattern $s^-(q \rightarrow \sim X)$ is created.
- Line 31: At first, we assume that X does not occur after q in any source sequence. Hence, the initial $sup^{s^-}(q \rightarrow \sim X) = sup_q$ is the number of sequences $t \in T$ that contain q. In other words, X is maximally rare ($\sim X$ is maximally frequent). This also means that the initial $con^{s^-}(q \rightarrow \sim X)$ is 100%.
- Lines 32–34: Calculate support of candidate X using all maximum complements that contain q and are stored in CDB_q (see lines 5–9).
- Lines 33–34: If the intersection of X and maximum complement C^{max} is non-empty (at least one X's item belongs to C^{max}) then one source sequence does not support conclusion $\sim X$. It means that X is less rare and $\sim X$ is less frequent. Thus, initial support $sup^{s^-}(q \rightarrow \sim X)$ has to be decremented. The processing could quit the *for each* loop when $sup^{s^-}(q \rightarrow \sim X)$ falls below $threshold_q$.
- Lines 35–40: If X is rare enough ($\sim X$ is frequent enough) then add a new sequential pattern $s^-(q \rightarrow \sim X)$ to the output set S^-.
- Line 36: X is k-item set. Since X is rare enough $\Leftrightarrow$ $\sim X$ is frequent enough X can be used to generate new $(k+1)$-item candidates by means of R_k.
- Line 37: Add a new sequential pattern with negative conclusion $s^-(q \rightarrow \sim X)$ to the output collection S^-.
- Line 38: $sup^{s^-}(q \rightarrow \sim X)$ is expressed as the number of source sequences.
- Line 39: Confidence $con^{s^-}(q \rightarrow \sim X)$ can be calculated using Eq. (5.4).

Procedure *check_and_add* tests the input candidates and returns only the rare ones, i.e. frequent negative. In the typical algorithms for association rule mining, the frequent itemsets are extracted whereas in the SPAWN algorithm, only the rare candidates X from $CAND_k$ are selected (line 35), used to create new patterns (lines 37–39) and to generate new candidates (added to R_k, line 36). The more often X occurs in CDB_q, the worse. The occurrence means the non-empty intersection with maximum complements stored in CDB_q (line 33). For the negative conclusion the lower the frequency of the candidate X, the better.

The final support $sup^{s^-}(q \rightarrow \sim X)$ is expressed as the number of sequences divided by $card(T)$ and it can obviously be converted to percentage value. Confidence values $con^{s^-}(q \rightarrow \sim X)$ are percentages.

5.4. Experiments

The experiments have been performed on web logs collected by the main web server of the Wrocław University of Technology. They contained 1000 user ses-

sions (multiset T) with total 1421 HTTP requests for 67 distinct web pages (set D). 257 regular sequential patterns q of the length of up to 7 pages were discovered using $minsup$=0.2% (2 sequences). Next, sequential patterns with negative conclusions were extracted using the SPAWN algorithm, see Sec. 5.3.2.

The number of sequential patterns with negative conclusions is the highest for conclusions of the length 11 (2,835,457 patterns) and 10 (2,834,) whereas the least quantity of patterns is for 22-page conclusions (only 2 patterns), 21-page (46), 20-page (506) and 1-page conclusions (536), Fig. 5.2. The number of patterns strongly depends on minimum support $minsup^{s-}$ (Fig. 5.3) and less on minimum confidence $mincon^{s-}$ (Fig. 5.4).

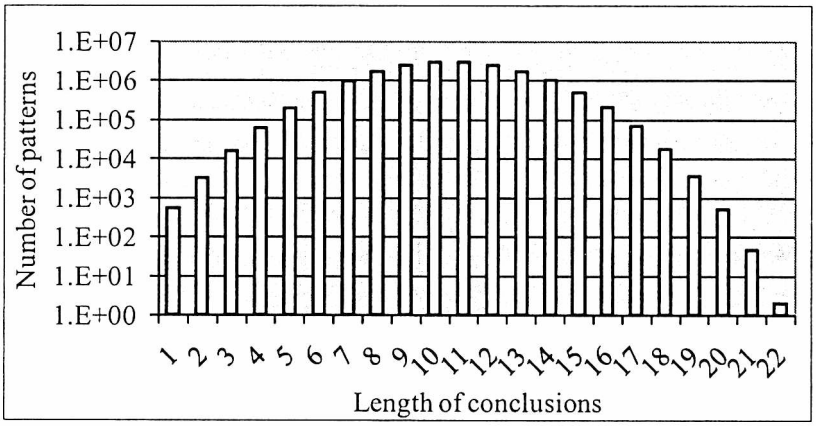

Fig. 5.2. No. of patterns with negative conclusions, $minsup^{s-}$=0.2%, $mincon^{s-}$=90%

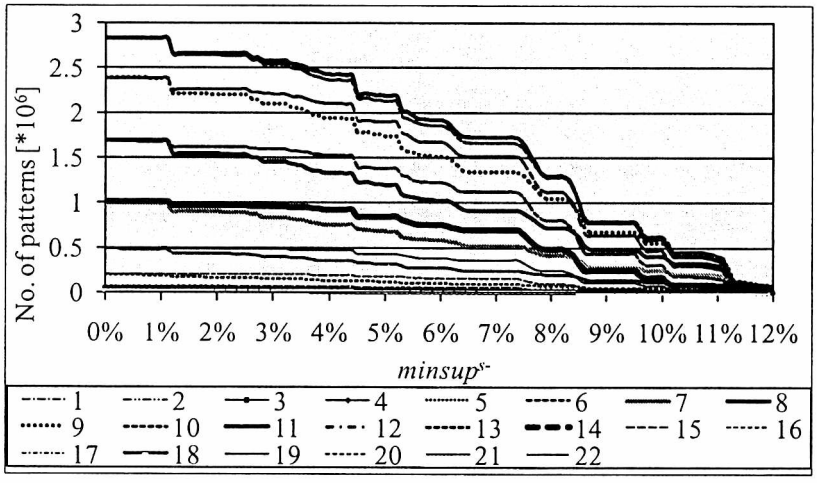

Fig. 5.3. No. of patterns in relation to $minsup^{s-}$ for different lengths of conclusions

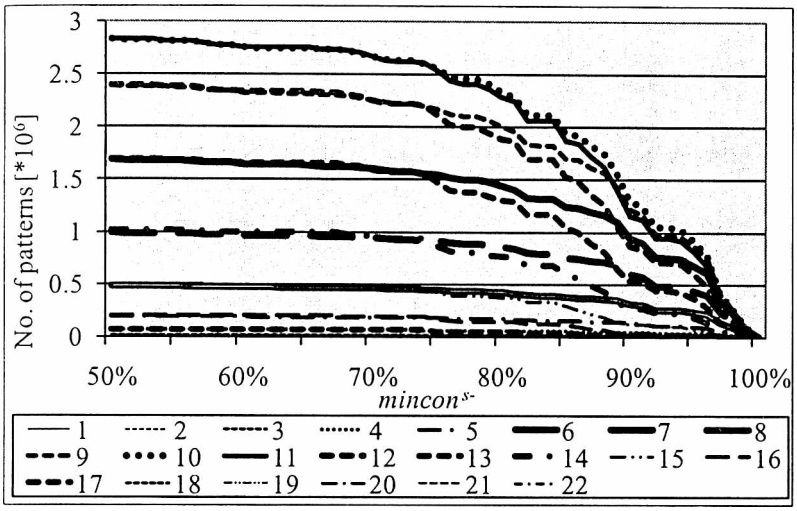

Fig. 5.4. No. of patterns in relation to $mincon^{s-}$ for different lengths of conclusions

The usability of sequential patterns with negative conclusions depends on their interpretations and often requires some kind of filtering.

After such manual scanning process, some interesting patterns were identified in the output set. Users who read information about student hostels and next about working possibilities (sequence q) do not visit Socrates/Erasmus Programme ($\sim X$); $sup^{s-}(q \rightarrow \sim X)=1.1\%$, $con^{s-}(q \rightarrow \sim X)=91\%$. In other words, those who consider working on the spot, are not likely to study abroad. Based on another pattern, users read general promotion information in English, next about study fees in English (q), however, they do not navigate to any contact information in English ($\sim X$); $sup^{s-}(q \rightarrow \sim X)=1.3\%$, $con^{s-}(q \rightarrow \sim X)=92\%$.

5.5. Negative Patterns in Verification of Web Hyperlinks

5.5.1. General Concept

The usage data - log files are processed to obtain sessions (unordered set of visited pages) and navigational paths, Fig. 5.5. Next, the positive and negative association rules are extracted from sessions, see Sec. 4. Simultaneously, paths are used to discover positive sequential patterns and the corresponding negative patterns - sequential patterns with negative conclusions, Definition 5.6. Since all the patterns operate on sets of items, they need to be aggregated to provide single values for pairs of pages, see Sec. 4.4.2 and 5.5.2. These aggregated values are used to verify hyperlinks extracted from the content of the web site in either posi-

tive or negative way. Positive verification combines positive association rules and sequential patterns whereas negative association rules and sequential patterns with negative conclusions are used to discover useless hyperlinks that can later be removed by content managers, see also Sec. 4.4.

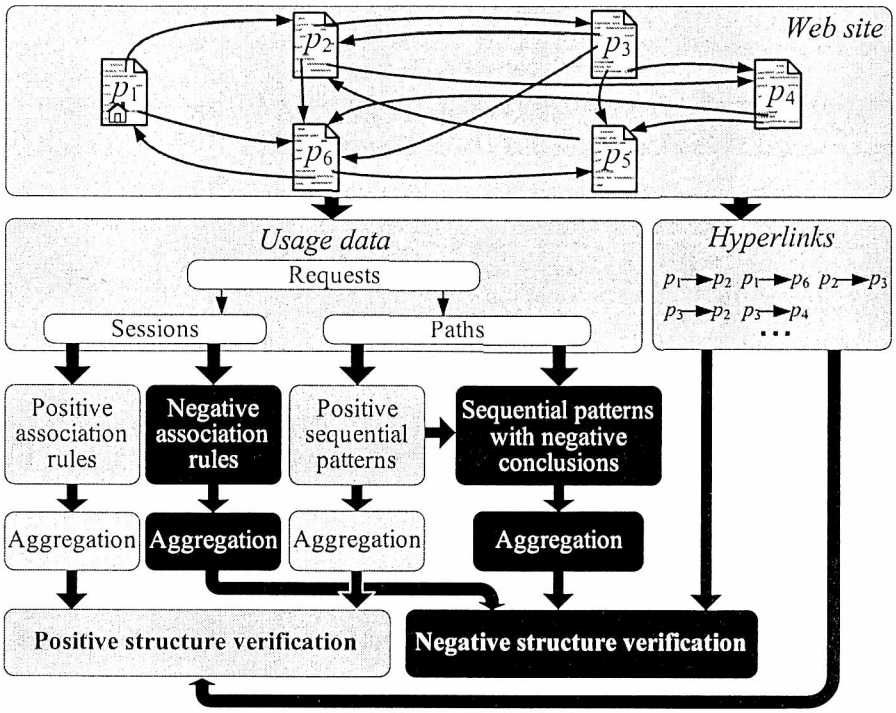

Fig. 5.5. Positive and negative verification of the user interface structure based on usage patterns

5.5.2. Aggregation of Sequential Patterns

Similarly to association rules, Eq. (4.5) and (4.6), we can calculate Positive Recommendation function for positive sequential patterns $PR^{seq}(p_i,p_j)$, for each pair of web pages p_i and p_j, using support of sequences $sup(q)$, Eq. (5.1):

$$PR^{seq}(p_i,p_j)=\frac{\sum\{sup(q):<p_i,p_j> \text{ is the subsequence of } q\}}{card(\{q:<p_i,p_j> \text{ is the subsequence of } q\})}. \tag{5.5}$$

Separately, Negative Recommendation function $NR^{seq}(p_i,p_j)$ is evaluated for sequential patterns with the negative conclusions based on the confidence values $con^{s}(q\rightarrow\sim X)$, Eq. (5.3):

$$NR^{seq}(p_i,p_j)=\frac{\sum\{con^{s^-}(q\rightarrow\sim X):<p_i>\text{ is the subsequence of } q, p_j\in X\}}{card(\{s^-(q\rightarrow\sim X):<p_i>\text{ is the subsequence of } q, p_j\in X\})}. \quad (5.6)$$

5.5.3. Comprehensive Verification Function

Aggregated confidences of positive association rules $PR(p_i,p_j)$, Eq. (4.5), and aggregated support of positive sequential patterns $PR^{seq}(p_i,p_j)$, Eq. (5.5), are utilized to evaluate positive verification function $verif^+(p_i,p_j)$ that corresponds to the connection from page p_i to p_j:

$$verif^+(p_i,p_j)=\alpha*PR(p_i,p_j)+\beta*PR^{seq}(p_i,p_j), \quad (5.7)$$

where α and β are constants that help to balance the influence of association rules and sequential patterns.

Similarly, the negative verification function $verif^-(p_i,p_j)$ is calculated based on average negative confidences of association rules $NR(p_i,p_j)$, Eq. (4.6) and sequential patterns with negative conclusions $NR^{seq}(p_i,p_j)$, Eq. (5.6) separately for each connection from p_i to p_j:

$$verif^-(p_i,p_j)=\gamma*NR(p_i,p_j)+\delta*NR^{seq}(p_i,p_j) \quad (5.8)$$

where γ and δ are adjustment constants.

Based on the values of $verif^+(p_i,p_j)$ and $verif^-(p_i,p_j)$ we are able to verify usefulness of links between pages. The high value of $verif^+(p_i,p_j)$ positively supports the existence of the hyperlink from p_i to p_j. Moreover, due to association rule contribution, the verification can even suggest new connections between pages.

On the other hand, a significant value of $verif^-(p_i,p_j)$ can be an important sign for removal of the hyperlink from p_i to p_j. Since all components of verification functions are calculated from the usage data, i.e. http requests (navigational sessions and paths) then the entire verification process is based on historical user behaviours.

Note that the above concept is the extension of the ideas presented in Sec. 4.4. However, both $verif^+(p_i,p_j)$ and $verif^-(p_i,p_j)$ reflect user behaviours extracted from web logs in the more comprehensive way.

5.6. Filtering of Recommendation Lists Based on Both Positive and Negative Patterns

5.6.1. Recommendation Lists Based on Web Content Mining

Since content-based recommender systems usually operate on text-based items, e.g. textual web pages, the content in such systems is normally described with

descriptors, i.e. terms which are expected to be the most informative and distinctive. One of the best known measures for descriptor selection, based on term weights, is the *term frequency - inverse document frequency* (*tf-idf*) measure [Kaz04d, Sal89]. Terms, which occur relatively frequently in one document (*tf*), but rarely in the rest of the set (*idf*), are more likely to be relevant to the topic of the document. Thus, *tf-idf* measure is based on the weight w_{ki} of the term t_k in the page (document) p_i, as follows:

$$w_{ki} = tf_{ki} \times idf_k = tf_{ki} \times \log(N/N^k), \tag{5.9}$$

where tf_{ki} - term frequency, i.e. the number of the term t_k's occurrences in p_i; N - the number of all pages in the web site; N^k - the number of pages in which term t_k occurs.

Terms that appear on many pages are not useful in distinguishing between a relevant page and irrelevant ones. The inverted document frequency idf_k reduces the influence of these terms. Moreover, terms t_k with too low or too high idf_k, as the bad content descriptors, can be excluded from further processing [Kaz04d]. In another approach to w_{ki} estimation, the terms that occur in some specific parts of the HTML content like title, description and keywords are reinforced, see Eq. (6.1) and Sec. 6.5.1.

A content-based system recommends pages similar to the just viewed one - this is kind of item-to-item correlation [Bil00, Moo00]. Documents (pages) are usually defined as vectors, e.g. the page p_i is represented by the M-dimensional vector $p_i=<w_{1i}, w_{2i}, \ldots, w_{Mi}>$, where M is the number of all terms not excluded from the web site content. The similarity $sim(p_i,p_j)$ between the page p_i and p_j can be calculated using the cosine function:

$$sim(p_i,p_j) = \cos(p_i,p_j) = \frac{\sum_{k=1}^{M} w_{ki} * w_{kj}}{\sqrt{\sum_{k=1}^{M} (w_{ki})^2 * \sum_{k=1}^{M} (w_{kj})^2}}, \tag{5.10}$$

or the formula usually known as Jaccard coefficient, see also Eq. (4.7) and (6.3):

$$sim(p_i,p_j) = \frac{\sum_{k=1}^{M} w_{ki} * w_{kj}}{\sum_{k=1}^{M} (w_{ki})^2 + \sum_{k=1}^{M} (w_{kj})^2 - \sum_{k=1}^{M} w_{ki} * w_{kj}}. \tag{5.11}$$

Based on the similarity measure, the system creates a recommendation list L_i for each web page p_i in the site. The list L_i consists of pages p_j for which similarity $sim(p_i,p_j)$ exceeds the given lower threshold λ^{low} and simultaneously is less than the upper threshold λ^{upper}: $\lambda^{low}<sim(p_i,p_j)<\lambda^{upper}$. The application of the upper limit prevents recommendation of nearly the same pages, i.e. with too high a similarity [Bil00].

5.6.2. Verification of Recommendation Lists Based on Usage Patterns

Both positive and negative usage patterns (association rules and sequential patterns) deliver valuable, user-oriented information about relation between web pages in the web site. The former confirm and strengthen the position of particular items in the ranking lists whereas the latter show the lack of connection between pages. This regards separately association rules and sequential patterns and is expressed by recommendation functions:

1. Positive Recommendation function for positive, regular association rules $PR(p_i,p_j)$, Eq. (4.5);
2. Negative Recommendation function for negative association rules $NR(p_i,p_j)$, Eq. (4.6);
3. Positive Recommendation function for positive sequential patterns $PR^{seq}(p_i,p_j)$, Eq. (5.5);
4. Negative Recommendation function for sequential patterns with the negative conclusions $NR^{seq}(p_i,p_j)$, Eq. (5.6).

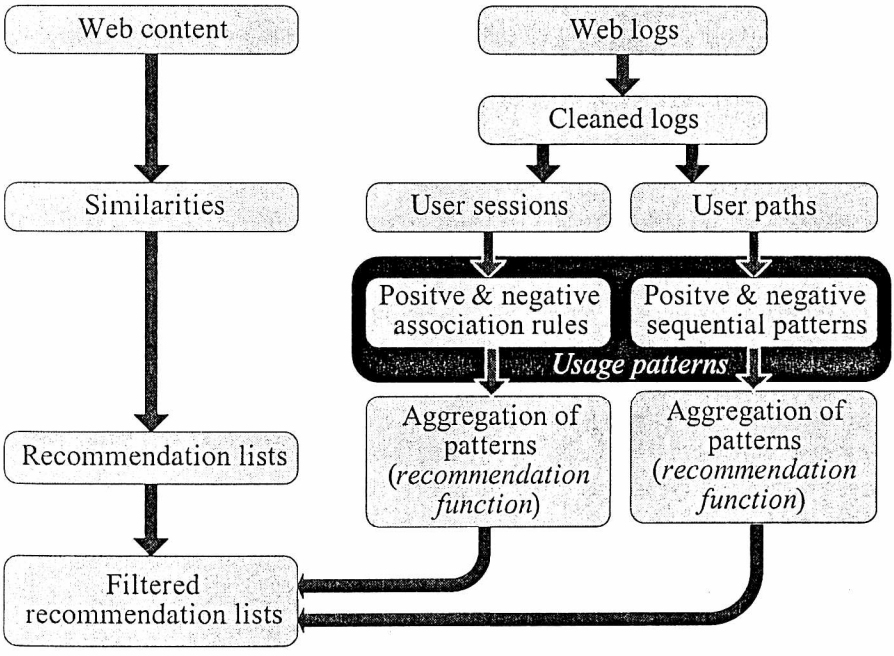

Fig. 5.6. Positive and negative verification of the user interface structure based on usage patterns

Based on these values, we can recalculate the closeness between web pages and filter recommendation lists created upon content similarity, Eq. (5.10) or

(5.11), Fig. 5.6. In other words, usage patterns that reflect user behaviour and are extracted from web logs can verify content-based suggestions. As a result, the final ranking function $rank(p_i \rightarrow p_j)$ is proposed:

$$rank(p_i \rightarrow p_j) = \eta \cdot sim(p_i, p_j) + \chi \cdot [PR(p_i, p_j) - NR(p_i, p_j) + PR^{seq}(p_i, p_j) - NR^{seq}(p_i, p_j)], \quad (5.12)$$

where η and χ are parameters that enable adjustment of the influence of content similarity and usage component, respectively; $\eta, \chi \in [0,1]$.

The verification involves all usage patterns considered, i.e. positive and negative association rules, positive sequential patterns as well as sequential patterns with negative conclusions. Since positive and negative association rules and in consequence Positive and Negative Recommendation functions for the given pair p_i, p_j are mutually exclusive (see Sec. 4.4.2) and the same case is valid for sequential patterns then at most two usage components can be greater than 0 in Eq. (5.12). Since all of them belong to the range [0,1], then $rank(p_i \rightarrow p_j) \in [-2 \cdot \chi, \eta + 2 \cdot \chi]$.

Note that the positive usage patterns, i.e. both association rules and sequential patterns, reinforce content similarities whereas the negative ones reduce the position of individual web pages in the ranking list.

Finally, for the given page p_i, a fixed number N^{disp} of target pages p_j with the greatest value of ranking function $rank(p_i \rightarrow p_j)$ form the ultimate, filtered recommendation list and these top N^{disp} web pages are suggested to the user upon visiting the page p_i.

5.7. Conclusions

Sequential patterns with negative conclusions are the new patterns that describe frequent sequences not followed by some items. In the web environment, these patterns can reflect typical user behaviours. For example, e-commerce users put some products into their basket, afterwards start filling order form but rather do not finish their purchases, i.e. skip all payment pages.

To extract sequential patterns with negative conclusions the SPAWN algorithm can be used, see Sec. 5.3.2. Note that sequential patterns with negative conclusions differ both from typical sequential patterns (the new patterns have negative conclusions) and from negative association rules (the left-hand side of the new patterns is a sequence instead of a set). Positive and negative association rules (see Sec. 4) extracted from web log data reflect typical usage patterns but they do not respect the navigational order whereas sequential patterns strongly depend on the sequence of navigation. Association rules and sequential patterns complement one another; by making use of the aggregated versions of them (see Sec. 4.4.2 and 5.5.2) we obtain a comprehensive and compressed view onto how users utilize the structure of the web site, i.e. connections between pages.

Due to the general profile of sequential patterns with negative conclusions, there is usually large amount of them. Hence, to make use of them it may be necessary to apply some filtering mechanisms in many application domains.

The new patterns, similarly to negative association rules, can be utilized to assess the previously existing connections between objects like hyperlinks binding web pages, see Sec. 5.5. This is especially useful for web content managers who can remove useless hyperlinks based on the knowledge provided by the negative patterns. As a result, the structure of the web site user interface can be simplified and adapted to user preferences reflected by their behaviour, i.e. patterns extracted from web logs. In this case, the new patterns are filtered according the set of hyperlinks.

Besides, the new patterns can extend typical content-based recommendation systems by verification of similarities calculated by means of textual content analysis, see Sec. 5.6.

6 Personalized Associations in Web Advertising

Adaptation to individual preferences of users - personalization - is an important challenge for the development of electronic commerce. Jeff Bezos, CEO of Amazon.com, expressed it as follows: "If I have 3 million customers on the web, I should have 3 million stores on the web" [Sch01]. Moreover, 80% of Internet users in 2005 were interested in receiving personalized content on sites they visited [Cho05]. As a result, advertisements often placed on web pages as vital and profitable parts of presented content should also be considered for personalization. However, the current demographic targeting of freely available web content - popular in web advertising - appears to be insufficient in an age of disappearing borders and mixed societies. Since the market consists of human beings, not demographic objects, web personalization should depend on an individual's behaviour rather than on stereotypes created according to their geographical location or other demographic features such as gender or age. Traditional advertising presents the same offers for everyone, but it does not meet the current requirements of businesses. If we want to increase effectiveness; the right person should receive the right message at the right time and in the right context [Ada04].

The AdROSA system described in this section for automatic web banner personalization tries to find associations between contents viewed by the current user and advertisers' contents pointed by the banners. It integrates web usage and content mining techniques to reduce user input and to respect users' privacy. Furthermore, certain advertising policies, important factors for both publishers and advertisers, are taken into consideration. The integration of all the relevant information is accomplished in one vector space to enable online and fully personalized advertising.

An example of a typical web advertisement takes the form of banners - rectangular images placed on web pages (Fig. 6.1) or other graphical elements displayed in a new layer or new window of the browser. There are also many other forms of online advertisements, e.g. sponsored hyperlinks and articles, or mail-outs, but in this section, we will concentrate on banners and similar forms.

A banner ordinarily includes the company name, the product name, and/or short message from the advertiser to the potential customer. Its goal is to encourage visitors to click on the image for more detailed information. There are two main participants in web advertising: an advertiser and a publisher. The former would like to attract as many users as possible to visit its web sites using advertisements displayed on the web pages of the latter (Fig. 6.1). The advertiser is charged for placing banners on the sites of the publisher for whom these fees may be the major or even sole source of income.

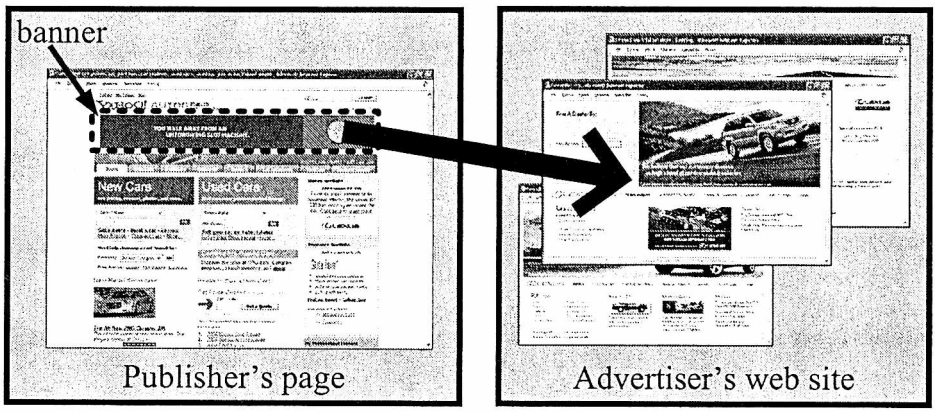

Fig. 6.1. Association between the publisher's page and the advertiser's web site

Users are showered with hundreds of advertisements, and they often pay little attention to banners appearing on a web page as bitmap images or animations. This seems to be the main problem of web advertising. The solution is to increase the correspondence between user interests and the subject of the displayed advertisement [Bau97], i.e. creation of personalized association between the viewed content and the advertising content.

This section has been prepared based on [Kaz08f].

6.1. Background

Two significant research domains may be distinguished within Internet advertising: scheduling and personalization. The main goal of scheduling is to maximize the total click-through rate for all advertisements by appropriately managing display time and advertising space on the web page. The problem is NP-hard and can be solved using linear programming [Abe99, Chi03, Nak02], extended with statistically derived entropy maximization [Tom00], Lagrangean decomposition [Ami03], or some other approximation algorithms [Daw03].

Personalization seems to be an important and difficult challenge for current advertisers and is more "individualized" than target advertising, which simply divides customers in a market into specific segments [Iye05, Yan06, Zho04]. It aims to assign a suitable advertisement to a single web user rather than to a group of individuals. To achieve this goal, personalization systems need to have some information about the user. Many web portals create user profiles using information gained during the registration process or ask the user to answer some questions about their preferences. However, this requires a lot of time and effort and can discourage many users. Besides, users tend to give incorrect data when there are concerns about their privacy [Mon03]. Even reliable data becomes out of date with the evolution of online customers' interests. An alternative solution is to exploit information stored in the web server logs. With regard to privacy fears, this method is safe and may also be useful for news portals or web sites where users do not need to log in to use the service [Bae03]. Another approach to advertisement personalization involves identifying short- and long-term user interests [Lan99]. Short-term interests are derived from keywords submitted by a user during a search. However, such keywords may often have nothing in common with the user's regular preferences. Long-term interests are taken from user profiles, which are completed by users and stored in the system database. However, advertising personalization was performed using only short-term information.

A system based on web usage mining, i.e. the clustering of navigation paths to create usage patterns, was presented in Bae et al. [Bae03]. Experts manually classify pages from both the publisher's web site and the target sites of the advertisements into thematic categories. The assignment of appropriate advertisements to each active user is accomplished according to pages (categories) visited by the given user during the current session. This matching is based on fuzzy rules stored in the system. The fuzzy approach was also used in target advertising based on user profiles [Yag00].

Many personalized advertising methods were proposed that make use of explicit user profiles, which are gathered, maintained, and analyzed by the system. Such methods often make use of data mining techniques [Lai00, Per02].

Apart from personalization, the problem of advertising placement has also been considered. There are two main approaches to the management of online advertising placement: categorical and site-based. In the first, ROC (run of category), a banner appears randomly on any page within a thematic category of a web site or ad network. Hence, each advertisement is assigned by the advertising manager to one or more categories, and each web page simultaneously belongs to the individual category so that the banner is presented only on matching pages. In the second approach, ROS (run of site), an ad placement is extended to the whole single web site. An advertisement is assigned to a web site rather than to its subcategory of subject matter. Ngai suggested a multicriteria approach

(AHP) for the selection of web sites for online advertising campaigns. He considered five static features: the impression rate depending on the traffic at the site, the monthly rental cost, the match of the audience in terms of age and education level, the general content quality, and the subjective "look and feel" rate [Nga03].

A commercial online advertising system, AdSense, is provided by Google Inc. [Goo08]. It delivers a targeted advertisement to the web site of a publisher and consists of two options: "AdSense for content" and "AdSense for search." The former delivers text or image advertisements based on the content of a publisher's site. Advertisements appropriate to the analyzed content of the site are displayed for the user in the "Ads by Google" page frame built into the publisher's site. The publisher's site content is periodically analyzed by Google's search engine to update the assignment of appropriate advertisements to the publisher. The latter encourages publishers to add the Google search box to their pages. Each time a user makes use of this box and searches the publisher's site or the web, some targeted text-based advertisements are attached to the search result pages in the form of "sponsored links". Google pays the publisher for each click on an advertisement delivered by AdSense. The complementary Google program, AdWords, is targeted to advertisers, who define and deliver to Google keywords associated with their advertisement. This helps Google match the available advertisements with all activities in which given keywords occur, which is accomplished by Google monitoring the use of the search engine and the navigation of publishers' sites involved in AdSense. Google also developed for its AdWords a matching algorithm called BALANCE that enables daily revenue of the entire system to be maximized [Meh05, Meh07]. It respects limits on daily budgets of individual advertisers as well as separate bids=revenue specified by the advertiser for the keyword. Thus, a bid is the price that the given advertiser is ready to pay for the exposition of its ad while searching the given keyword. Similarly, Rusmevichientong and Williamson studied algorithms for the selection of profitable search keywords that are especially useful for fixed advertising budgets [Rus06]. Since the AdSense/AdWords system can access only data available for the Google search engine and the content of web sites, it is able to provide only "ephemeral personalization" of advertising. The ephemeral approach can deliver a different item on every page of a web site but be the same for all users [Sch01]. The much more adaptive method – "persistent personalization" – uses the history of a user behaviour and generates a different item for each user in each context [Sch01].

Another example of the ephemeral personalization approach was presented by Yih *et al.* [Yih06]. The authors focused on the discovery of terms that would represent the content of web pages; these would then be used to match them with advertised web sites. An additional learning mechanism was introduced to enable better term selection for new textual resources.

6.2. Advertising Models

Three main web advertising models can be distinguished depending on who leads the advertising management process: broker, portal, and advertiser model [Bil03]. In the *broker model* (Fig. 6.2), there is an advertising broker-mediator that connects many publishers with many advertisers. The broker hosts and manages all advertisements and supplies them to the publisher or directly to the end user online. Also, the broker usually provides some, more or less, advanced targeting criteria such as the selected publisher's site (or its thematic section), geographical location, or other user demographic data (age, gender). For both advertisers and publishers, the broker appears as an advertising agency and takes some profit from the campaign money. Based on this model, several advertising networks were developed in 1995 by Real Media [Rea07] and DoubleClick [Dou04], which have recently become powerful players in the world advertising market.

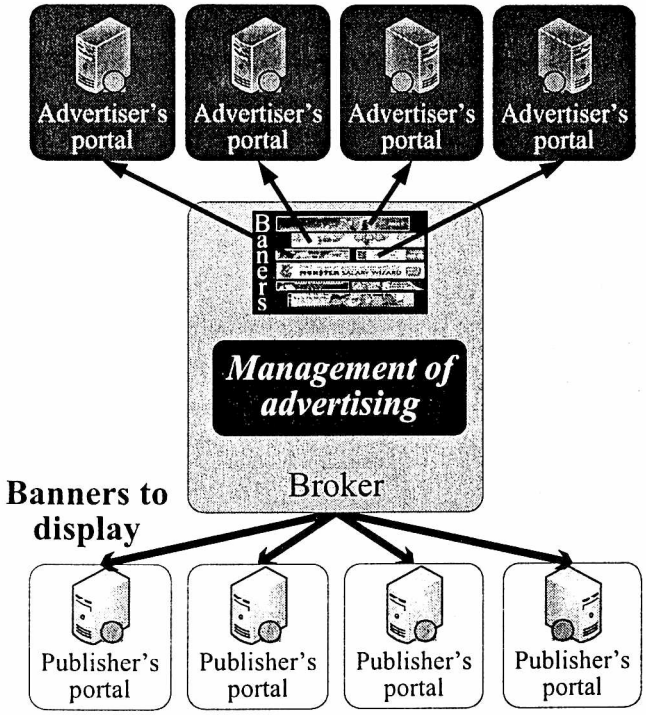

Fig. 6.2. Broker model of advertising

In the *portal (publisher) model,* the publisher itself is responsible for advertisement management and cooperates with many advertisers (Fig. 6.3). This is a model used by publishers that are large enough to offer advertising services. Additionally, por-

tals using this model can take advantage of gathered user profiles, exploiting the data in any advanced personalization system such as adaptive link [Kaz04d], product recommendation [Kaz04b], or e-marketing [Per02].

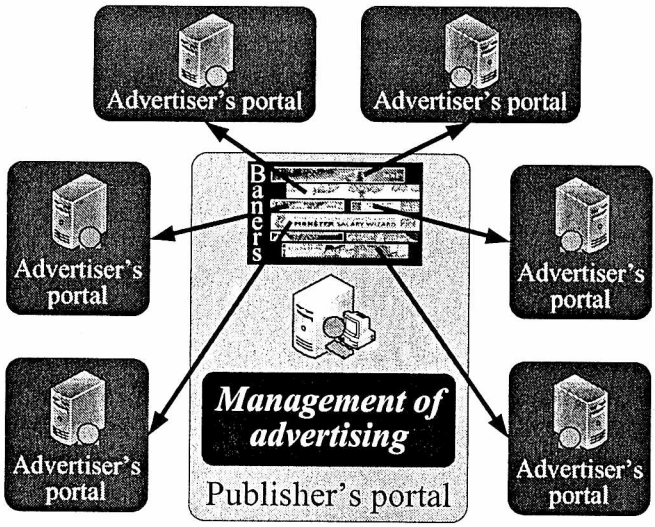

Fig. 6.3. Portal (publisher) model of advertising

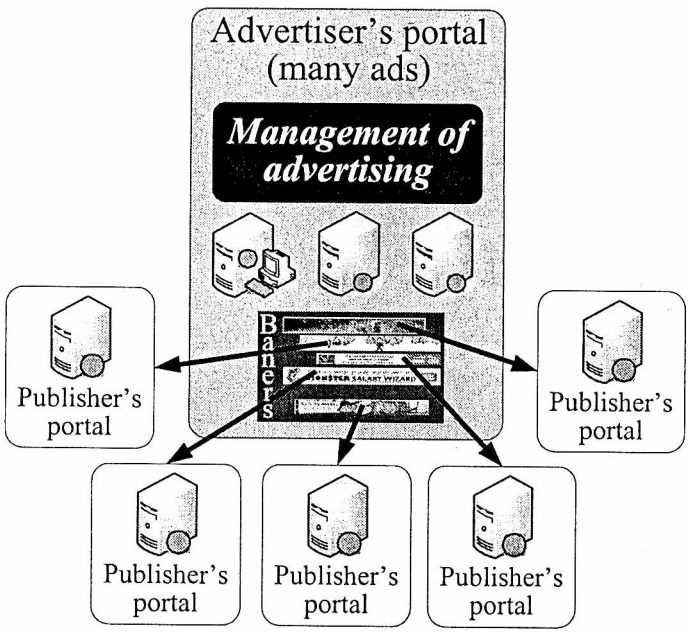

Fig. 6.4. Advertiser's model of advertising

The third, *advertiser model*, may be used by large online stores, which are able to advertise themselves directly to their customers (Fig. 6.4). In this case, the advertiser itself is responsible for the management of advertising and for banner distribution among particular pages of selected publishers. Moreover, the advertiser possesses the information on which page and at which publisher its banner was clicked. This model of advertising is used rather rarely and, most often, within a group of closely related companies (e.g. inside holdings).

The AdROSA system described below is designed to be used in the portal model of advertising. However, it could be easily introduced into the broker model by treating the set of publishers' portals as one coherent publishing space, see Sec. 6.7.1.

6.3. Advertisement Features

Nowadays, most online advertising systems use the principle of *customer-based targeting*. Each user is identified and classified according to the user's geographical location (IP address) and browser settings sent with the HTTP request, navigation habits, and user profiles (preferences) completed by the user during the registration process. This data is used to personalize displayed banner advertisements [Agg98, Lan99].

Analyzing the advertising offers of the largest Polish portals (*www.wp.pl, www.onet.pl*), several international ones, and reviews provided by ClickQuick.com [Aff03], we have observed many targeted criteria available to advertisers. Apart from the demographic data of a user (age, gender, location, etc.), advertisements can be targeted towards their education, profession, or interests. Furthermore, the publisher can choose the time of day of the emission, particular parts of the web page, and limit the number of emissions for a single user. An advertiser is usually charged based on cost per month per one thousand emissions of advertisement (CPM - cost per mille) [McC98]. However, several other payment models exist: CPA – cost per action, CPS - cost per sale, CPC - cost per click, and CPI – cost per single impression. In the CPA model, an advertiser pays for every action of a visitor related to their advertisement. An action can be a sale, user registration, filling in a form, rating a product or a text, voting, establishing an account, or anything else defined by the advertiser. If such an action is a successful sale, then we have the CPS model. CPC is another online payment model, in which advertisers pay for each click made through on their advertisement. An important and commonly used measure of advertising is click-through rate (CTR), the ratio of the number of clicks to the emission number [Agg98]. The typical value of CTR ranged from 2% to 4% at the end of the 1990's [McC98]. Nevertheless, it should be mentioned that the average CTR is currently decreasing as a consequence of the increasing number of total

advertisements displayed [Rod04]. Afterwards, the average CTR decreased to below 1% [Cli03]. Some other, slightly more sophisticated, pricing models for web advertising were proposed in [Nov00].

The method presented below takes into consideration most of the contemporary, applied aspects of advertising campaigns.

6.4. The Concept of the AdROSA System

The proposed advertising method included in the AdROSA system (*Advertising Remote Open Site Agents*) solves the problem of automatic personalization of web banner advertisements with respect to user privacy (none of the user's personal details are stored in a database) and recent advertising policies. It is based on extracting knowledge from the web page content and historical user sessions as well as the current behaviour of the online user, using data mining techniques. The implementation of data mining to web content and web usage is usually called web content and web usage mining, respectively [Mob00b, Yao02]. There are also some integration methods of both these approaches [Kaz04d, Mob00b].

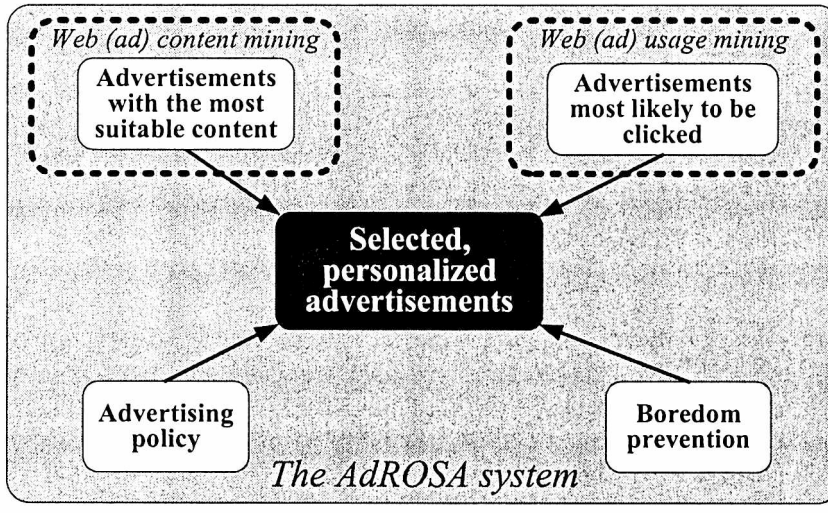

Fig. 6.5. Factors of advertisement selection in the AdROSA system

The proposed method uses both web mining techniques and combines in one personalized framework several useful factors of advertising: the most suitable content (the content of the advertiser's web site), click-through probability, advertising policy arising from contracts, and boredom prevention mechanisms (Fig. 6.5). The last factors determine a periodical rotation or scheduling of advertisements for the user.

Historical user sessions are stored in the form of vectors in the AdROSA database and they are clustered to obtain typical, aggregated user sessions (Fig. 6.6). The centroid of the cluster corresponds to one *usage pattern* of the publisher's web site. A usage pattern contains information about one typical navigational behaviour of similar users. Also, each user session is linked to the set of advertisements visited (clicked) by the user during this session. Having a cluster of sessions, the AdROSA system can also extract a cluster of visited advertisements, i.e. *ad visiting pattern*. Thus, one web usage pattern (centroid) corresponds to exactly one ad visiting pattern (centroid).

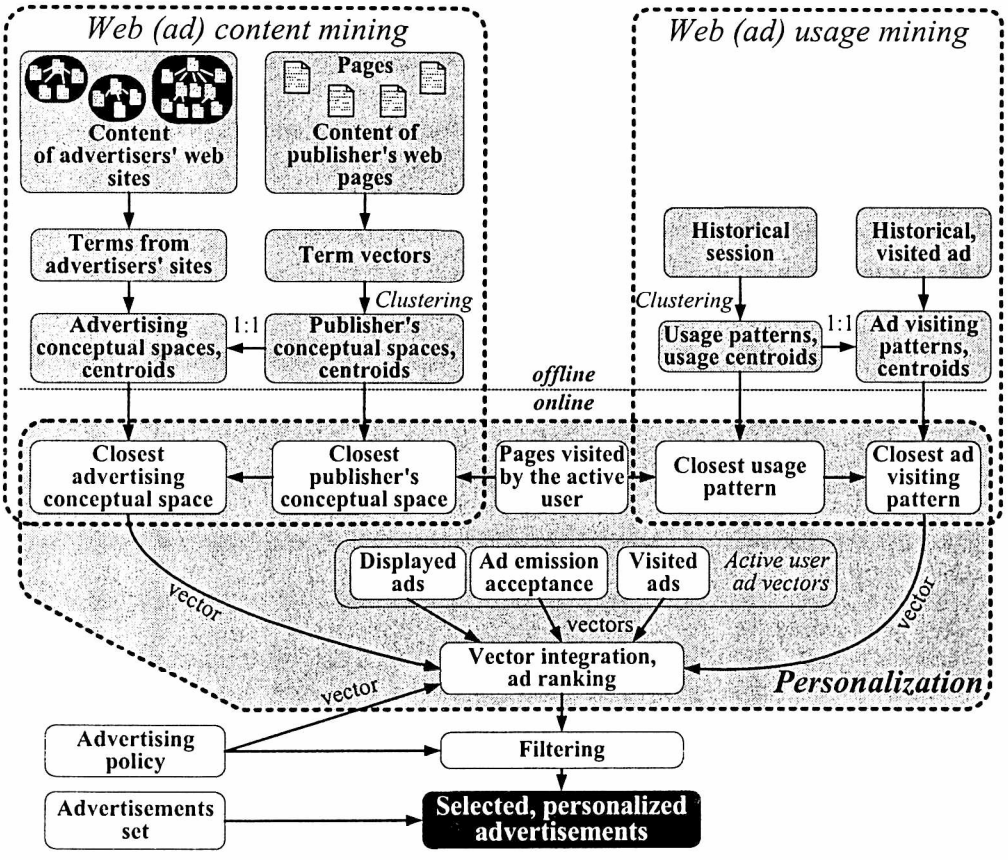

Fig. 6.6. Overview of the personalized advertising method in the AdROSA system

The site content of the publisher's web pages is automatically processed in a similar way. The system extracts terms from the HTML content of each page. Next, the most representative terms are clustered and, as a result, we obtain content thematic groups - *conceptual spaces*. Conceptual spaces denote separate sub-

jects existing in the publisher's portal like terms related to different domains that form separate conceptual spaces. For example, sports vocabulary is represented in the *sports conceptual space*, while terms related to travelling are included in the *travel conceptual space*. Note that each publisher's page can to a greater or lesser extent belong to each conceptual space. In other words, we can say that page *A* is mostly about *sports;* however, it is also a little bit about *travelling*.

To recommend a suitable advertisement for the user, we have to know its general subject matter. This is achieved by text (HTML) content analysis of the advertisement target web site. The AdROSA system automatically downloads advertiser's web pages and processes only the terms that occur in the publisher's web pages. As a result, we obtain *advertising conceptual spaces* corresponding to the appropriate *publisher's conceptual spaces*. For example, if terms relevant to travelling formed a distinct conceptual space, then publisher pages about tourism would probably be matched with the advertisements of travel agencies.

A user requesting a web page is assigned online to both the closest usage pattern and the closest conceptual space, based on the user's previous behaviour during the active session; the system retains and analyses pages recently visited by the active user. An assignment to the usage pattern helps to recognize what kind of behaviour the current user represents, whereas the closest conceptual space indicates the recent content-based interests of the user. For example, if the user navigates through pages about tourism, they would be assigned to the travel conceptual space. Advertisements, e.g. from travel agencies, which are relevant to the established, closest conceptual space, would be linked to web sites with appropriate content for the current user's interests, so the system would present advertisements of travelling agencies to the current user. In this way, we obtain content-based association between the recent user interest (expressed by navigational behaviour) and the advertisers' portals offers represented by advertisements.

Additionally, the closest usage pattern - and, consequently, the closest ad visiting pattern - enables the selection of advertisements that are most likely to be clicked by the current user. For example, if the user is behaving like a new user, they would be presented with banners which had been followed especially by new users. This means that the system favours advertisements that appeal to other users who behaved similarly to the current one.

Moreover, assignment to the closest conceptual space and usage pattern is performed at each user step, i.e. HTTP request. Hence, personalized associations are adapted to the continuously changing user needs. Finally, a user can be reassigned from one conceptual space or usage pattern to another. If the user moved from tourism pages to sports pages, they would be relatively quickly reallocated from the travel conceptual space to the sports conceptual space. The matter of reassignment is discussed further, in Sec. 6.8, in detail.

User behaviour and information about already displayed or visited banners are separately stored by the system for each active user in the form of appropriate vectors. These vectors are utilized to prevent overly frequent emissions of the same advertisement for one user and to provide control over the number of emissions to satisfy contractual obligations. This is performed in the vector integration stage.

The entire process of banner selection exploits not only the information mentioned above but also the targeting parameters established by the advertiser (*advertising policy*), such as limiting emissions per user during a single session. Another policy could be some additional priority features, which could be manually set up separately for each advertisement, see Sec. 6.5.4. This is an additional advantage for the publisher that provides an opportunity to increase the ranking positions for more profitable advertisements.

Finally, the personalized ranking list is filtered using additional advertising policy features like the limitation to certain web browsers, time of day of the emission, etc. As a result, the AdROSA system returns to the web server the list of n top ranked, filtered advertisements that are dynamically incorporated into the returned web page content. Note that, since the assignment, integration and filtering processes are launched at each user request, the system is able to adapt its advertisements to the variable interest of the current user.

The AdROSA system proposed here is a significantly modified and extended version of the general concept presented in [Kaz04c]. The new version is based on the new approach to content processing (Sec. 6.5.1), cluster representative calculation (Sec. 6.5.1 and 6.5.2), and a simplified method of user monitoring (Sec. 6.5.3). In addition, some new analyses were performed to study model limitations, extensions, potential problems, and correlation with other typical models (Sec. 6.7). New advertising metrics were proposed in Sec. 6.7.1. The novel multi-agent architecture and prototype implementation of the system with a working example are presented in Sec. 6.8 and 6.9. The AdROSA system can be incorporated into the ROSA project - a framework for hyperlink recommendation [Kaz03b, Kaz03c] and, as a result, we can obtain one homogeneous and adaptive web system with complex personalization.

6.5. Knowledge Processing in AdROSA

Four main data sources are used and processed by the AdROSA system: web content, usage data, active user behaviour, and policy data. Only behavioural data is processed online and related to an individual active user, whereas the processing of the other three data sources is performed offline and delivers knowledge common for all visitors.

The web content data is the content of both the publisher's pages and all advertisers' portals, see Sec. 6.5.1. The usage data is the set of historical user sessions together with information about advertisements clicked during these sessions, see Sec. 6.5.2. The current user behaviour consists of data about visited pages as well as the presented and clicked advertisements during the active session, see Sec. 6.5.3. The policy data contains some general features of particular advertisements that enable them to be tailored to advertising strategy, see Sec. 6.5.4.

Most activities of the AdROSA system are based on data that is retained in vector form. Here, we have a list of all utilized vectors:

- *term page vector* $\mathbf{tp}_j$ that includes information about the publisher's pages that contain term t_j; M-dimensional;
- *advertiser term vector* $\mathbf{ta}_j$ that includes information about advertisers' sites that contain term t_j; N-dimensional;
- *publisher's conceptual space* $\mathbf{ctp}_k$ – the mean vector from the vectors $\mathbf{tp}$ that belong to the cluster k; M-dimensional;
- *advertising conceptual space* $\mathbf{cta}_k$ – the mean vector from vectors $\mathbf{ta}$ that belong to the cluster k; $\mathbf{cta}_k$ corresponds to $\mathbf{ctp}_k$; N-dimensional;
- *session vector* $\mathbf{s}_j$ that contains information about pages visited during one historical user session; M-dimensional;
- *usage pattern* $\mathbf{cs}_k$ – the mean vector from the k'th cluster of sessions $\mathbf{s}$; it describes one typical user behaviour; M-dimensional;
- *visited ad vector* $\mathbf{v}_j$ – advertisements visited (clicked) by the user during historical session $\mathbf{s}_j$; N-dimensional;
- *ad visiting pattern* $\mathbf{cv}_k$ – the mean vector from the kth cluster of sessions $\mathbf{s}$; corresponding to $\mathbf{cs}_k$; it contains the advertisements most likely to be clicked within usage pattern $\mathbf{cs}_k$; N-dimensional;
- *active user page session vector* $\mathbf{ps}_j$ that contains data about pages visited during the jth active session; M-dimensional;
- *active user ad session vector* $\mathbf{as}_j$ that contains data about advertisements recently presented during the jth active session; N-dimensional;
- *ad emission vector* $\mathbf{e}_j$ that contains data about the number of times each advertisement was displayed during the jth active session; N-dimensional;
- *emission per user vector* $\mathbf{epu}$ that includes the number of permitted emissions for each advertisement, common for all users; N-dimensional;
- *ad emission acceptance* $\mathbf{eau}_j$ that denotes whether a certain advertisement can still be considered for presentation during the jth active session; N-dimensional;
- *ad priority vector* $\mathbf{p}$ that allows setting a priority value for each advertisement according to the advertising policy; N-dimensional.

Each of the above vectors belongs to one of two vector spaces: the M-dimensional space of all publisher's pages or the N-dimensional space of all advertisements (advertisers' portals).

6.5.1. Web Content Mining - Content Processing

The publisher's web content is processed using *crawler*, an agent that downloads and indexes the content of all pages from the web site. Terms obtained from the HTML content are filtered using several statistical features to extract the best descriptors, for instance, terms occurring too rarely and too frequently are excluded. These are terms no. 8 and 9 in Fig. 6.7. For each selected term t_j, an M-dimensional *term page vector* $\mathbf{tp}_j = < w_{j1}^{tp}, w_{j2}^{tp}, ..., w_{jM}^{tp} >$ is created, where M is the total number of web pages in the publisher's web site. The coordinate w_{ji}^{tp} denotes the weight of the term t_j in the document (page) d_i, according to Information Retrieval theory [Sal89] with respect to the web page-specific profile:

$$w_{ji}^{tp} = (tf_{ji}^{b} + \alpha\, tf_{ji}^{t} + \beta\, tf_{ji}^{d} + \gamma\, tf_{ji}^{k})^{*} \log\left(\frac{M}{n^{tj}}\right), \tag{6.1}$$

where tf_{ji}^{b}, tf_{ji}^{t}, tf_{ji}^{d}, tf_{ji}^{k} – term frequency (number of occurrences) of term t_j respectively, in the body, title, description, and keywords – HTML meta elements of the page d_i; α, β, γ are coefficients that increase the importance of the term that occurs in the selected parts of the HTML documents; and n^{tj} – the number of pages in which the term t_j occurs.

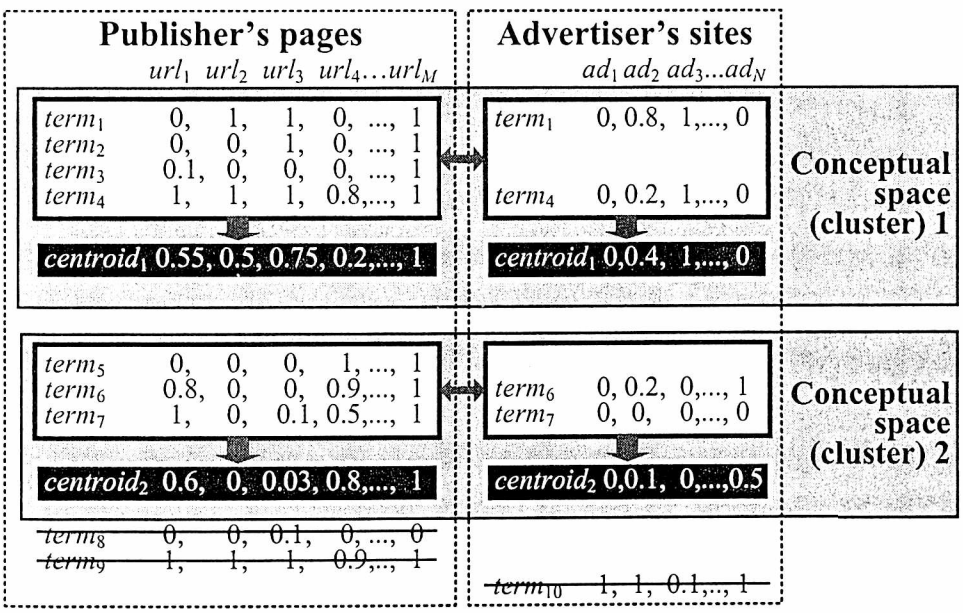

Fig. 6.7. Two conceptual spaces with component term vectors and derived centroids

Since some specific parts of the web page header, i.e. title, description, and keywords carry potentially more information in their individual terms than terms from regular sentences in the body, we can emphasize the former using α, β, and γ, respectively. Based on the experiments from [Kaz00], these coefficients can be set as follows: α=10, β=5, γ=5.

Note that the value of w_{ji}^{tp} may exceed 1 because both the term frequency components in Eq. (6.1), and especially common logarithm $\log\left(\frac{M}{n^{t_j}}\right)$, are often greater than 1. Since M is the total number of publisher's web pages and n^{t_j} - only the number of pages containing term t_j, the fraction $\frac{M}{n^{t_j}}$ often exceeds 10, in particular in the case of large web sites. As a result $\log\left(\frac{M}{n^{t_j}}\right)$ is greater than 1 for many terms t_j. To ensure that values of w_{ji}^{tp} belong to the range [0,1], normalization has been applied to all term page vectors:

$$w_{ji}'^{tp} = \frac{w_{ji}^{tp}}{max^t}, \tag{6.2}$$

where w_{ji}^{tp} - normalized value of w_{ji}^{tp}; max^t - the maximum value of w_{ji}^{tp} among all vectors.

The set of normalized $\boldsymbol{tp}_j$ vectors is clustered, using the group average link - a hierarchical agglomerative clustering method (HACM) - to discover groups of terms that are close to each other [Ras92, Tan06 - Chap. 8]. The similarity between two content vectors $sim(\boldsymbol{tp}_j,\boldsymbol{tp}_k)$, essential for clustering, is obtained from the formula known as the Jaccard coefficient Eq. (6.3), see also Eq. (5.11) and (4.7).

$$sim(\boldsymbol{tp}_j,\boldsymbol{tp}_k) = \frac{\sum_{i=1}^{M} w_{ji}'^{tp} * w_{ki}'^{tp}}{\sum_{i=1}^{M} (w_{ji}'^{tp})^2 + \sum_{i=1}^{M} (w_{ki}'^{tp})^2 - \sum_{i=1}^{M} w_{ji}'^{tp} * w_{ki}'^{tp}}. \tag{6.3}$$

Applying this method to a test web site with about 3,100 pages and 500 filtered descriptors, 41 clusters were obtained [Kaz04d]. The filtering removed terms that occurred either sporadically or too frequently.

Note that the similarity $sim(\boldsymbol{tp}_j,\boldsymbol{tp}_k)$ reflects the level of association between two vectors $\boldsymbol{tp}_j$ and $\boldsymbol{tp}_k$.

Terms from one cluster describe the publisher's conceptual spaces (thematic groups) existing within the publisher's web site (Fig. 6.7). Once we have clusters,

we can easily calculate the representation of the conceptual spaces–centroid (mean) vectors – as follows:

$$\boldsymbol{ctp}_k = \frac{1}{n_k} \sum_{l=1}^{n_k} \boldsymbol{tp}_{lk} \tag{6.4}$$

where $\boldsymbol{ctp}_k$ – the centroid of the kth cluster; $\boldsymbol{tp}_{lk}$ – the lth term page vector belonging to the kth cluster; n_k – the number of terms in the kth cluster.

The content of the target web site of an advertisement is similarly processed. A typical banner is linked to the main page of the target service (level 0), which often includes just a menu or is the redirection page devoid of significant textual content. For this reason, the AdROSA system usually analyzes all pages from the next level (level 1) – pages from the same domain linked from the level 0 page. If required, further levels of pages from the advertiser's site can be processed. The content of all chosen pages is concatenated and treated by the system as the single advertiser's content, corresponding to one publisher's page. In general, the number of processed levels is one of the system parameters that can be set by the AdROSA administrator, either separately for each advertiser's portal or as one common value for all.

For each term extracted from target pages, which simultaneously exists in the set of term page vectors, *the advertiser term vector* $\boldsymbol{ta}_j = < w_{j1}^{ta}, w_{j2}^{ta}, ..., w_{jN}^{ta} >$ is created; where N is the number of advertisements (advertisers' web sites). The coordinate w_{ji}^{ta} denotes the weight of the term t_j in the advertiser's web site (a_i) and is calculated using Eq. (6.1). Note that one advertiser term vector, $\boldsymbol{ta}_j$ corresponds to exactly one publisher's term page vector $\boldsymbol{tp}_j$. For this reason, terms from publisher's web pages that do not occur in any advertiser's web site have an empty vector, $\boldsymbol{ta}_j$ with all coordinates set to 0. Examples of such nonexistent terms are $term_2$, $term_3$, and $term_5$ in Fig. 6.7. There are also some terms from the advertiser's site which are passed over. They have no equivalent counterpart on the publisher's pages – $term_{10}$ in Fig. 6.7. Such an approach ensures a uniform term domain for both the publisher's and the advertiser's content.

Advertiser term vectors $\boldsymbol{ta}_j$ are not clustered because the equivalent publisher's term page vectors have already been clustered. Since one vector $\boldsymbol{ta}_j$ corresponds to one vector $\boldsymbol{tp}_j$, one publisher's conceptual space is equivalent to one advertising conceptual space. Only one mean vector, centroid $\boldsymbol{cta}_k$, for each kth advertising conceptual space is calculated.

It may happen that no term from the given kth publisher's conceptual space occurred in any advertiser site. In that case, all coordinates of the centroid $\boldsymbol{cta}_k$ would be equal to 0. There is no usefulness to such a conceptual space. If we assign the current user to this cluster, we will not have any advertisements to dis-

play. Nevertheless, the existence of empty advertising conceptual spaces may be helpful in the future, if we admit new banners to fill this gap. Assuming that new advertisements are inserted much more often than the changes in the content of the publisher's portal, then new advertiser site content may be usually added to ***cta*** vectors without rebuilding the ***ctp*** vectors. This simplifies processing. The only thing we need to do is to mark the ***ctp*** vectors as temporally inactive for advertising. In the case of relatively frequent modifications in the publisher's site, we have to periodically repeat all data processing and simply remove each conceptual space with the corresponding empty advertising conceptual space. However, conceptual spaces may be exploited at the same time by the hyperlink recommendation system (the ROSA core system) [Kaz03b, Kaz03c, Kaz04d] that blocks destruction of vectors not useful for advertising.

6.5.2. Usage Mining - Session and Clicked Advertisement Processing

The first step of usage mining is the acquisition of HTTP requests and the extraction of sessions. A user session is a series of pages requested by the user during one visit to a publisher's web site. Since web server logs do not provide an easy method for grouping these requests into sessions, each request coming to the web server should be captured and assigned to a particular session using a unique identifier passed to a client's browser. During the first user request in the session, the system assigns this ID either to the returned cookies or to the dynamically generated hyperlinks as an additional query part of the URLs. The client returns this identifier to the server at each user request. In this way, the system is able to both monitor current activities of the user and gather the entire user session after it has finished. The procedure for closing the session is launched after a certain amount of idle user time, e.g. 30 min. Note that the user identification is performed only once within each session. Even though the browser returns the old cookies at the next user visit, the system creates a new user session. See Sec. 6.7.2 for further discussion concerning the possible continuation of the previous session for the same user.

Each jth historical user session stored by the system is represented by the M-dimensional *session vector* $\boldsymbol{s}_j = < w^s_{j1}, w^s_{j2}, ..., w^s_{jM} >$; where $w^s_{ji} \in \{0,1\}$ denotes whether the ith page was visited (1) or not (0) during the jth session. The coordinate w^s_{ji} is set to 1 for the ith page, no matter how many times this page was requested by the user within the confines of the jth session. For this reason, there is no need to perform any normalization as in the case of content processing.

Historical session vectors $\boldsymbol{s}_j$ are clustered into K' separated usage clusters in the same way as term page vectors $\boldsymbol{tp}_j$, using hierarchical clustering. The centroid $\boldsymbol{cs}_{K'}$ of a cluster (usage pattern) describes one typical user's behaviour – the navigation path throughout the web site. For a web site with more than 7,700 sessions

(35,000 requests), 19 clusters were created [Kaz04d]. Note that the coordinates of the centroid $cs_{k'}$ belong to the range [0,1].

Data about visited (clicked) advertisements during the jth historical user session is stored by the system in *the visited ad vector* $\boldsymbol{v}_j = < w^v_{j1}, w^v_{j2}, ..., w^v_{jN} >$; w^v_{j1} is the number of click-throughs of the ith advertisement during the jth session. For each user session s_j, there exists exactly one corresponding visited ad vector v_j. Thus, having the k'th session cluster $cs_{k'}$, we also obtain the appropriate cluster of visited ad vectors without a clustering procedure - similar to the publisher's and the advertising conceptual spaces. For each k'th cluster, the centroid-ad visiting pattern $cv_{k'} = < w^{cv}_{k'1}, w^{cv}_{k'2}, ..., w^{cv}_{k'N} >$, $w^{cv}_{k'i} \in [0,1]$, is found:

$$\boldsymbol{cv}_{k'} = \frac{1}{n_{k'}} \sum_{l=1}^{n_{k'}} \boldsymbol{v}_{lk'}, \tag{6.5}$$

where $n_{k'}$ - the number of vectors in the k'th cluster.

The problem of empty usage and ad visiting patterns should be handled in the same way as empty conceptual spaces, see Sec. 6.5.1. For usage patterns corresponding to "zero", the corresponding ad visiting patterns have to be either deleted or temporarily deactivated.

Since in our approach, each user remains anonymous, the single historical session vector s_j as well as the associated visited ad vector v_j are related to the jth stay of an anonymous user in the publisher's portal. Hence, j is the index of visits rather than human beings.

6.5.3. Monitoring of Active User Behaviour

The behaviour of each active user visiting the publisher's web site is monitored from the beginning until the end of the user session. The AdROSA system keeps the information about pages visited by all active users. For the jth active user, the appropriate *active user page session vector* $\boldsymbol{ps}_j = < w^{ps}_{j1}, w^{ps}_{j2}, ..., w^{ps}_{jM} >$ is maintained; where $w^{ps}_{ji} \in [0,1]$ denotes the temporary weight (timeliness) of the ith page for jth active user. The vector coordinate is set to 1 for the just viewed page, but for all previously requested pages, the coordinates are decreased to emphasize recent user behaviour and place less importance on what came before.

Two approaches to active user page session vector updates have been considered: an exponential and linear function. In the exponential approach, the system retains in ps_j the data about all pages visited by the jth user during the active session. Decreasing the coordinate values is accomplished by multiplication by the constant, which is less than 1, and that is equivalent to the exponential function (Fig. 6.8):

$$w_{ji}^{ps} = \begin{cases} 1, & \text{when the page } d_i \text{ is just requested,} \\ \lambda * w_{ji}'^{ps}, & \text{when the page } d_i \text{ was visited during the } j\text{th active session,} \\ 0, & \text{when the page } d_i \text{ was not visited during the } j\text{th active session,} \end{cases} \quad (6.6)$$

where λ - the constant parameter for the interval [0,1], determined experimentally, in the implementation $\lambda = 0.75$ was assumed; $w_{ji}'^{ps}$ –the previous value of the coordinate.

Note that if page d_i is visited again, the value of w_{ji}^{ps} is set all over again to 1 at each such request.

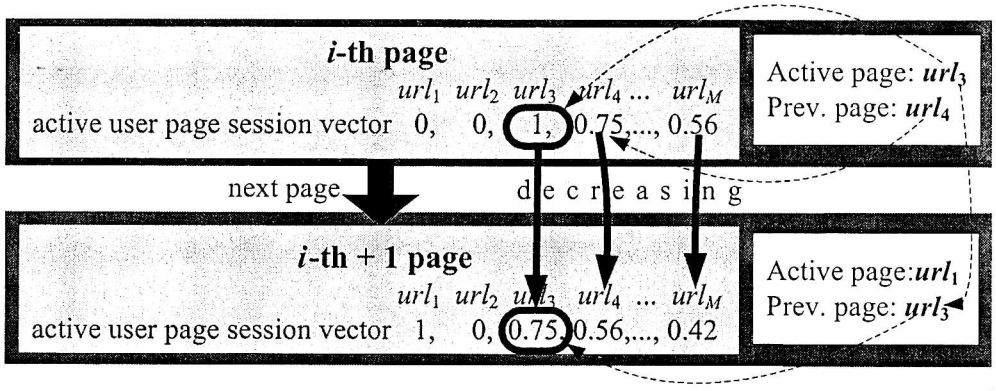

Fig. 6.8. Modification of active user page session vector ps_j after the next user web page request using the exponential function; λ=0.75

The second linear approach to user monitoring needs only a limited amount of data to store user behaviour. Only K lately visited pages are stored for each user:

$$w_{ji}^{ps} = \begin{cases} \dfrac{K-k}{K}, & k \le K, \\ 0, & \text{otherwise,} \end{cases} \quad (6.7)$$

where K - the constant parameter; k - the consecutive index of the document d_i in the jth active session in reverse order.

For the just viewed page, k=0; for the previous page, k=1; etc. In other words, w_{ji}^{ps} is linearly decreased from 1 to 0 with the step $1/k$ (Fig. 6.9). The value of K depends on the number of potential concurrent users and available system resources. The greater the K we use, the more resources we need.

First, the exponential function Eq. (6.6) suddenly drops, then gradually approaches 0. This enables recent changes in user behaviour to have the greatest influ-

ence on user assignment to the closest conceptual space and the best usage pattern, see Sec. 6.6.1. Nevertheless, this also considers all previous pages. However, a lot of space may be required to store all ps_j vectors in the case of many concurrent users. Since only K numbers need to be retained for each user in the linear approach Eq. (6.7), we save space but at the same time we probably lose accuracy.

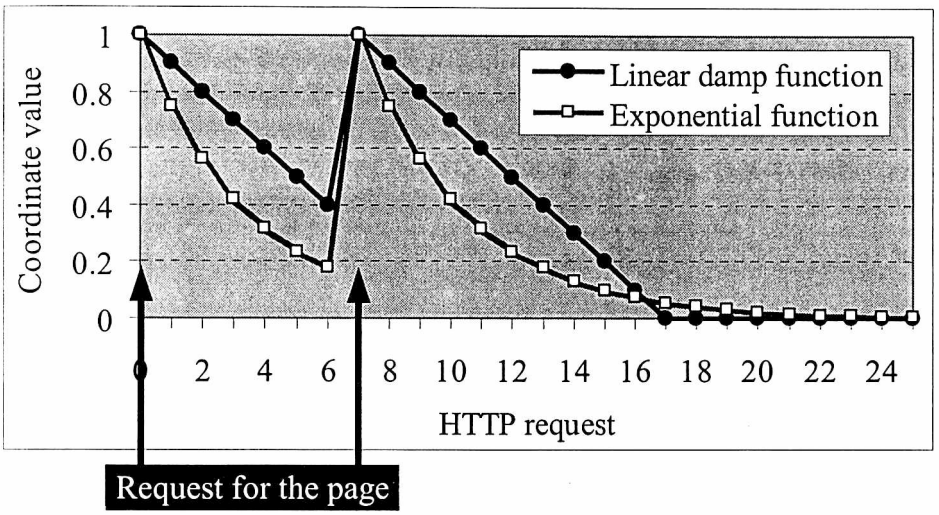

Fig. 6.9. The value of the coordinate of the active user page session vector for page d_i with two different ways of calculation. Page d_i is visited at request no. 0 and 5. λ=0.75, K=10

The active user ad session vector $as_j = < w_{j1}^{as}, w_{j2}^{as}, ..., w_{jN}^{as} >$ plays a role similar to the active user page session vector in relation to displayed advertisements. It prevents advertisements from being displayed too often and enables their periodical rotation. The coordinate $w_{ji}^{as} \in [0,1]$ denotes when the ith advertisement was shown to the jth current user. Values in the vector are always updated after advertisements have been assigned to the user and displayed on the web page. The w_{ji}^{as} value is set to 1 for the just emitted ith advertisement after the jth user's request. At the same time, all other previous values $w_{ji}^{'as}$ are decreased using factor $\alpha \in [0,1]$, as follows (Fig. 6.10):

$$w_{ji}^{as} = \alpha * w_{ji}^{'as} . \tag{6.8}$$

It was assumed in the implementation that α=0.8. The active user ad session vector as_j with a value of 0 at all positions, is created with the first request from the current active user and is removed after the user's session has finished.

Formula (6.8) is equivalent to Eq. (6.6), and as with the active user page session vector ps_j we would need to restrict the space required for as_j because of resource limitations, depending on the number of all advertisements (N) and the potential number of concurrent users. If the value of these numbers is too high, the linear function can be introduced as in Eq. (6.7).

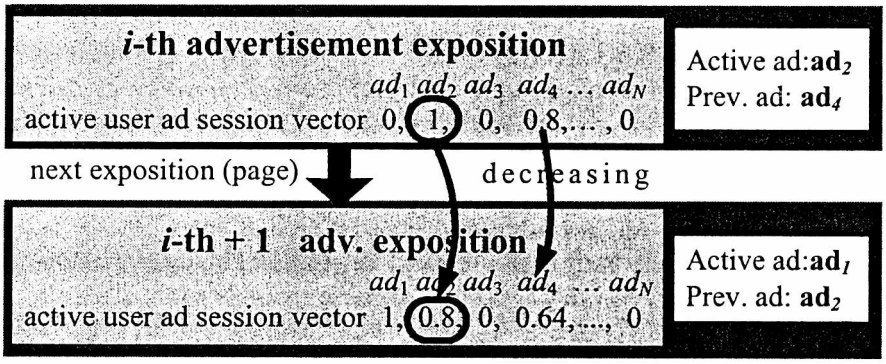

Fig. 6.10. Decreasing the coordinates of active user ad session vector as_j after displaying an advertisement; α=0.8

Information about the number of emissions of every advertisement is stored in the *ad emission vector* $\boldsymbol{e_j}=< w^e_{j1}, w^e_{j2},..., w^e_{jN} >$; where the value of w^e_{ji} is the number of emissions of the ith advertisement for the jth active user. Information kept in the ad emission vector is necessary in order not to display one advertisement too many times to one user and is useful in controlling advertising policy.

6.5.4. Advertising Policy: Emission Limits, Priority

Many publishers allow advertisers to specify an emission limit of one advertisement to a user during a single user session. The number of permitted emissions of the ith advertisement is denoted by the coordinate w^{epu}_i of *the emission per user vector* $\boldsymbol{epu}=< w^{epu}_1, w^{epu}_2,..., w^{epu}_N >$. The information about the acceptance of the emission of particular advertisements for the jth active user is stored in the vector *ad emission acceptance for the jth user* $\boldsymbol{eau_j}=< w^{eau}_{j1}, w^{eau}_{j2},..., w^{eau}_{jN} >$. The value of the coordinate w^{eau}_{ji} depends on the general limit of emissions of the ith advertisement (***epu***) and the current number of emissions of this advertisement to the jth active user (***e***), as follows:

$$w^{eau}_{ji} = \begin{cases} 1, & \text{if } w^{epu}_i - w^e_{ji} > 0 \text{ or "emission is unlimited",} \\ 0, & \text{otherwise.} \end{cases} \tag{6.9}$$

As mentioned above, the publisher is able to increase the importance of each advertisement. The appropriate, manually set priorities are stored in *the ad priority vector* $p = < w_1^p, w_2^p, ..., w_N^p >$, where $w_i^p \in [0,1]$.

6.6. Personalization and Final Filtering

The personalization process in AdROSA consists of two stages: user assignment and vector integration (Fig. 6.11). Both are accomplished individually for each user visiting the web site at each user request. In the former (see Sec. 6.6.1), the user is assigned to previously obtained offline patterns, while the latter (see Sec. 6.6.2) integrates all information obtained about the user, their behaviour, and the advertising policy features to provide the most suitable personalized advertisement. In fact, the first stage of personalization consists in building personalized associations between current user behaviour and either textual contents or visiting patterns.

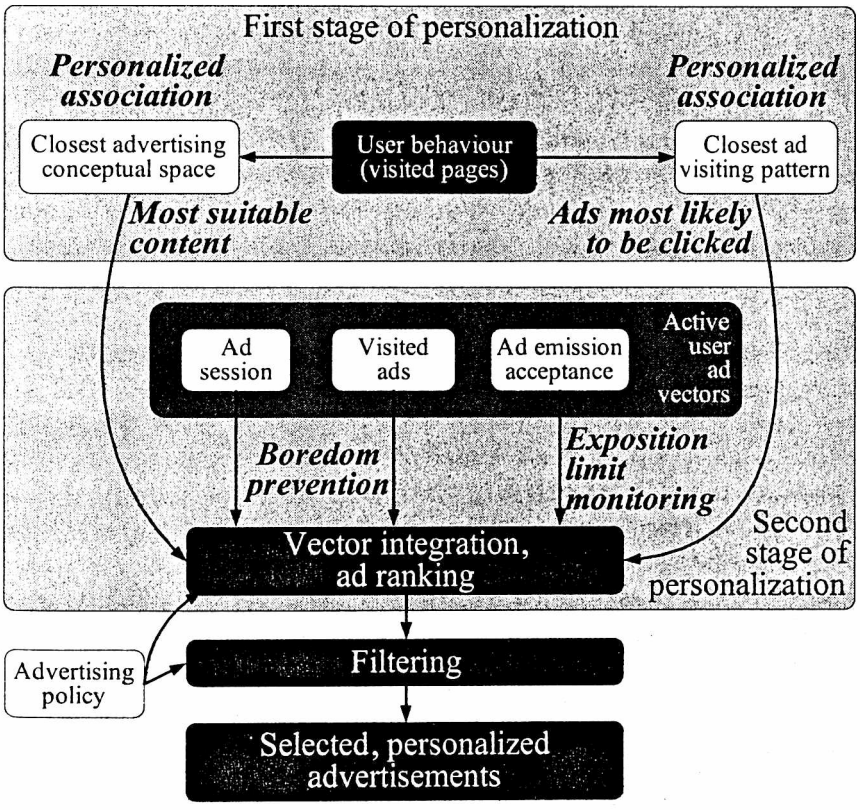

Fig. 6.11. Personalization in AdROSA

6.6.1. User Assignment - First Stage of Personalization

Current user behaviour is reflected in the data about pages visited during the user session kept in the active user page session vector ps_j, see Sec. 6.5.3. At each HTTP request from the jth active user, the AdROSA system again assigns this user to the closest publisher's conceptual space ctp_k (Fig. 6.12) and independently to the closest usage pattern $cs_{k'}$ (Fig. 6.13), by searching for centroids with a minimum value of $cos\ (ps_j, ctp_k)$ and $cos\ (ps_j, cs_{k'})$, respectively. The closest conceptual space indicates the thematic part of the publisher's portal that the current user is just visiting, e.g. music or sports, while the selected usage pattern points to the group of users with similar behaviour, e.g. sports fans or buyers.

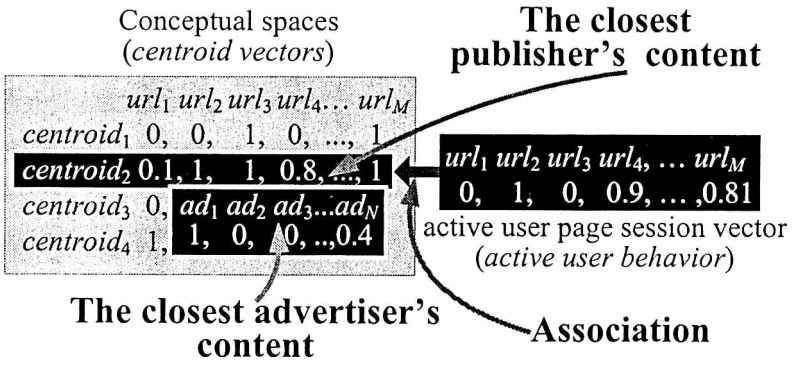

Fig. 6.12. User assignment to the closest conceptual space (thematic group)

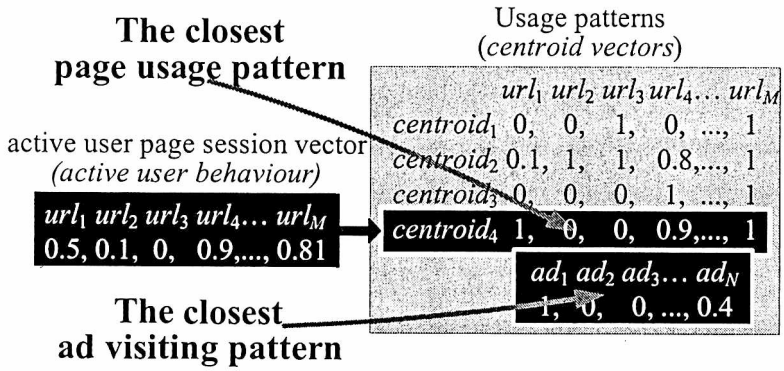

Fig. 6.13. User assignment to the closest ad visiting pattern

Each publisher's conceptual space ctp_k corresponds to one advertising conceptual space cta_k and each usage pattern $cs_{k'}$ is related to one ad visiting pattern $cv_{k'}$.

As a result, we obtain cta_k and $cv_{k'}$ suitable for the current behaviour (ps_j) of the jth active user. Note that the user can be assigned to many distinct advertising conceptual spaces and independently to many ad visiting patterns within one session. These assignments can be treated as the personalized associations between the user and the contents or behavioural patterns.

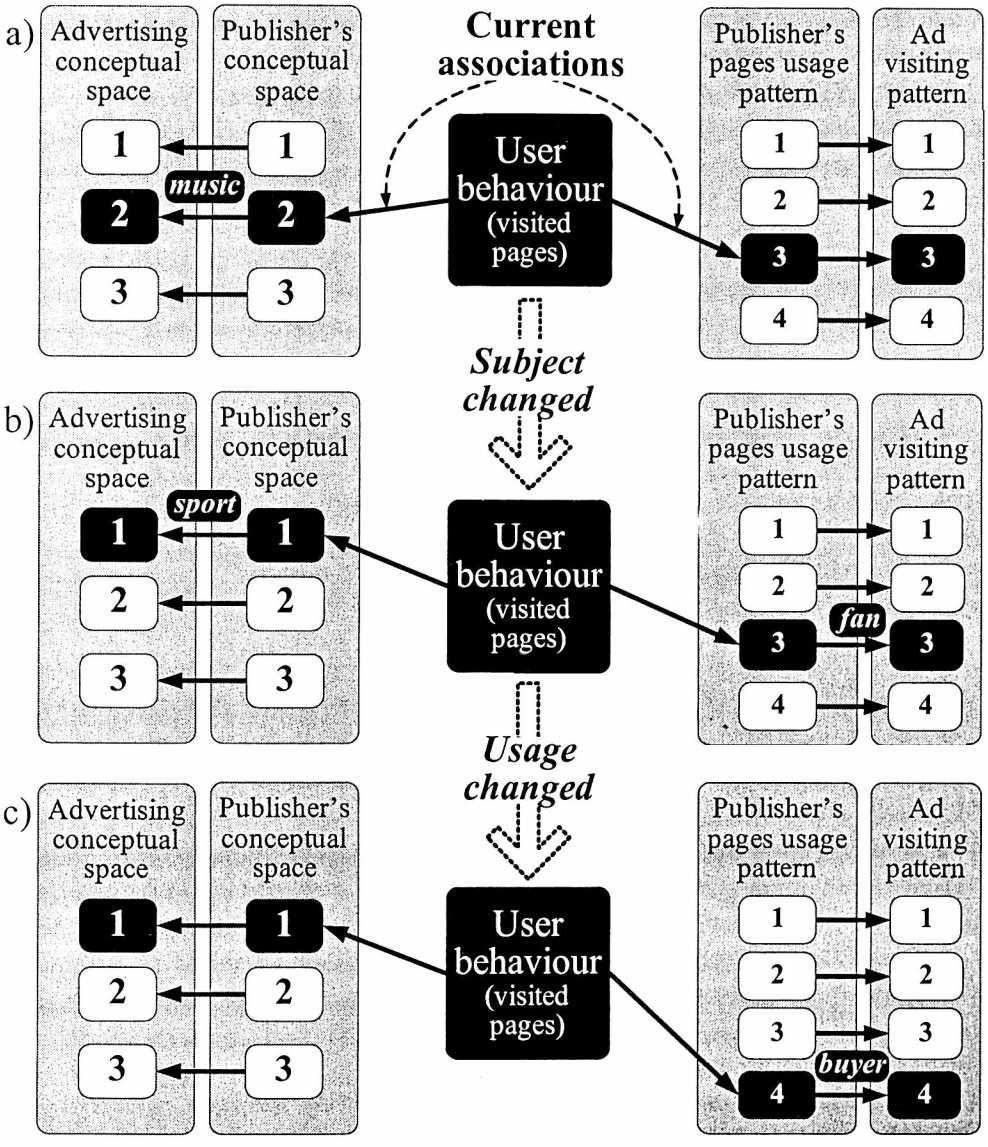

Fig. 6.14. User assignment to the most appropriate advertisements with respect to content and click-through probability – adaptive, personalized associations

Figure 6.14 illustrates how the assignment (personalized association) may change when the behaviour of the visiting user changes. In the first phase of the session – one or more initial HTTP requests – the user was assigned to the second conceptual space cta_2, appropriate to the subject matter of their current interest, e.g., music (Fig. 6.14a). In the next phase (Fig. 6.14b), the user changed the subject of viewed pages from music to sports. This made the system reassign the user to another conceptual space number 1 (cta_1) that is much closer to sports than cta_2, resulting in the selection of advertisements with different contents, e.g., sports equipment.

During the first two phases, the user behaved like a typical "fan" so the third usage pattern was the closest – cv_3. Nevertheless, the user changed behaviour and visited diverse pages in the third phase of the session. This made user behaviour more similar to that of another group of previous users–"buyers". Thus, the user was reallocated to another publisher's usage pattern (number 4) and in consequence to another ad visiting pattern (cv_4). Finally, the advertisements typically clicked by buyers were presented to the user during the third phase.

Both selected centroids (cta_k, $cv_{k'}$) are processed in the second personalization stage–vector integration.

6.6.2. Vector Integration – Second Stage of Personalization

Having obtained all the above mentioned vectors, a personalized advertisement ranking is created for each user: list of the most appropriate advertisements is obtained by sorting the coordinates of the *rank vector* – $\boldsymbol{rank}_j$. This vector integrates all the N-dimensional vectors engaged in the personalization process:

$$\boldsymbol{rank}_j = (1-v_j) \otimes (1-\boldsymbol{as}_j) \otimes \boldsymbol{eau}_j \otimes \boldsymbol{p} \otimes (\boldsymbol{cta}_k+\boldsymbol{cv}_{k'}), \tag{6.10}$$

Operator ⊗, used for two vectors, denotes the multiplication of the individual coordinates of these vectors: the ith coordinate of the first vector is multiplied by the ith coordinate of the second vector, i=1,2,...,N. This produces the third vector with the same dimension.

The rank vector includes all the information useful for recommendation of the advertisements. Owing to $(1-v_j)$, banners clicked by the current user are omitted, while $(1-\boldsymbol{as}_j)$ prevents individual advertisements from being exposed too often for one user. $\boldsymbol{eau}_j$ is responsible for monitoring whether the limit of advertisements per user has been reached and $\boldsymbol{p}$ complies with manually-specified priorities. $\boldsymbol{cv}_{k'}$ is used to encourage the display of advertisements that have been clicked by users who visited similar web pages. Similarly, the use of $\boldsymbol{cta}_k$ promotes the display of advertisements linked to web sites that contain similar words to pages previously visited by the current user.

Note that both the closest advertising conceptual space $\boldsymbol{cta}_k$ and ad visiting pattern $\boldsymbol{cv}_{k'}$ have some coordinates greater than 0. Otherwise, such centroids

would be removed or marked as inactive for advertising, see Sec. 6.5.1. The only situation where both vectors would be empty is at the start of the system, when there are no relevant advertisements at all.

All component vectors in Eq. (6.10), except priority vector $\boldsymbol{p}$, are user-dependent and may change their values according to current user behaviour.

6.6.3. Filtering

Next, an ordered list of advertisements is filtered using additional advertising policy features stored in the database. In this way, the requirements of certain web browsers or the time of day of the emission can be fulfilled. All advertisements are also filtered according to their shape, strictly determined by the page layout. As a result, the AdROSA system delivers personalized, periodically changed advertisements meeting various advertising policy features (Fig. 6.15).

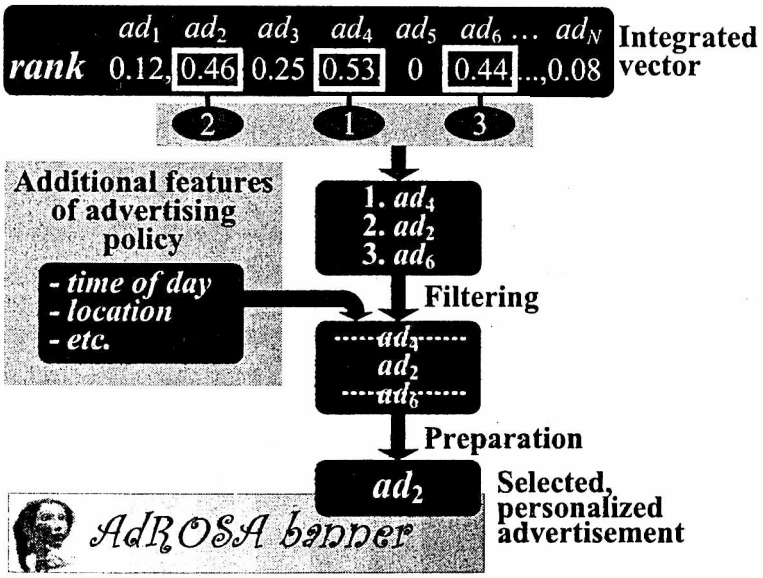

Fig. 6.15. Filtering and preparation

6.7. Discussion

The AdROSA system is compared with other typical recommendation approaches in Sec. 6.7.1. Its application to different advertising models, see Sec. 6.2, is discussed as well. The concept of AdROSA also enables the introduction of some new advertising measurements useful for new kinds of advertising policies.

Legal issues, user concerns, the problem of new items (cold start), the possible continuation of previous sessions as well as maintenance and efficiency matters are discussed in Sec. 6.7.2.

6.7.1. AdROSA vs. Other Models, Advertising Metrics

One of the most popular recommendation approaches is collaborative filtering [Buo01, Cun01, Her00, Lee02, Ter97], widely used in commercial recommendation systems utilized by many companies like Amazon.com or CDNow [Mon03, Sch01]. Typically, the collaborative filtering method is based on item ratings explicitly delivered by users. The system recommends items (web pages, movies, products in e-commerce, etc.) that have been evaluated positively by similar users. The similarity between individuals is a fundamental issue in this method. Usually, a set of "nearest neighbours," whose ratings have the strongest correlation with the user, is extracted by the system. This general idea is - in a sense - close to the AdROSA method. The user is also assigned to the most similar set of web sessions - usage pattern. However, there are some significant differences. The assignment is performed dynamically at each user HTTP request, which results in flexible adaptation to changes in user interest, see Sec. 6.6.1, Fig. 6.14. Since the necessary user data is gathered without cooperation from the user, the user is not forced to deliver any information. AdROSA respects "the freshness of elements" in the user profile by means of decreasing the importance of pages visited long ago, see Sec. 6.5.3. The user profile is the set of vectors monitoring user activities. They are equivalent to the set of ratings provided by the user in a collaborative filtering approach. Yet another obvious difference is that user identification is not necessary, so AdROSA can be introduced to open-public web portals.

Demographic targeting, in which advertisements are assigned to the potential customer according to their demographic features such as age, gender, education, location, interests, etc., is another quite popular approach to advertising [Hun98, Kru97, Paz99]. However, this model requires the identification of a customer and prior knowledge about them that hardly fit web portals with anonymous users. In addition, it is very difficult to collect reliable data and keep it up-to-date because web users usually avoid filling out forms, not to mention updating them. Demographic approaches do not provide any kind of adaptation to the interest changes of individual users. Since collected demographic features are usually very general, the item assignment is not precise enough [Mon03]. Nevertheless, some hybrid approaches have been proposed in which the static, stereotypical data delivered by the user is combined with information about user activities gathered automatically by the system [Ard00]. This partly overcomes the difficulty of keeping personal data up-to-date as well as the cold start problem (see Sec. 6.7.2), but such methods still require active cooperation with the user at least at the beginning. In the AdROSA model, the user

remains anonymous, while the system draws conclusions from their current behaviour, which is indicated by recently visited pages and clicked banners. All necessary data is monitored without any user effort and even without their knowledge.

In typical demographic targeting, suggested items are statically assigned to the user, meaning that an only fixed set of items is destined for each user based on their profile. For example, if they were young and liked rap music, advertisements linked to portals with this type of music would always be presented to such users, regardless of their current interests. AdROSA acts in a dynamic and adaptive way that reflects current user behaviour.

Yet another recommendation approach is based on content. In content-based filtering, suggested items are selected based on past user preferences. Thus, a good similarity measure between items is essential to calculate the closest ones. Since content-based systems are normally used to recommend text-based items (articles, books, web pages), the content in such systems is normally described with descriptors–terms, which are expected to be the most informative and distinctive. There are many different user-dependent data sources such as web pages or books rated by users [Moo00, Paz97, Paz99] or user queries [Lu02], from which descriptors are extracted and selected. Having obtained these descriptors, the system retrieves items containing similar content – the nearest neighbour method. AdROSA works, in principle, in an analogous way. The main difference is its dynamic adaptation and rotation mechanisms that overcome the problem of overspecialization [Ado05, Mon03], which results in recommending the same, limited set of items usually already seen or rated by the user. Since the allocation process in AdROSA is repeated at each user request, the user is dynamically assigned the different advertisements at each step. Additionally, the boredom prevention mechanism – vector ***as*** (see Sec. 6.5.3) also decreases the emission frequency of the same advertisement. Unique term weighting – formula (6.1) – perfectly adapts the AdROSA system to the web environment. The application of previously calculated representatives of conceptual spaces enables reduction of necessary online comparisons – the nearest neighbourhood – which is rarely used in typical content-based approaches.

The AdROSA system is a hybrid, benefiting from a combination of different methods: usage patterns, conceptual spaces, and a rotation mechanism.

Originally, AdROSA was developed to be used in the publisher model of advertising (see Sec. 6.2), in which a single publisher organizes and manages all advertising processes. In another recently quite popular broker model, there is an additional element – a broker that links many independent publishers with many advertisers. To enable the introduction of AdROSA to the broker model, we would need to extend all the vectors to one integrated publishing space, meaning that all publishers' web pages would be joined in one consistent vector space. In addition, it would be crucial to relay data from the publisher's web server to the broker because only the latter would be responsible for the entire advertising process (Fig. 6.6). The necessary data would

be the URL address of the requested page and the session ID. The only significant problem in this case is identification of the session. If the publisher detects the user session then it has to create and delete the session ID and monitor each user request using, e.g. cookies. In another approach the broker generates cookies at the first user request and returns it to the publisher that relays cookies to the user. In that case, the broker is responsible for the detection and monitoring of user sessions. The problem of a session crossing many publisher sites can be solved by splitting it into separate subsessions that are equivalent to those for single publishers.

Any metrics mentioned in Sec. 6.3, i.e. CPM, CPA, CPC, CTR, etc., can be used in the AdROSA system to settle accounts for advertising. Moreover, its additional features like priority p or the selection of suitable advertisements, which are most likely to be clicked - vector ***cs*** and closest to user interests - ***ctp*** (see Sec. 6.6.1), enable the publisher to increase these metrics, especially for the most profitable advertisements. Monitoring the emissions of advertisements to a single user makes it possible to propose some new metrics such as

- TNU - *the total number of users*, i.e. the number of sessions in which the given advertisement was exposed. To maximize the value of TNU, for the ith advertisement the appropriate coordinate in ***epu*** should be set to 1: $w_i^{epu}=1$.
- TNUK - *the total number of users with k level reached* - the measure useful when we want to gain users who have watched the ith advertisement at least k times. Use $w_i^{epu}=k$ for this case.

Both for TNU and TNUK, the ith advertisement will be presented at most a certain number of times to each user: once or k times, respectively. Note that all advertisements that are shown more than the specified limit are, in a sense, a financial loss for the publisher, so component ***eau*** in Eq. (6.10) prevents such superfluous displays.

Component $(1 - v_j)$ in Eq. (6.10) that excludes already clicked advertisements and the rotation mechanism help to avoid useless or annoying repetitions. This creates space for the presentation of other potentially profitable advertisements.

The introduction of the TNU and TNUK metrics to online advertising can create a significant competitive advantage. This would be hardly possible in traditional advertising systems.

6.7.2. "Cold Start", System Maintenance, Vector Size, Privacy Prevention, and Other Problems

Many personalization systems suffer from so-called "cold start" problems related to the shortage of data for new items [Sch02]. The AdROSA system is based on data reflecting current user behaviour so it has a certain shortcoming that appears at the beginning of a user session. At that time, the active user page session vector $\boldsymbol{ps}_j$ is

empty and it cannot be reliably matched to any existing navigation patterns. The question is which advertisement should be proposed first. This problem could be partially solved by treating the current session as a continuation of the previous one. However, the realization of this idea is attainable only if the user is identified, i.e. they have logged in. On anonymous sites, a user could be identified only with a fair degree of certainty either by using cookies or by the appropriate matching of IP addresses. However, what about new or unrecognized users? After a user's first request, the system has information delivered by a web browser in the header of the HTTP request. This data may be used in geographical targeting, which can be introduced at the filtering stage, see Sec. 6.6.3. Note that after a few requests, AdROSA already has enough information to properly personalize advertisements according to the method presented in the previous sections.

Another problem appears when a new advertisement is recently added and the system does not possess any usage data concerning its visits. As a result, such an advertisement does not occur in any ad visiting pattern (null values of appropriate coordinates in all $\boldsymbol{cv}$ vectors). Thus, advertising conceptual spaces ($\boldsymbol{cta}$) had to be updated at insertion, and the user had a chance to have the new advertisements suggested, if they had the most suitable content. This significantly decreases the problem of a new item. An additional solution is provided by dynamically updating an advertisement's priority vector $\boldsymbol{p}$, which is determined from the time of the advertisement insertion and the time remaining until the end of the advertising campaign as well as the number of visits so far. Thus, new advertisements are promoted with the value of the priority coordinates. Moreover, the priority coordinates may be automatically adjusted according to the given schedule of the advertising campaign.

The appearance of a new page in a publisher's web site results in the extension of appropriate vectors and the appreciation of M - the total number of web pages. A new coordinate is added in all term vectors $\boldsymbol{tp}$ using formulas (6.1) and (6.2). Additionally, Eq. (6.4) is used but only with reference to the new coordinate to extend clusters adequately. Note that we do not rebuild clusters each time a new page is inserted. This process is performed only after a certain number of new pages have been added.

Any updates of page content are also monitored, but no vector modification is performed until the given threshold of changes is exceeded. Usage clusters are readapted either periodically or after a certain number of new user sessions. The deleted advertisements as well as publisher's pages are simply marked as inactive. The problem of synchronizing the update process in a multi-agent recommendation system was considered in [Kaz03b], while change factors in advertising were presented in [Kaz05a].

The entire AdROSA system uses one coherent vector space with the dimension of $N+M$, where N, M - the number of advertisements and web pages, respectively, see Sec. 6.5.1. The quantity of some vectors is reduced by performing clustering offline. It regards term vectors - conceptual spaces, and session vectors

- usage and ad visiting patterns, see Sec. 6.5.1 and 6.5.2. Since the reasonable total number of clusters is 10–50, the representatives of these clusters - formulas (6.4) and (6.5) - occupy relatively little space on the disk even if N and M are large. Yet another potential problem with vector space explosion is related to online user vectors, see Sec. 6.5.3. With a great number of simultaneous users, there are many session vectors ***ps*** and ***as***, which have to be matched with the closest conceptual space and usage pattern. For the sake of efficiency, the number of non-zero coordinates in ***ps*** and ***as*** can be reduced by using Eq. (6.7) with small number of K instead of Eq. (6.6), and similarly for Eq. (6.8). To optimize calculations only non-zero values of the coordinates of all vectors are retained and processed.

The system treats one user visit in the web site as a single user session, see Sec. 6.5.2. Nevertheless, it would be possible to continue the user's previous session even after a long time, e.g., based either on the cookies returned by the browser or on the IP address joint with the browser name. In this case, a relatively large amount of online processing would be necessary to find the previous session in the historical database. Hence, such an approach would probably be hard to execute in open public portals with thousands of daily hits. Additionally, such an attitude may be either illegal or restricted in some countries or regions, e.g. according to Teledienstedatenschutzgesetz (German Teleservices Data Protection Act) [Tel97] and Directive on Privacy and Electronic Communications by the European Parliament [Dir02] usage logs must be deleted or made anonymous after each user session. Static IP addresses are generally treated as identifying characteristics of web users [Dir02 - section 28, Kob02].

In general, Internet users are interested in receiving personalized content on the web sites they visit: about 80% of users in 2005; 63% of users were concerned with the security of their personal data, and only 59% of respondents indicated a willingness to provide preference information by themselves, compared with 65% in 2004. Additionally, only 46% of users were ready to provide demographic data in 2005, down from 11% in 2004 [Cho05]. Thus, we can say that people would like to have personalized content but possibly without any usage of their personal data.

The primary AdROSA features utilize data that come from monitoring human behaviour, i.e. only the user's navigation is tracked by the system and any personal data is gathered like name, location, or interests. Nevertheless, in some countries, even activities for monitoring an active user may be restricted due to privacy protection given by national regulations or international agreements [Dir02, Kob02, Tel97]. Note that AdROSA does not need to preserve personal data after the user session has finished. For anonymity purposes, an IP address for each closed session can be replaced with the consecutive session ID generated by the system. In this case, only this session ID is used to store session vectors in the database. Also, if we used incremental clustering algorithms instead of typical ones, the session vectors would be removed after clustering.

The simple - but limited - remedy for privacy legal problems as well as user fears is the proper introduction of P3P (Platform for Privacy Preferences) that simply makes public the privacy policy of the publisher in a standardized format [Mar02]. Another solution is that personalization is performed only after the user's explicit consent at the entrance to the portal [Tel04]. Privacy can also be protected by the introduction of the special, dedicated anonymity and pseudonymity mechanism in which many systems within one network exchange user information using cryptography and some mixing techniques. Nevertheless, this approach requires many web portals to cooperate with one another and it is completely useless in the case of the publisher model of advertising (Fig. 6.3) [Kob03]. In conclusion, we should emphasize that the AdROSA system satisfies most crucial privacy postulates, since it exploits neither typical demographic data nor the previous behaviour of current users. It also does not need to retain any personal data from the finished user sessions.

6.8. Demonstration of AdROSA Activities

For demonstration purposes, the AdROSA prototype was installed on the local copy of a Polish portal, poland.com. Next, advertising campaign parameters for 10 banners were inserted into the system (Fig. 6.16) and a hypothetical user session was carried out.

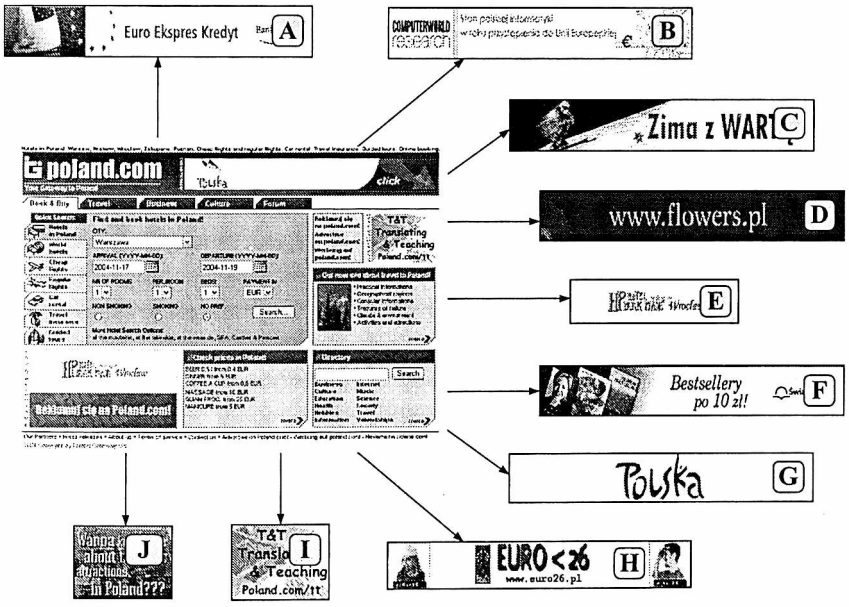

Fig. 6.16. The main web page of the poland.com portal with all banners considered

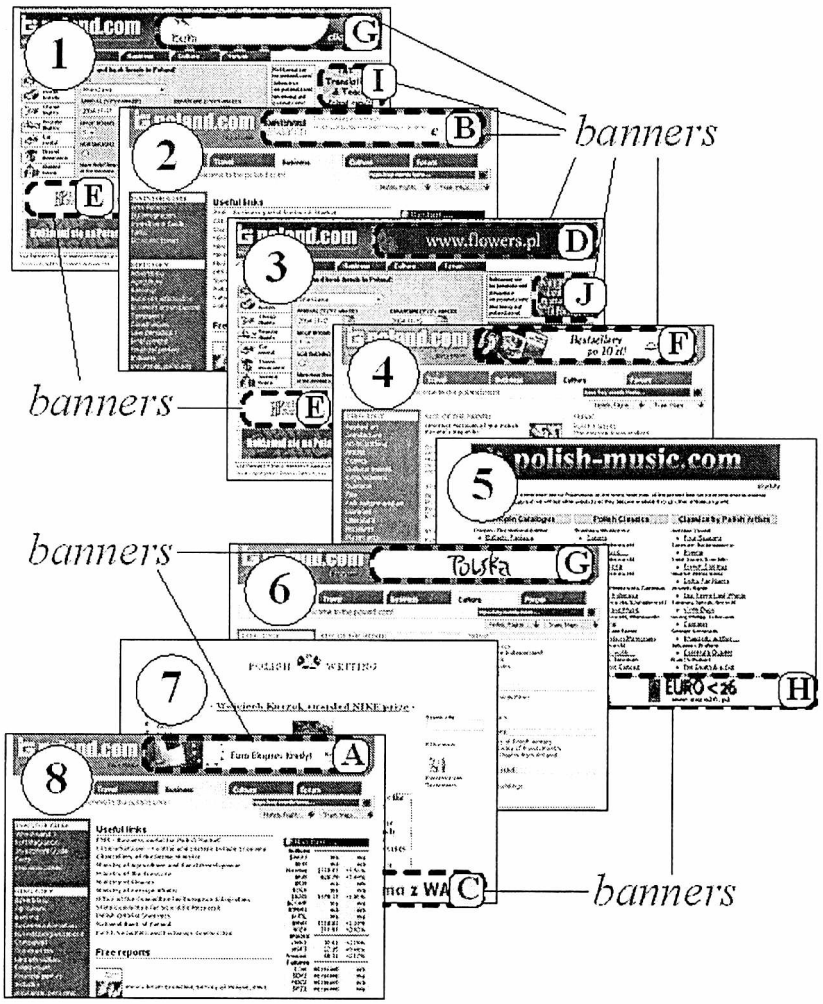

Fig. 6.17. A sequence of eight user requests

A sequence of eight user requests during a single session is presented in Fig. 6.17. On each page, some advertisements were selected by AdROSA and displayed to the user. The navigation path and the displayed advertisements, as well as the closest centroids, are shown in Table 6.1. At the beginning of the session (the first request, main page), the user was assigned to the ad visiting pattern 17 and advertising conceptual space 2, and, as a result, banners E, G, and I were displayed. The user clicked on banner I. After the third request, when the user came back to the main page, he was reassigned to the ad visiting pattern 14, even though the advertising conceptual space remained the same. Banner E was displayed again, but two other new advertisements (D, J) were suggested instead of

G and I. This was the result of the assignment of a new ad visiting pattern (it changed from 17 to 14), so new banners D and J were promoted, while G was excluded. Additionally, banner I, which was clicked in the first request, was blocked owing to "1" in the v vector. After the user moved to the *Culture Section* in the fourth step, the closest advertising conceptual space changed and banner F was displayed. In the following requests, the user visiting the *Culture* and *Music sections* continued to be assigned to the previous patterns but the banners kept changing.

Table 6.1. An example sequence of user requests

Request sequence	Requested page	The closest ad visiting pattern	The closest ad conceptual space	Displayed ad	Clicked ad
1	Main Page - *http://www.poland.com*	17	2	E, G, I	I
2	Business - *http://business.poland.com*	17	2	B	
3	Main Page - *http://www.poland.com*	14	2	D, E, J	D
4	Culture Section - *http://culture.poland.com/*	14	11	F	
5	Music - *http://music.poland.com/*	14	11	H	
6	Culture Section - *http://culture.poland.com/*	14	11	G	
7	Polish Writing - *http://www.poland.com/directory/link.php?rangeid=2063&url=go.php?id=2063*	21	19	C	C
8	Business - *http://business.poland.com*	21	6	A	

Having analyzed the data collected in Table 6.1, we observed that, depending on the navigation path, the user was reassigned to different navigation patterns, therefore to a different advertising conceptual space. Similarly, the closest ad visiting pattern changed. According to AdROSA's assumptions, advertisements, which were clicked on during the session, were never displayed again to the user. If the user returned to the same page (e.g., the *Culture Section* was visited in requests 4 and 6), the offered advertisements were different because banners are not statically assigned to a specific page.

Some advertisements were displayed twice during the session, but due to the AdROSA boredom prevention mechanism (active user ad session vector $\boldsymbol{as}_j$), this never happened in successive requests, i.e. advertisement G was displayed at the first and then only at the sixth request. This mechanism also forced changes of the banners during the 4-5-6th requests, even though the user was assigned to the same conceptual space and ad visiting pattern. Additionally, each advertisement

was prevented from being displayed too many times, according to the value of eau_j vector, e.g., banner E is allowed to be presented only twice.

A user may be assigned various centroids on the same page, depending on their behaviour, e.g., the closest conceptual space for the *Business* page was once 2 and once 6. This results from having different previous pages and a different value of the active user page session vector ps_j.

To conclude, the user was offered personalized advertisements, depending on their behaviour and the content of the pages visited during the analyzed session.

6.9. Multi-agent Architecture of AdROSA System

AdROSA consists of two main parts: AdROSA Server, which is a part of ROSA (*Remote Open Site Agents*) server, and AdROSA web. The Server is a multi-agent system performing all offline operations and it executes a remotely invoked advertisement personalization process. The AdROSA web is an application responsible for advertisement display and session data management. The preliminary version of the AdROSA multi-agent architecture was presented in [Kaz05a], while the one described below was finally developed and implemented in a prototype.

Both ROSA and AdROSA have been developed as multi-agent systems, whose expert agents cooperate with one another and may be distributed among many hosts. Every agent is responsible for a single task and encapsulates specific functions that would be available for the rest of the system. For that reason, the agents possess their own knowledge, which they interchange with one another. The multi-agent architecture of the system is presented in Fig. 6.18. ROSA agents, marked in white, are responsible for the operations on a publisher's site, while AdROSA agents (marked in black) accomplish tasks strictly related to advertisements.

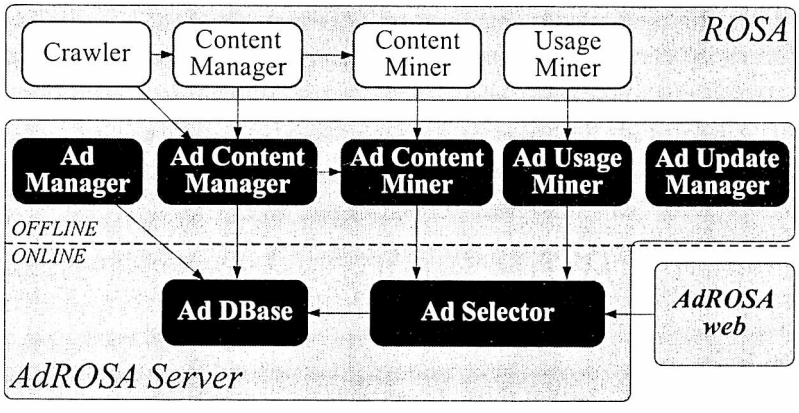

Fig. 6.18. Multi-agent architecture of AdROSA system

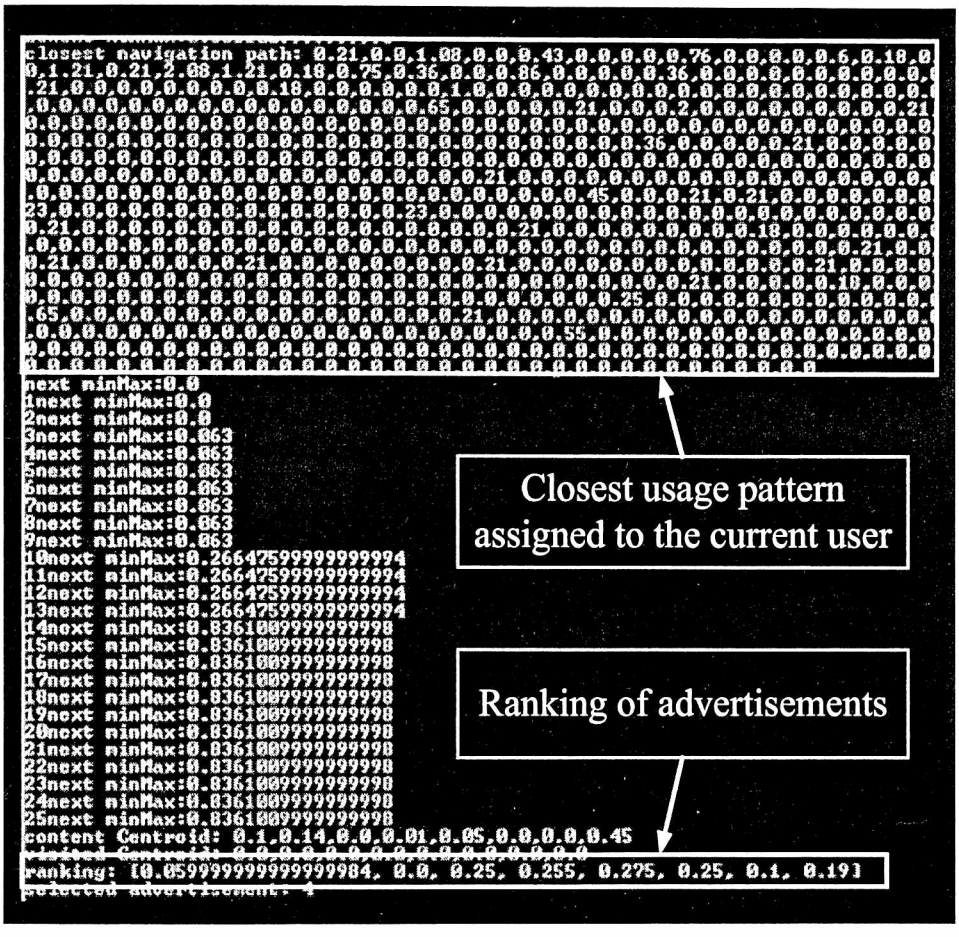

Fig. 6.19. AdROSA management console after ranking advertisements

Crawler retrieves the site content using HTTP; it extracts terms and calculates the document-term frequency. It is used to analyze both the publisher's web site and the advertised web sites linked by advertisements.

Content Manager creates the vector representations of terms, which occurred on the web site, and determines its significance according to their frequency. *Ad Content Manager* uses the capabilities of Content Manager and performs analogous operations on the terms retrieved from advertised web sites.

Content Miner and *Ad Content Miner* cluster term vectors and calculate the mean vector (centroid) of each cluster. They execute the tasks for the publisher site and the advertised sites, respectively.

Usage Miner clusters historical sessions and then calculates their centroids representing typical usage patterns.

Ad Usage Miner creates visited ad vectors, clusters them, and calculates a centroid. The centroid represents the most likely clicked advertisements by users navigating the web site according to the corresponding usage pattern.

Ad Update Manager is responsible for periodic web site monitoring and recognizing whether changes are serious enough to start the update process. Ad Update Manager controls the validity of campaigns and performs an automatic update of clusters, which is necessary for correct recommendation.

Ad Manager includes a user interface and provides advertising campaign management tools.

Ad Selector performs an online personalization process: it creates necessary vectors, carries out a ranking, and returns a set of the most appropriate advertisements. It is also responsible for filtering of advertisements - the second personalization stage.

After each request, the *Ad Selector* agent may display to the administrator on a Java console the values in all vectors involved in the personalization process and the weights of individual advertisements in the ranking. The example screenshot of the console is presented in Fig. 6.19.

Ad DBase provides database connectivity. It performs operations requested by other AdROSA agents on the database.

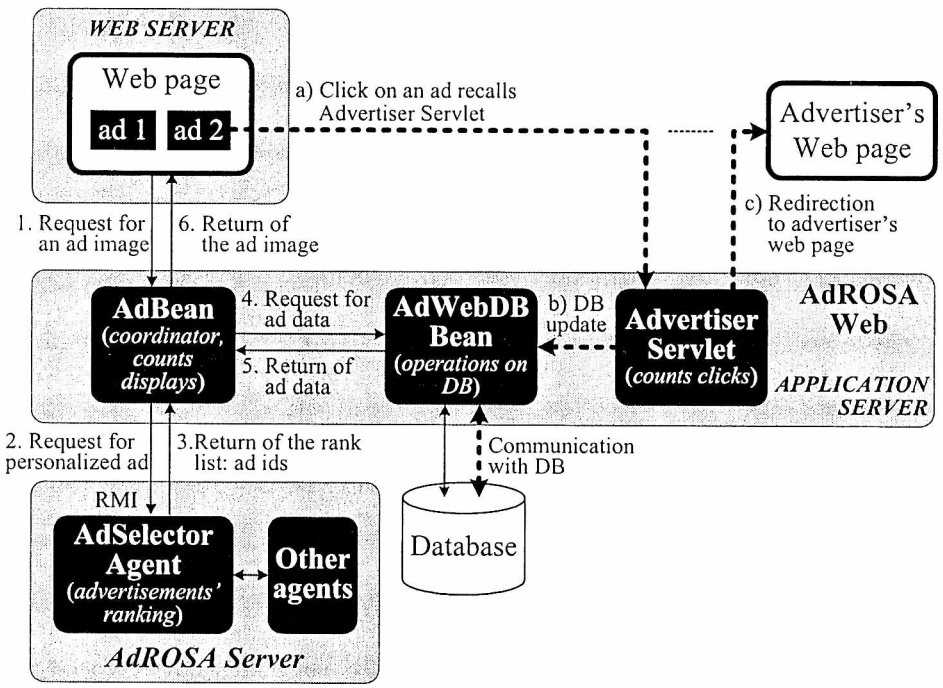

Fig. 6.20. Physical layers of the AdROSA system

As mentioned before, AdROSA Web is a web application, which controls the display of advertisements on a web site. It runs on the application server (Tomcat) and consists of a set of JSP pages, Java Beans, Servlets, and Java Scripts communicating with each other to manage the online process of advertisement personalization. Two main parts of the online process can be mentioned (Fig. 6.20): *emission* (constant line), and *click* (dashed line). The former is an emission of a personalized advertisement on a web page requested by a user. The latter is run after a user clicks on the advertisement leading to the advertiser's web site.

Working AdROSA is totally invisible for a web user visiting a web site, who may not even be aware of the personalization process performed after each request.

6.10. Conclusions

The integration of information coming from different sources such as web usage mining, web content mining, advertising policy, and boredom prevention is the essential issue of the hybrid, personalized advertising method presented in this section. The large number of factors considered means that the same user on the same page may be recommended different advertisements each time.

The crucial component of the method is the adaptive creation of the association between the current user and the advertisers' web sites with respect to advertising policies. Additionally, the advertisements recommended to the user according to the computed associations are more likely to be clicked.

Almost all processes in the method (Fig. 6.6) are performed automatically by the system, which decreases management costs. The idea of personalization based on "user-friendly" data acquisition (without the user's effort) makes the AdROSA system applicable in most open-access, anonymous web portals and can widen typical personalization systems based on demographic or collaborative filtering used in many web sites. Although the system works in the portal (publisher) model of advertising (Fig. 6.3), it can be relatively easily extended, with some limitations, to the broker model. The integration of the AdROSA system with the ROSA core system [Kaz03b, Kaz03c, Kaz04d], which recommends hyperlinks, results in a complex, coherent personalization framework that can satisfy both users and advertisers.

7 Using Associations in Virtual Social Networks to Evaluate Social Position of Individuals

Web-based services, which enable people to communicate with one another and share their interests, reveal that human relationships have moved more and more from the real world to the virtual, internet world. This process began when the first email service started, but currently we can observe its substantial expansion in many systems such as instant messengers, blogs, wikis, dating systems, online social network systems like Friendster [Boy04] or LinkedIn multimedia publishing systems like Flickr or YouTube, and many, many others. Humans with their spontaneous but at the same time social and collaborative behaviour are the most significant, and concurrently the least predictable element in each of these systems.

Users who interact with one another or share common activities form in the natural way a user social network. A social network consists of two components: a finite set of users, usually called actors, who are the nodes of the network, and a set of ties that denote the associations between the nodes [Gart97, Han06, Was94, Yan06]. In this section, ties reflect behavioural interactions between users or their common activities. For example, a couple of people who send email messages to each other, who learn the same lessons in an e-learning system, talk to each other or who comment on the same internet blogs for the sake of these activities are in a mutual relationship. Each of the enumerated services can provide suitable source data about communication between their users.

Based on the data derived from a source system, we can build a social network of its users and then analyze its links [Don05]. This especially refers to studies on positions and importance of each user within the network. This would help to discover some users who occupy the highest social statement and probably the highest level of trust [Gol04, Ran04] and importance. These users can be recognized as key persons or representatives of the entire community. A small group of key persons is potentially very useful for the social network management since they can initiate new kinds of actions, spread new services or activate other network members. On the other hand, users with the lowest social position should probably be stimulated to greater activity or be treated as the mass, target receivers for the prior prepared services that do not require a high level of involvement.

The social position of an individual described in this section is a new centrality measure of personal social statement within the weighted and directed network. The value of social position depends on both the strength of association a person maintains as well as the social positions of all their acquaintances. Moreover, not only is the social position inherited from others but the level of inheritance depends on the commitment in activity of the acquaintances directed to this person.

The main features of the social position measure that result from its iterative calculation as well as experiments on real data including email communication are presented in this section.

7.1. Social Networks

The virtual social networks, which are a special case of regular social networks, have different kinds of backgrounds and associations' characteristics. Pujol *et al.* distinguish several types of social networks based on different data sources used for their construction: personal web pages, reports or document authorships, participation in projects, hierarchical structure of organization, sharing of physical or virtual resources (news groups, forums), emails [Puj02].

Many researchers identify the communities within the web based on link topology [Fla00, Gib98], while others analyze emails to discover the social network [Cul04, Gib98, Thu79, Zhu06]. Furthermore, some of them compare the web to a social network and try to study the connection between web pages [Kum02]. Adamic and Adar analyze the text on the user homepage, links from the user homepage to other pages as well as links from other pages to the user homepage and mailing lists. From all this data they extract social networks [Ada03]. The online social networks that are defined as the set of people who are connected by a computer network are studied in [Gart97].

7.1.1. Virtual Social Networks

A virtual social network (VSN) is the network of users who meet or cooperate by means of the Internet. It consists only of humans, but not organizations or groups. Moreover, virtual network members communicate with one another over distance and maintain their relationships only through the internet services. For that reason, virtual social networks suffer from the lack of person to person contacts [Wel01].

Definition 7.1. A virtual social network $VSN=(M,R)$ is the social network in which M is the finite set of non-anonymous internet user accounts - internet identities, called network members, that communicate with one another or participate in common activities provided by internet services. R is the set of associations derived from such communication or common activities. An asymmetric association $(m_i,m_j)\in R$, which links member $m_i\in M$ to member $m_j\in M$, exists if and only if there exists any communi-

cation from m_i to m_j or activity common for both m_i and m_j. The set of members M must not contain isolated members, i.e. $\forall m_i \in M\ \exists m_j \in M,\ i \neq j\ ((m_i,m_j) \in R \vee (m_j,m_i) \in R)$, $card(M)>1$.

Definition 7.1 is based on the definition of the regular social networks formulated by Wasserman and Faust who claimed that "a social network consists of finite set or sets of actors and the relation or relations defined on them. The presence of relation is a critical and defining feature of a social network." [Was94].

The internet identity is a digital representation of the physical, social entity in an internet service, usually related to a single human being. This representation must unambiguously identify the social entity (the user in the internet service) and its task is to move the physical entity from the real to the virtual world.

If the network member x does not possess any association neither from nor to anybody, then such individual x is called isolated and should be excluded from the member set M - see member t in Fig. 7.5. The conclusion that can be drawn from this constraint is the minimum quantity of network members: $card(M)>1$. All further formulas as well as the social position function considered in this section are valid only for virtual social networks $VSN(M,R)$ without isolated members.

According to Definition 7.1, social associations are asymmetric, which means that if for example member m_i sends emails to member m_j, then member m_j does not have to reply these emails; in consequence $(m_i,m_j) \in R$ but $(m_j,m_i) \notin R$. However, common activities of two members m_i and m_j often result in two associations $(m_i,m_j) \in R$ and $(m_j,m_i) \in R$. Nevertheless, in such a case the strength of the association (m_i,m_j) usually differs from the strength of the reverse association (m_j,m_i).

Definition 7.1 prevents associations R to be derived directly only from data available in internet services. This especially regards the information about interpersonal communication like sent emails, talks through instant messengers, VoIP calls. Associations can also be created upon user activities or achievements that involve more than one member such as comments on the same blogs, links between user homepages, participation in projects or teams, e.g. membership in scientific conference program committees, authorship of documents, etc. Note that the accessibility of data can vary depending on the service: email data is available only on SMTP/NNTP servers whereas blog comments and homepages are usually public.

7.1.2. Associations and Their Automatic Extraction

An association in the social network is the connection from one member to another that reflects their common acquaintance, private or professional relation or even similarity of their activities or inclinations. The maintenance or even only existence of an association usually requires our trust, commitment, emotion, dedication of time and effort.

Several significant features can characterize a human relationship (association) like mutuality, durability, intensity, intentions, culture conditionings, emotional level and finally strength [Han06, Was94].

An association does not have to be symmetrical, e.g. Tom could be friends with John but John might not see Tom as his friend. Nevertheless, if it is symmetrical then it is usually more durable. Moreover, an association may be durable for a certain period after which it could weaken or diminish. Thus, an association is either more persistent or more temporal and the time factor is very important. If Tom sent John 20 emails over two weeks, five years ago, then John would have most probably forgotten Tom by now. However, John would remember and feel a kind of durable association with Bill who has regularly sent John one email every quarter for the last five years. Each human relationship requires periodic support and refreshment. Furthermore, the longer the acquaintance is, the more durable it is in future terms.

The importance of contact intensity and communication features on the strength of the association may result from the culture both participants live in. Ten emails sent by the representative of one nation may have greater significance than the same number of emails exchanged between individuals from other nations. Many phone calls made late at night or in time off nurture a stronger association than the same calls in regular working hours.

The strength of an association can depend on its basis, especially the kind of communication or mutual activity. The meeting of commentators of the same blog or even hyperlinks between homepages generally connect people much less than the co-authorship of a scientific paper.

Some unusual factors may also be the sign of stronger associations. An intensive correspondence in Polish is the evidence for stronger association between foreigners in Japan rather than the same communication in Japanese between natives. Nevertheless, the opposite meaning would be true in Poland.

In some environments like the worldwide internet, that is multicultural in its nature, the detection of some differences appears to be very sophisticated. Moreover, some features of human associations may either require complicated content processing like extraction of the emotion level or even be very hard to discover like intentions.

7.2. Social Position of Individuals in Virtual Social Networks

7.2.1. Social Position Concept

On the web, there is a great need to assess not only the significance of web pages created by people and published in web services [Bri98], but also the importance

of people within virtual social networks. The measure studied in this section is called social position and it enables us to estimate how valuable the particular individual within the human community is. In other words, the importance of every member can be assessed by calculating their social position. This significance of the nearest neighbours of a member is taken into consideration as well as the quality of their mutual associations.

The importance of the member in the weighted social network, expressed by the social position function, tightly depends on the strength of the associations that this individual maintains as well as on the social positions of their acquaintances, i.e. the first level neighbours influence the importance of the member in the network. In other words, the member's social position is inherited from others but the level of inheritance depends on the activity of the members directed to this person, i.e. intensity of common interaction, cooperation or communication. The activity contribution of one user absorbed by another is called commitment and is presented in Fig. 7.1 as weights of edges.

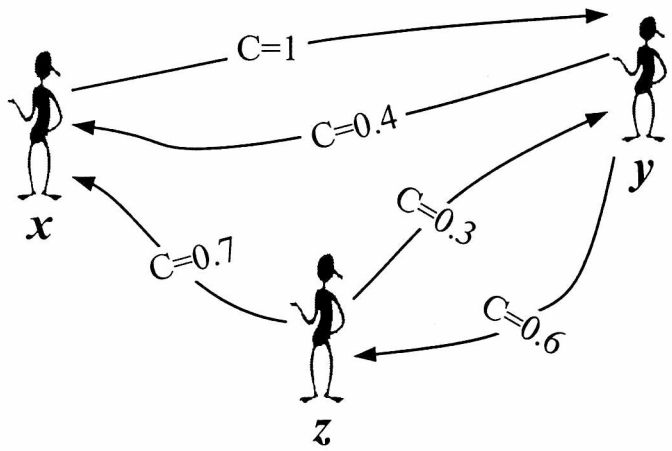

Fig. 7.1. Example of the social network with the assigned commitment values

Social position function $SP(x)$ of member x in the virtual social network $VSN=(M,R)$ respects both the value of social positions of all other network members as well as the level of their activities in relation to x:

$$SP(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{y \in M} SP(y) \cdot C(y \to x), \tag{7.1}$$

where ε - the constant coefficient from the range [0,1]; $C(y \to x)$ - the commitment function that denotes the contribution in activity of y directed to x. In other words, $C(y \to x)$ expresses the strength of the association from y to x.

The value of ε reflects the openness of human social position on external influences: how much x's social position is static (small ε) or influenced by others (greater ε).

In general, the greater the social position one possesses, the more valuable this member is for the entire community. It is often the case that we only need to extract the highly important persons, i.e. with the greatest social position. Such people are likely to have the biggest influence on others. As a result, we can focus our activities like advertising or target marketing solely on them and we would expect that they would entail their acquaintances. The social position of user x is inherited from the others but the level of inheritance depends on the activity of the users directed to this person, i.e. intensity of mutual communication. Thus, the social position depends both on the number and quality of associations.

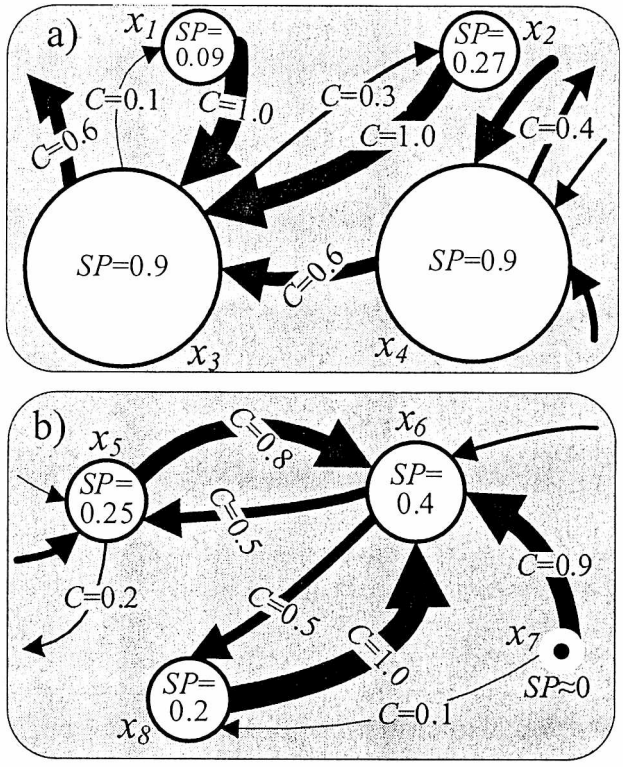

Fig. 7.2. Two fragments of a social network. The size of the node corresponds to the value of its social position. The arrows reflect commitment values. $\varepsilon \approx 1$

A user can possess the high social position if some other people transfer their high SP to them. For example, the social position of user x_3 in Fig. 7.2a is 0.9. It

mostly comes from x_3's high commitment in the activities of user x_4, $C(x_4 \rightarrow x_3)=0.6$ and $C(x_4 \rightarrow x_3)*SP(x_4)$ equals as much as 0.54. The contribution of two other users x_1 and x_2 in $SP(x_3)$ is only 0.36, even though their commitment values are the greatest possible $C(x_1 \rightarrow x_3)=C(x_2 \rightarrow x_3)=1$. On the other hand, despite the very high $SP(x_3)$, the value of $SP(x_1)$ is only 0.09 due to very low x_1's participation in x_3's activity, $C(x_3 \rightarrow x_1)=0.1$. User x_3 is the only one who is active towards user x_1. The social position of user x_6 is medium-sized: $SP(x_6)=0.4$, although three other persons x_5, x_7, and x_8 pass most of their activities to x_6: $C(x_5 \rightarrow x_6)=0.8$, $C(x_7 \rightarrow x_6)=0.9$, and $C(x_8 \rightarrow x_6)=1$, Fig. 7.2b. This results from the low or very low social position of x_6's acquaintances: $SP(x_5)=0.25$, $SP(x_8)=0.2$ and $SP(x_7)$ is almost zero. Hence, $SP(x_3)$ is high because of high $SP(x_4)$ as well as big $C(x_1 \rightarrow x_3)$ and $C(x_2 \rightarrow x_3)$; $SP(x_1)$ is low due to small $C(x_3 \rightarrow x_1)$; and $SP(x_6)$ is medium with respect to the low importance of its neighbours.

7.2.2. Association Measure - Commitment Function and Its Constraints

Commitment function $C(y \rightarrow x)$ is the measure that describes strangeness of the association from user y to user x. It is a slightly enriched version of relationship commitment $C^{rel}(y \rightarrow x)$. Function $C^{rel}(y \rightarrow x)$ is based on the association data within the virtual social network $VSN(M,R)$.

There are four important constraints regarding commitment function derived from the associations $C^{rel}(y \rightarrow x)$:

1. Relationship commitment function $C^{rel}(y \rightarrow x)$ is directly derived from the data describing associations from y to x in $VSN(M,R)$, $x,y \in M$, $x \neq y$. If there exists the association $(y,x) \in R$ then $C^{rel}(y \rightarrow x)>0$. If there is no association from y to x, i.e. $(y,x) \notin R$ then $C^{rel}(y \rightarrow x)=0$.

2. The value of relationship commitment is from the range [0,1]: $\forall (x,y \in M)$ $C^{rel}(y \rightarrow x) \in [0,1]$.

3. Relationship commitment function to itself equals 0: $\forall (y \in M)$ $C^{rel}(y \rightarrow y)=0$.

4. If at least one association commitment from y is greater than 0, then the sum of all relationship commitments from y has to equal 1:

$$\forall (y \in M)\ \exists (x \in M)\ C^{rel}(y \rightarrow x)>0 \Rightarrow \sum_{z \in M} C^{rel}(y \rightarrow z)=1. \tag{7.2}$$

To satisfy condition 4 for all network members y, not only those for whom $\exists (x \in M)$ $C^{rel}(y \rightarrow x)>0$, a new condition has been appended to the final commitment function $C(y \rightarrow x)$.

The full set of conditions for commitment function $C(y \rightarrow x)$ in $VSN(M,R)$ is as follows:

1. Commitment function $C(y\rightarrow x)$ reflects the strength of the association from y to x in $VSN(M,R)$, $x,y\in M$, $x\neq y$ and for that reason if $C^{rel}(y\rightarrow x)>0$ then $C(y\rightarrow x)=C^{rel}(y\rightarrow x)$, where $C^{rel}(y\rightarrow x)$ is the value of association commitment directly derived from the data about association (activities) from y to x. If there is no association from y to x then $C^{rel}(y\rightarrow x)=C(y\rightarrow x)=0$, except condition 5.

2. The value of commitment is from the range [0,1]: $\forall(x,y\in M)\ C(y\rightarrow x)\in[0,1]$.

3. Commitment function to itself equals 0: $\forall(y\in M)\ C(y\rightarrow y)=0$.

4. The sum of all commitments has to equal 1, separately for each network member:

$$\forall(y\in M)\ \sum_{x\in M} C(y\rightarrow x)=1. \tag{7.3}$$

5. If a member y is not active to anybody, then some other members x are active to y, since based on Definition 7.1 no isolated members are allowed in $VSN(M,R)$, i.e. $\forall(y\in M)\ \exists(x\in M)\ \sum_{z\in M} C^{rel}(y\rightarrow z)=0 \Rightarrow C^{rel}(x\rightarrow y)>0$. In this case, to satisfy condition 4, Eq. (7.3), the sum 1 is distributed equally among all y's acquaintances - x (Fig. 7.3), i.e. all values of $C(y\rightarrow x)$:

$$\forall(x\in M)\sum_{z\in M} C^{rel}(y\rightarrow z)=0\Rightarrow$$

$$\forall(x\in M: C^{rel}(x\rightarrow y)>0)\ \ C(y\rightarrow x)=\frac{1}{card(\{x\in M: C^{rel}(x\rightarrow y)>0\})}. \tag{7.4}$$

In other words, the value of commitment function $C(y\rightarrow x)$ from y to x is usually obtained from raw data about direct activity of member y in relation to x or as the equal potential contribution in activity in case of the total lack of y's activity.

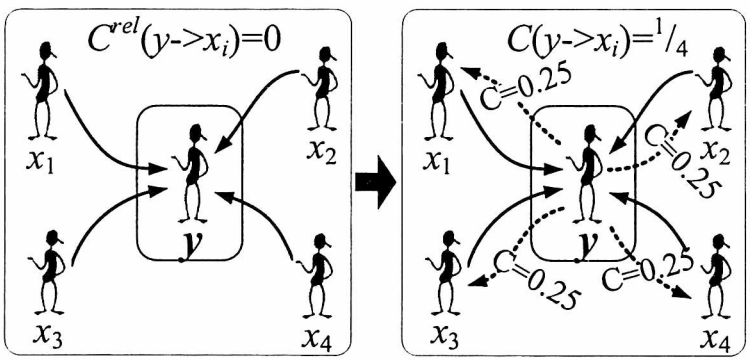

Fig. 7.3. Distribution of the commitment for an inactive member x equally among all x's acquaintances

Member y from Fig. 7.3 is not active to anybody within the network, but there are four members (x_1, x_2, x_3, x_4) who are active to user y. In this case, the commitment function is equally distributed among all y's acquaintances.

Note that the virtual social network $VSN(M,R)$ must not contain any isolated members (Definition 7.1). This restriction is derived from the lack of possibility to satisfy all conditions enumerated above for such members, especially condition 4, Eq. (7.3).

The consequence of the 4th constraint is that if member y is active to only one other member x, then $C(y \rightarrow x) = 1$.

According to the above requirements for the commitment function $C(y \rightarrow x)$, formula (7.1) can be expressed in a slightly modified way. Hence, social position function $SP(x)$ of individual x in $VSN=(M,R)$ respects only the values of social positions of direct member's x acquaintances as well as their activities in relation to x:

$$SP(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{i=1}^{m_x} SP(y_i) \cdot C(y_i \rightarrow x), \tag{7.5}$$

where y_i - x's acquaintances, i.e. the members who are in direct association to x: $C(y_i \rightarrow x) > 0$; m_x - the number of x's acquaintances.

The reduction of the number of elements in Eq. (7.5) compared to Eq. (7.1) can be important from the point of view of the implementation.

In general, social position refers to the social standing in the community. For that reason, it can be a crucial component of social capital that a human possesses in the network [Kaz06c].

7.2.3. Commitment Evaluation

To assess the strength of the association between two individuals x and y within the virtual social network the commitment function $C(y \rightarrow x)$ is used. It denotes the amount of the member y's activity that person y passes to member x and is easily derived from association commitment function $C^{rel}(y \rightarrow x)$, see Sec. 7.2.1.

The commitment $C^{rel}(y \rightarrow x)$ of member y within activity of their acquaintance x is directly evaluated from source data as the normalized sum of all contacts, cooperation, and communications from y to x in relation to all activities of y:

$$C^{rel}(y \rightarrow x) = \begin{cases} \dfrac{A(y \rightarrow x)}{\sum_{x \in M} A(y \rightarrow x)}, & \text{when} \sum_{x \in M} A(y \rightarrow x) > 0, \\ 0, & \text{when} \sum_{x \in M} A(y \rightarrow x) = 0, \end{cases} \tag{7.6}$$

where $A(y \to x)$ - the function that denotes the activity of person y directed to member x, e.g. number of emails sent by y to x; $A(y \to x) \geq 0$; m - the number of people within the virtual social network.

Note that according to requirement 3 for the commitment function (see Sec. 7.2.1) we need to ensure that $A(y \to y)=0$, i.e. emails sent to themselves are excluded.

As we can easily prove Eq. (7.6) fulfils also all other requirements for association commitment function. Note that there may exist some inactive members y in the network, for which $\sum_{x \in M} A(y \to x) = 0$ and in consequence $\sum_{x \in M} C^{rel}(y \to x) = 0$. Such inactive members y are additionally processed through transformation from $C^{rel}(y \to x)$ to $C(y \to x)$, see Sec. 7.2.1, condition 5, Eq. (7.4), in order to fulfil the fourth condition, Eq. (7.3).

The presence of the time is not considered in the formula (7.6). Similar approach is utilized by Valverde *et al.*, where the strength of the associations is established by the number of emails sent to a person in the group [Val06]. However, the authors do not respect the general activity of the given individual. This general, local activity exists in the form of denominator in Eq. (7.6).

In another version of association commitment function $C^{rel}(y \to x)$, all member's activities are considered with respect to their time. The entire time from the first to the last activity of any member is divided into k periods. For instance, a single period can be a month. Activities in each period are considered separately for each individual:

$$C^{rel}(y \to x) = \begin{cases} \dfrac{\sum_{i=0}^{k-1} (\lambda)^i \cdot A_i(y \to x)}{\sum_{x \in M} \sum_{i=0}^{k-1} (\lambda)^i \cdot A_i(y \to x)}, & \text{when } \sum_{x \in M} \sum_{i=0}^{k-1} (\lambda)^i \cdot A_i(y \to x) > 0, \\ 0, & \text{when } \sum_{x \in M} \sum_{i=0}^{k-1} (\lambda)^i \cdot A_i(y \to x) = 0, \end{cases} \tag{7.7}$$

where i - the index of the period: for the most recent period $i=0$, for the previous one: $i=1$, ..., for the earliest one $i=k-1$; $A_i(y \to x)$ - the function that denotes the activity level of person y directed to member x in the ith time period, e.g. number of emails sent by y to x in the ith period; $(\lambda)^i$ - the exponential function that denotes the weight of the ith time period, $\lambda \in (0;1]$; k - the number of time periods.

The activity of person y is calculated in every time period and after that the appropriate weights are assigned to the particular time periods, using $(\lambda)^i$ factor. The earliest period $(\lambda)^i=\lambda^0=1$, for the previous one $(\lambda)^i=\lambda^1=\lambda$ is not greater than 1, and for the most former period $(\lambda)^i=\lambda^{k-1}$ attains the smallest value. For example, if one year's dataset is proceeded and a period is a month then $k=12$. For $\lambda=0.9$, the

data from January is considered with the factor $0.9^{11}=0.31$, for February we have $0.9^{10}=0.35$, ..., for October $0.9^{2}=0.81$, for November - 0.9 and finally for December $0.9^{0}=1$. This in a sense is similar to an idea which was used in the personalized systems to weaken older activities of recent users [Kaz07a].

One of the activity types is the communication via chat. In this case, $A_i(y \rightarrow x)$ is the number of chats that are common for x and y in the particular period i; and $\sum_{x \in M} A_i(y \rightarrow x)$ is the number of all chats in which y took part in the ith period. If person y had many common chats with x in comparison to the number of all y's chats, then x has greater commitment within activities of y, i.e. $C^{rel}(y \rightarrow x)$ will have greater value and in consequence the social position of member x will grow.

Note that $C^{rel}(y \rightarrow x)$ will have value 1 when member x is the only interlocutor of person y.

However, not all of the elements can be calculated in such a simple way. Other activities are much more complex, e.g. comments on forums or blogs. Each forum consists of many threads where people can submit their comments. In this case, $A_i(y \rightarrow x)$ is the number of user y's comments in the threads in which x has also commented, in period i, whereas sum $\sum_{x \in M} A_i(y \rightarrow x)$ is the total number of comments that have been made by all x who are y's friends on these threads, at the same time.

7.2.4. Social Position Calculation, the SPIN Algorithm

The social position is calculated in the iterative way, which means that the left-hand side of Eq. (7.1) is the result of iteration while the right-hand side is the input:

$$SP_{n+1}(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{y \in M} SP_n(y) \cdot C(y \rightarrow x), \tag{7.8}$$

where $SP_{n+1}(x)$ and $SP_n(x)$ are the social positions of member x after the n+1st and nth iteration, respectively.

To perform the first iteration, we also need to have an initial value of social position $SP_0(x)$ for all $x \in M$:

$$SP_1(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{y \in M} SP_0(y) \cdot C(y \rightarrow x). \tag{7.9}$$

Since the calculations are iterative, we also need to introduce a stop condition. For this purpose, a fixed precision coefficient τ is used. Thus, the calculation is stopped when the following criterion is met:

$$\forall (x \in M) \ | SP_n(x) - SP_{n-1}(x) | \leq \tau. \tag{7.10}$$

The SPIN algorithm

Input: D – data about communication, interaction or common activities between members M in the virtual social network $VSN=(M,R)$.
$SP_0=<SP_0(x_1), SP_0(x_2), \ldots, SP_0(x_m)>$ – the vector of initial social positions, $m=\text{card}(M)$.
ε – coefficient from Eq. (7.1), $\varepsilon \in [0,1]$.
τ – stop condition, i.e. the precision coefficient, e.g. $\tau=0.00001$.

Output: $SP_{next}=<SP(x_1), SP(x_2), \ldots, SP(x_m)>$ – the vector of final social positions.
Rank – the ranking of individuals from M.
n – the number of iterations.

```
    /* commitment evaluation */
 1. for (each pair x,y∈M ) do
 2.   evaluate C^rel[x,y] from D, e.g. use Eq. (7.6) or (7.7)
 3. for (each member x∈M) do {
 4.   commitment_of_x = 0
 5.   acquaintances_of_x = 0
 6.   for (each member y∈M) do {
 7.    commitment_of_x = commitment_of_x+C^rel[x,y]
 8.    if (C^rel[y,x]>0) then
 9.      acquaintances_of_x = acquaintances_of_x+1
10.   }
11.  for (each member y∈M) do
12.    if (C^rel[x,y]>0) then
13.      C[x,y] = C^rel[x,y]
14.    else
15.      if (commitment_of_x=0 and C^rel[y,x]>0) then
16.       C[x,y] = 1/acquaintances_of_x /* cond. 5, Eq.(7.4) */
17.      else
18.        C[x,y] = 0
19.  }
    /* social position estimation */
20. n = 0
21. SP_prev = SP_0
22. repeat
23.   for (each member x from M) do {
24.     SP_next[x] = (1-ε)
25.     for (each member y from M) do
```

```
26.          SP_next[x] = SP_next[x]+ε*SP_prev[y]*C[y,x]
27.      }
28.      SP_prev = SP_next
29.      n = n+1
30.  until stop cond. Eq. (7.10) is fulfilled for all members
31.  create ranking list Rank based on SP_next, see Sec. 7.5.1
32.  }
```

Fig. 7.4. The SPIN algorithm for the iterative evaluation of social positions in the social network

Obviously, another version of the stop condition can also be applied, e.g.:

$$|SSP_n - SSP_{n-1}| \leq \tau,$$

where SSP_n and SSP_{n-1} are the sums of all social positions after the nth and $(n-1)$th iteration, respectively.

Based on Eq. (7.8), (7.9) and (7.10) we can develop the SPIN algorithm (*Social Position In the Network*), Fig. 7.4.

The SPIN algorithm can slightly accelerate by usage of database indexes and limitation of the internal loop (lines 25–26) only to member x's acquaintances, i.e. those y for whom $C[y,x]>0$, see Eq. (7.5).

The convergence of the calculation process is proved by Theorem 7.3, see Sec. 7.3.2.

7.2.5. Example of Social Network

An example of social network is presented in Fig. 7.5. The arc values indicate the values of commitments function $C(y \rightarrow x)$ from member y to x.

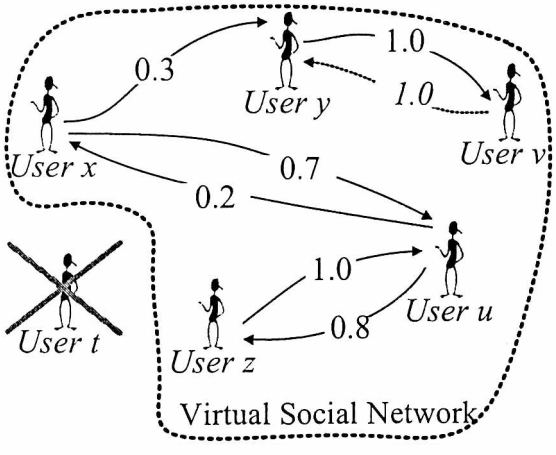

Fig. 7.5. The human society with the extracted virtual social network

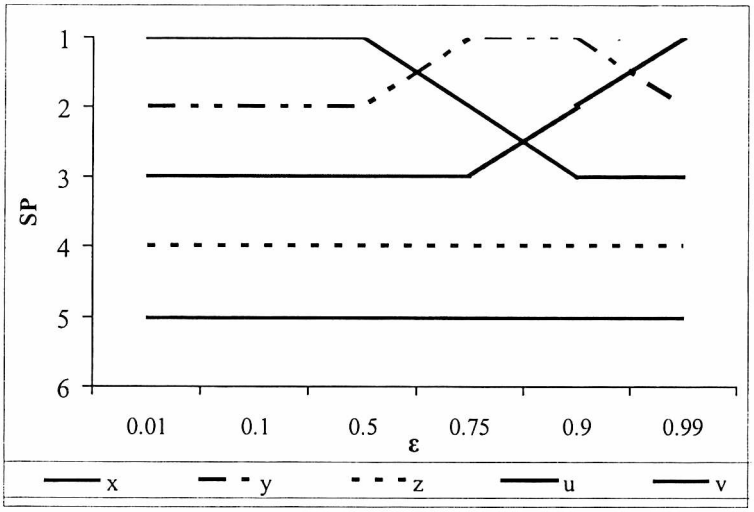

Fig. 7.6. The ranking positions based on social position in relation to ε

Table 7.1. Social positions for the social network from Fig. 7.4

ε	0.01		0.1		0.5		0.75		0.9		0.99	
	Value	Rank	Value	Rank	Value	Rank	Value	Rank	Value	Rank	Value	Rank
SP(x)	0.99201	5	0.92126	5	0.62091	5	0.43103	5	0.29086	5	0.07750	5
SP(y)	1.00297	2	1.02791	2	1.12418	2	1.22167	1	1.41333	1	2.15660	1
SP(z)	0.99805	4	0.98503	4	0.9836	4	0.97414	4	0.86345	4	0.27998	4
SP(u)	1.00692	1	1.06299	1	1.20915	1	1.20690	2	1.06035	3	0.34089	3
SP(v)	1.00003	3	1.00279	3	1.06209	3	1.16626	3	1.37199	2	2.14503	2
SP_{max}/ SP_{min}	1.0150		1.1538		1.9474		2.8000		3.6455		4.3988	
Std dev	0.0056		0.0529		0.2275		0.3332		0.4567		1.0551	
Avg. SP	1		1		1		1		1		1	
Sum of SP	5		5		5		5		5		5	

Note that member t has been excluded from the processed social network since no other member has an association to t and neither member t possesses any outgoing associations. In the original social network presented in Fig. 7.5, member v was not active with anyone within the network but member y was active to v. For that reason, the full member v's commitment is passed to member y (a dotted arrow) to satisfy condition 4, Eq. (7.3), according to condition 5, Eq. (7.4).

The initial social position is established as follows: $SP_0(x)=SP_0(y)=SP_0(z)=SP_0(u)=SP_0(v)=1$. The final calculated values of social position as well as ranking

index for each individual of the community from Fig. 7.5 are presented in Table 7.1. The ranking was based on the descending order of social positions. Note that both of the outcomes of the social position calculations and ranking order vary depending on the value of ε (see Fig. 7.6). The ranking for the first three positions, for $\varepsilon \leq 0.5$ is (u, y, v) and differs from that for $\varepsilon=0.75$ (y, u, v) - y changed place with u; and for $\varepsilon=\geq 0.9$: (y, v, u) - u swapped with v.

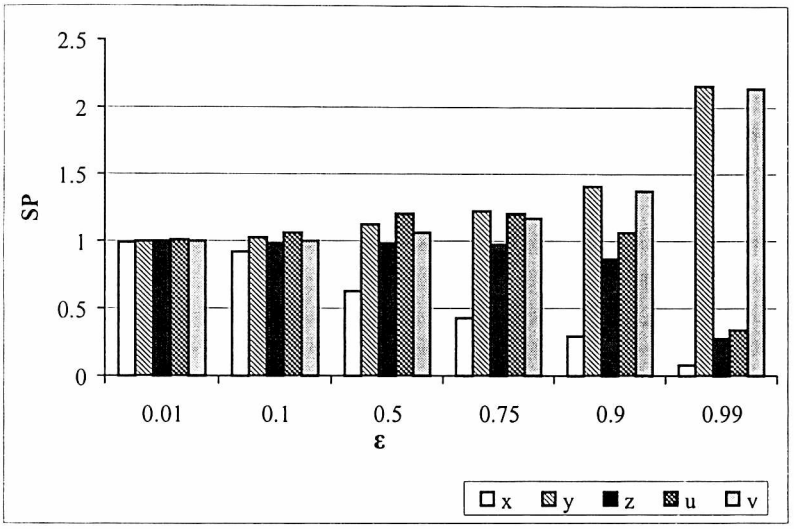

Fig. 7.7. The distribution of social positions within the social network in relation to ε

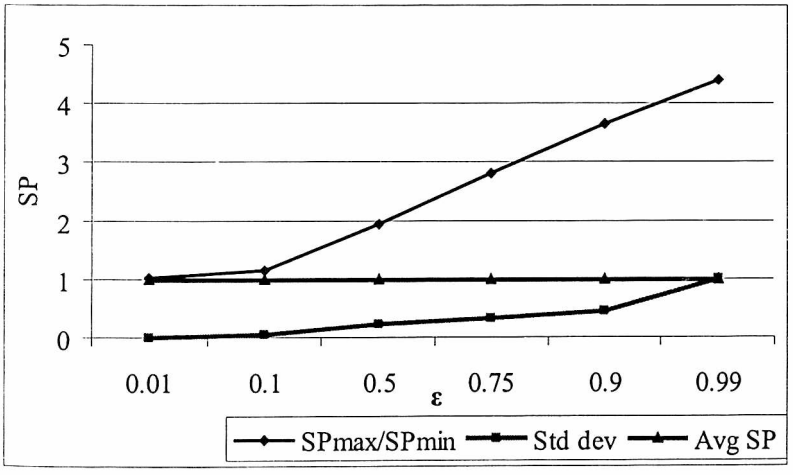

Fig. 7.8. The ratio SPmax/SPmin, average, and standard deviation of social positions calculated for the community from Fig. 7.5 in relation to ε

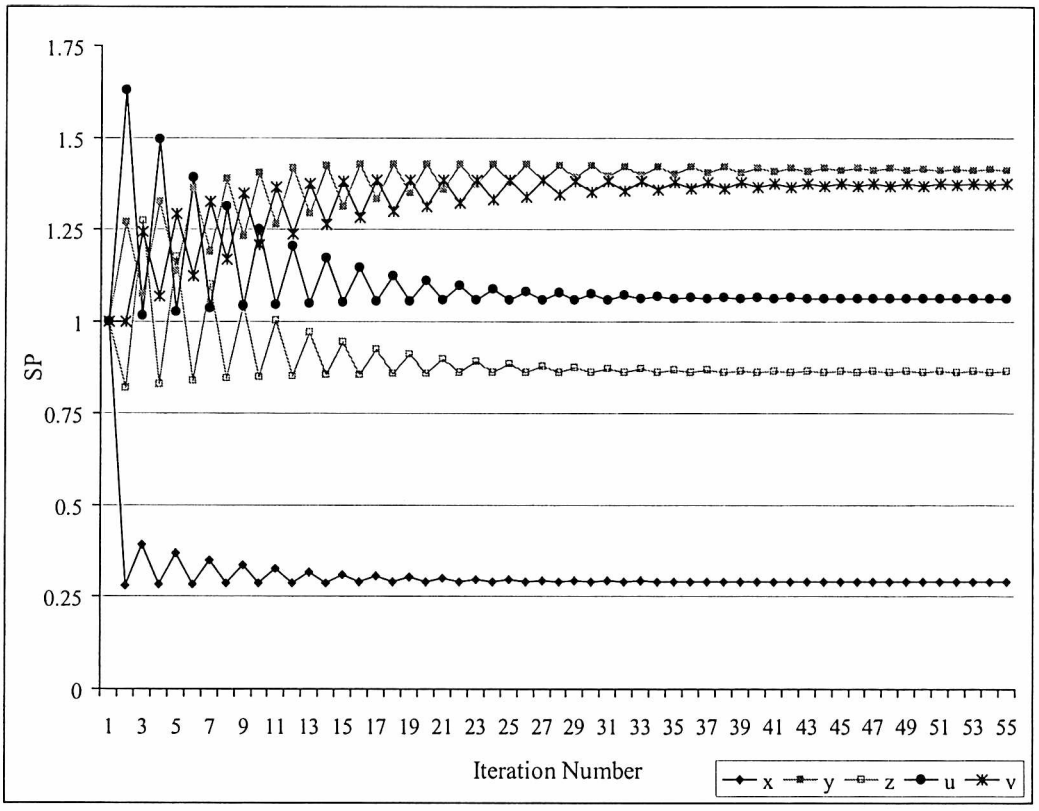

Fig. 7.9. The values of social positions after every iteration, ε=0.9

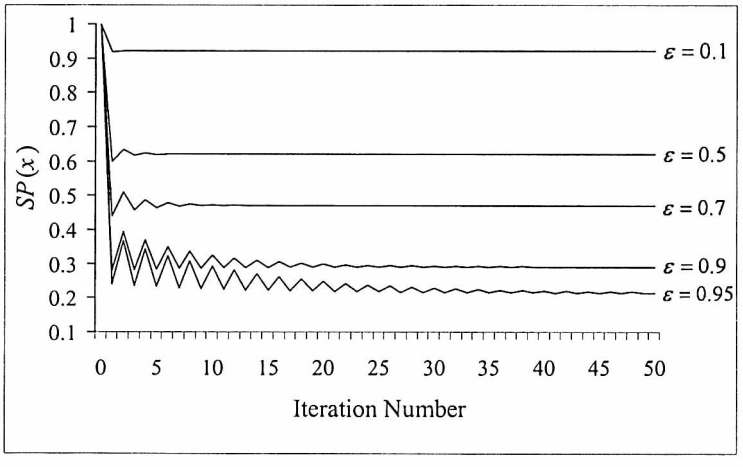

Fig. 7.10. The values of the member x's social position in relation to the number of iterations for various ε

Table 7.2. The initial social positions for set SP_01 and SP_02

	SP_01	SP_02
$SP_0(x)$	0.5	2
$SP_0(y)$	0.1	2
$SP_0(z)$	0.9	2
$SP_0(u)$	0.5	2
$SP_0(v)$	0.4	2
Total	2.4	10

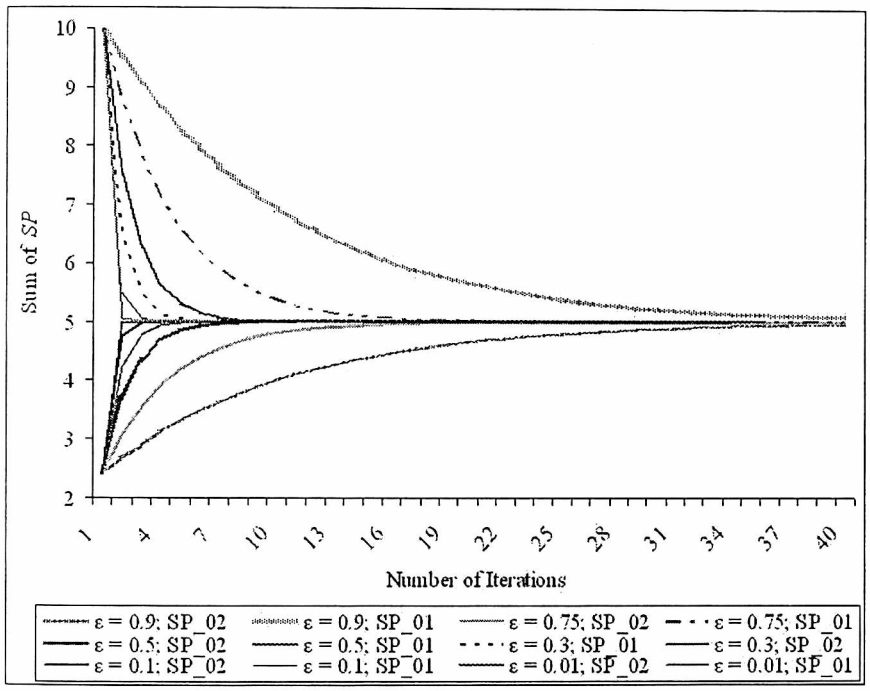

Fig. 7.11. The convergence of the social position sum for various ε and various initial sums

A comparison of social position values, which vary depending on the value of ε, is presented in Fig. 7.7. It can be noticed that the distribution of social position values increases with ε. Some additional information about the influence of the coefficient ε upon the members' social positions provides the average social position and the standard deviation (Table 7.1 and Fig. 7.8). If ε is greater, the distance between the minimum and maximum social position within human community increases. Moreover, the average social position always equals 1 and the sum of all social positions is 5, i.e. the number of members, regardless of the value of ε

(Table 7.1). This is one of the features of social position function – see Sec. 7.3, Theorem 7.1. However, the standard deviation differs depending on the coefficient ε value. The greater the value of ε, the bigger the standard deviation is (Fig. 7.8). Hence, the distribution of social position can be tailored with ε. Nevertheless, simultaneously the coefficient ε also slightly influences the ranking (Fig. 7.6).

The values of social position for every user in consecutive iterations, starting with 1 as the initial values of *SP* are shown in Fig. 7.9. The calculations have been performed for ε=0.9. The chart in Fig. 7.9 reveals that the algorithm used for evaluation is convergent, see Sec. 7.3, Theorem 7.3, for the formal proof. Additionally, the algorithm tends to converge faster for smaller ε rather than for the greater one (Fig. 7.10).

The studies performed for the network of Fig. 7.5 disclosed that the sum of all social positions is convergent to the number of people within the social network, i.e. to five (see Theorem 7.1). Two separate sets of initial social position values for the social network from Fig. 7.5 are presented in Table 7.2. The pace of convergence both for individual members (Fig. 7.10) and for their sum (Fig. 7.11) crucially depends on ε value and it is greater for smaller ε.

7.3. Features of Social Position

The social position is a new measure which possesses several interesting features: fixed limit of sum and average, convergence of social position iterative calculation as well as the main factors of this convergence.

7.3.1. Total and Average Social Position

First, let us focus on the total and the average value of social positions within the entire network.

Lemma 7.1. For every natural number $n \geq 0$, we have:

$$SSP_{n+1} = m \cdot (1 - \varepsilon) + \varepsilon \cdot SSP_n,$$

where $SSP_n = \sum_{i=1}^{m} SP_n(x_i)$ is the sum of all social positions after the nth iteration; $SSP_0 = \sum_{i=1}^{m} SP_0(x_i)$ is the sum of all initial social positions; m is the number of nodes, i.e. m=card(M).

Note that initial social positions SP_0 can have any positive, real values.

Proof. $SSP_{n+1} = \sum_{i=1}^{m} SP_{n+1}(x_i)$ and according to Eq. (7.8) and (7.9):

$$SSP_{n+1} = \sum_{i=1}^{m}\left((1-\varepsilon) + \varepsilon \cdot \sum_{j=1}^{m} SP_n(y_j)\cdot C(y_j \to x_i)\right) =$$

$$= m\cdot(1-\varepsilon) + \sum_{i=1}^{m}\left(\varepsilon \cdot \sum_{j=1}^{m} SP_n(y_j)\cdot C(y_j \to x_i)\right) =$$

$$= m\cdot(1-\varepsilon) + \varepsilon\cdot\left(\sum_{j=1}^{m}\left(SP_n(y_j)\cdot \underbrace{\sum_{i=1}^{m} C(y_j \to x_i)}_{=1}\right)\right) = m\cdot(1-\varepsilon) + \varepsilon\cdot\left(\sum_{j=1}^{m}\left(SP_n(y_j)\right)\right).$$

Thus, $SSP_{n+1} = m\cdot(1-\varepsilon) + \varepsilon\cdot SSP_n$. Q.E.D.

Lemma 7.2. For every natural number n, we have:

$SSP_n = m\cdot(1-\varepsilon^n) + \varepsilon^n\cdot SSP_0$,

where $SSP_0 = \sum_{x\in M} SP_0(x)$ is the sum of all initial social positions.

Proof

i) For $n = 1$, we have $SSP_1 = m\cdot(1-\varepsilon) + \varepsilon\cdot SSP_0$, which is true due to Lemma 7.1.

ii) Assume that the statement is true for $n = k$, i.e.:

$SSP_k = m\cdot(1-\varepsilon^k) + \varepsilon^k\cdot SSP_0$.

We want to prove it for $n = k+1$, i.e.:

$SSP_{k+1} = m\cdot(1-\varepsilon^{k+1}) + \varepsilon^{k+1}\cdot SSP_0$.

Indeed, by Lemma 7.1:

$$SSP_{k+1} = m\cdot(1-\varepsilon) + \varepsilon\cdot SSP_k = m\cdot(1-\varepsilon) + \varepsilon\cdot(m\cdot(1-\varepsilon^k) + \varepsilon^k\cdot SSP_0) =$$
$$= m\cdot((1-\varepsilon) + \varepsilon\cdot(1-\varepsilon^k)) + \varepsilon\cdot\varepsilon^k\cdot SSP_0 = m\cdot(1-\varepsilon^{k+1}) + \varepsilon^{k+1}\cdot SSP_0.$$

According to mathematical induction the statement $SSP_n = m\cdot(1-\varepsilon^n) + \varepsilon^n\cdot SSP_0$ is true for all natural numbers n. Q.E.D.

Theorem 7.1

i) For $\varepsilon\in[0,1)$, the sum of all social positions SSP in the virtual social network $VSN=(M,R)$ is convergent to the number of all members in the network:

$\lim_{n\to\infty}(SSP_n)=m$. As a result, the average social position in the network is convergent to 1.

ii) For $\varepsilon=1$ and all natural numbers n we have $SSP_n=SSP_0$, where $SSP_0=\sum_{i=1}^{m} SP_0(x_i)$ is the sum of all initial social positions.

Proof

i) From Lemma 7.2, for $\varepsilon\in[0,1)$, we have:

$$\lim_{n\to\infty}(SSP_n) = \lim_{n\to\infty}(m\cdot(1-\varepsilon^n)+\varepsilon^n\cdot SSP_0) = m.$$

ii) From Lemma 7.2, for $\varepsilon=1$, we have

$SSP_n = m\cdot(1-\varepsilon^n)+\varepsilon^n\cdot SSP_0 = SSP_0$, for every n. Q.E.D.

Note that for $\varepsilon=0$ the social position of every network member $\forall(x\in M)$ $SP(x)=1$ whereas for $\varepsilon=1$ the sum of all social positions always equals its initial value.

7.3.2. Convergence of Calculation

Convergence is the essential feature of every iterative algorithm. In the approach under consideration, it regards both the sum of all social positions and the social position of each individual.

Theorem 7.2. If $\varepsilon\in(0;1)$ and initial sum of social positions SSP_0 is different from limit m, i.e. $|SSP_0-m|>0$, then the less the value of ε is, the faster the sum of social positions is convergent to its limit $m=card(M)$.

Proof. The pace of convergence indicates after how many n iterations the value of $|SSP_n-m|$ becomes less than the given error level τ, $0<\tau<|SSP_0-m|$, i.e. when $|SSP_n-m|<\tau$.

From Lemma 7.2:

$$|SSP_n-m| = |m\cdot(1-\varepsilon^n)+\varepsilon^n\cdot SSP_0-m| = |\varepsilon^n\cdot SSP_0-m\cdot\varepsilon^n| = \varepsilon^n\cdot|SSP_0-m|<\tau.$$

$$\varepsilon^n<\frac{\tau}{|SSP_0-m|}<1.$$

$$n>\log_\varepsilon\frac{\tau}{|SSP_0-m|}>0.$$

The closer the value of ε to 0, the lesser the value of n. Q.E.D.

Hence, we need more iterations for the greater ε, or when we want the more precise results (the lower error level τ) or for the greater differences between the initial sum SSP_0 and its limit m.

This conclusion has been confirmed by experiments, see Fig. 7.11.

Lemma 7.3. If the initial values of social positions are non-negative, then social positions have an inferior and superior limit after every iteration: $\forall(n>0)\forall x$ $SP_n(x)\in[(1-\varepsilon), A]$, where $A=\max(SSP_0,m)$.

Proof

i) The inferior limit

From Eq. (7.8), by induction, we prove that $\forall n\ SP_n(x)\geq 0$.

Next, also from Eq. (7.8) we have:

$$SP_{n+1}(x) = (1-\varepsilon)+\varepsilon\cdot\sum_{y\in M}\underbrace{SP_n(y)}_{\geq 0}\cdot\underbrace{C(y\to x)}_{\geq 0} \geq (1-\varepsilon)$$

ii) The superior limit

Since all initial values of social positions are non-negative then $SSP_0\geq 0$.

From Lemma 7.2:

$$SSP_n = m\cdot(1-\varepsilon^n)+\varepsilon^n\cdot SSP_0 \leq \max(m,SSP_0)\cdot(1-\varepsilon^n)+\varepsilon^n\cdot\max(m,SSP_0) =$$
$$= \max(m,SSP_0) = A.$$

$SSP_n \leq A$ so any of the non-negative components of SSP_n, i.e. $SP_n(x)$, must not exceed A.

Hence, $SP_n(x)\in[(1-\varepsilon),A]$. Q.E.D.

Lemma 7.3 reveals that there is a fixed limit inferior and limit superior for every social position value, after every iteration and these limits are independent of iteration n.

Lemma 7.4. If initial values of social positions are non-negative, then $|SP_{n+k+1}(x) - SP_{n+k}(x)| \leq \varepsilon^k \cdot A\cdot m$.

Proof. Based on Eq. (7.8):

$$|SP_{n+k+1}(x) - SP_{n+k}(x)| = \varepsilon\cdot\left|\sum_{y_1\in M}(SP_{(n+k+1)-1}(y_1) - SP_{(n+k)-1}(y_1))\cdot C(y_1\to x)\right| \leq$$
$$\leq \varepsilon\cdot\sum_{y_1\in M}\left|SP_{(n+k+1)-1}(y_1) - SP_{(n+k)-1}(y_1)\right|\cdot C(y_1\to x) \leq$$

$$\leq \varepsilon^2 \cdot \sum_{y_1 \in M} \sum_{y_2 \in M} \left| SP_{(n+k+1)-2}(y_2) - SP_{(n+k)-2}(y_2) \right| \cdot C(y_1 \to x) \cdot C(y_2 \to y_1) \leq$$

$$\leq \ldots \leq$$

$$\leq \varepsilon^k \cdot \sum_{y_1 \in M} \sum_{y_2 \in M} \ldots \sum_{y_k \in M} \left| SP_{(n+k+1)-k}(y_k) - SP_{(n+k)-k}(y_k) \right| \cdot$$

$$\cdot C(y_1 \to x) \cdot C(y_2 \to y_1) \cdot \ldots \cdot C(y_k \to y_{k-1}) \leq$$

$$\leq \varepsilon^k \cdot \sum_{y_k \in M} \underbrace{SP_{(n+k+1)-k}(y_k) - SP_{(n+k)-k}(y_k)}_{\leq A, \text{Lemma 3}} \cdot \underbrace{C(y_1 \to x)}_{\leq 1} \leq$$

$$\leq \varepsilon^k \cdot A \cdot m .$$

The last but one inequality results from formula (7.3), i.e. $\sum_{y_{k-1} \in M} C(y_k \to y_{k-1}) = 1$, that has been applied k–1 times. Q.E.D.

Note that according to Lemma 7.4 the pace of convergence depends on both the value of ε and the number of network members. It means that for smaller ε and smaller networks the fixed difference between two consecutive iterations will be reached faster rather than for greater ε and larger networks. These results have been confirmed by experiments (Fig. 7.9, 7.10, and 7.23). A similar conclusion can be drawn from Theorem 7.2 in relation to the sum of all social positions.

Theorem 7.3. If initial values of social positions are non-negative, then their calculation based on Eq. (7.8) is convergent, that is, $\exists SP(x) = \lim_{n \to \infty}(SP_n(x))$.

Proof. By Lemma 7.4.

$$|SP_{n+k+l}(x) - SP_{n+k}(x)| \leq$$

$$\leq |SP_{n+k+l}(x) - SP_{n+k+l-1}(x)| + |SP_{n+k+l-1}(x) - SP_{n+k+l-2}(x)| + \ldots + |SP_{n+k+1}(x) - SP_{n+k}(x)| \leq$$

$$\leq A \cdot m \cdot (\varepsilon^{k+l-1} + \varepsilon^{k+l-2} + \ldots + \varepsilon^k) = A \cdot m \cdot \varepsilon^k \cdot (1 + \varepsilon + \varepsilon^2 + \ldots + \varepsilon^{l-1}) =$$

$$= A \cdot m \cdot \varepsilon^k \cdot \frac{1 - \varepsilon^l}{1 - \varepsilon} \xrightarrow{k \to \infty} 0 .$$

Hence, $\{SP_n(x)\}$ is a Cauchy sequence, therefore it is convergent. Q.E.D.

Theorem 7.3 assumes non-negative values of initial social position. However, similar reasoning can also be performed for negative values. In this case, each value should be split into non-negative and negative parts. From a practical point of view, the assignment of negative initial values to social position appears to be useless since the final social positions are positive and this can only increase the number of necessary iterations.

Another approach to the proof of convergence based on the concept of power series, concerning a similar problem, i.e. PageRank, was presented in [Bri06].

7.3.3. Interval of Limit Values

Regardless of the initial social position values, their limit values have to be from the range of the given interval.

Theorem 7.4. The limit value of social position does not exceed half of the number of members:

$$\forall(x \in M) \lim_{n \to \infty} (SP_n(x)) \leq m/2.$$

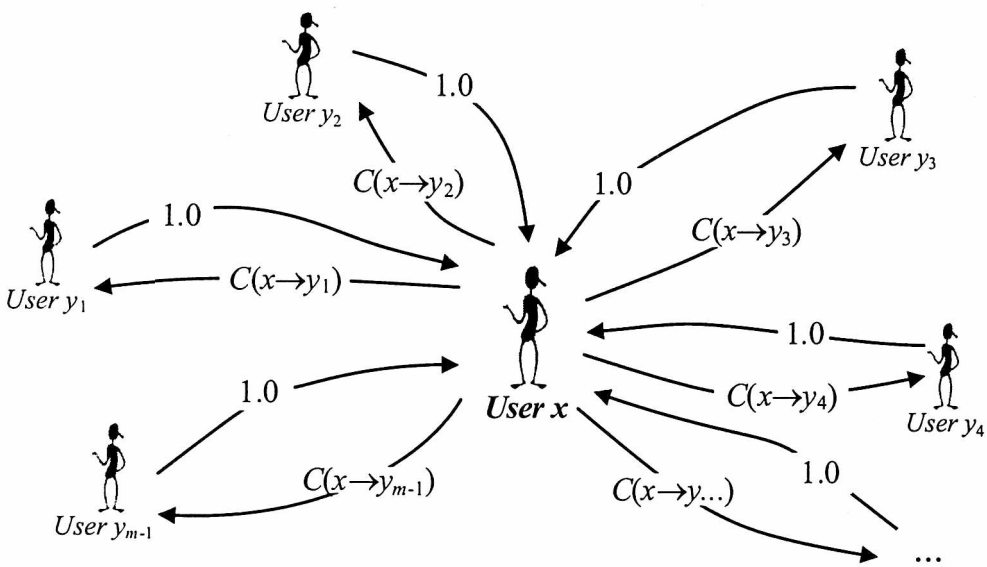

Fig. 7.12. The community where individual x has the greatest social position

The member x will have the greatest social position if all other members pass the whole of their commitment to the person x, i.e. $\forall(y \in M, y \neq x)\ C(y \to x)=1$, and the member x's commitment will be spread among all x's acquaintances, i.e. $\forall(y \in M, y \neq x)\ C(x \to y)>0$ (Fig. 7.12). Moreover, it is not important how the central member x's commitment is distributed. In other words, member x reaches the greatest social position if member x gathers all commitments from all members y in the social network, i.e. fully inherits social positions of all y.

Proof

$SP_{\max}(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{i=1}^{m-1} SP(y_i) \cdot C(y_i \to x)$, where $SP_{\max}(x)$ is the maximum value of limit $\lim_{n\to\infty}(SP_n(x))$.

$$SP_{\max}(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{i=1}^{m-1} \left((1-\varepsilon) + \varepsilon \cdot SP_{\max}(x) \cdot C(x \to y_i)\right) \cdot \underbrace{C(y_i \to x)}_{1}.$$

$$SP_{\max}(x) = (1-\varepsilon) + \varepsilon \cdot \sum_{i=1}^{m-1} \left[(1-\varepsilon) + \varepsilon \cdot SP_{\max}(x) \cdot C(x \to y_i)\right].$$

$$SP_{\max}(x) = (1-\varepsilon) + \varepsilon \cdot \left((m-1)\cdot(1-\varepsilon) + \varepsilon \cdot SP_{\max}(x) \cdot \underbrace{\sum_{i=1}^{m-1} C(x \to y_i)}_{1} \right).$$

$$SP_{\max}(x) = (1-\varepsilon) + \varepsilon \cdot \left((m-1)(1-\varepsilon) + \varepsilon \cdot SP_{\max}(x)\right).$$

$$SP_{\max}(x) = (1-\varepsilon) + \varepsilon \cdot \left(m - m\cdot\varepsilon - 1 + \varepsilon + \varepsilon \cdot SP_{\max}(x)\right).$$

$$SP_{\max}(x) = 1 - \varepsilon + \varepsilon \cdot m - \varepsilon^2 \cdot m - \varepsilon + \varepsilon^2 + \varepsilon^2 \cdot SP_{\max}(x).$$

$$(1-\varepsilon^2) \cdot SP_{\max}(x) = (1-\varepsilon) + m \cdot \varepsilon(1-\varepsilon) - \varepsilon(1-\varepsilon).$$

$$(1-\varepsilon^2) \cdot SP_{\max}(x) = (1-\varepsilon) \cdot (1 + m \cdot \varepsilon - \varepsilon).$$

$$SP_{\max}(x) = \frac{(1-\varepsilon)\cdot(1+m\cdot\varepsilon-\varepsilon)}{(1-\varepsilon)\cdot(1+\varepsilon)}.$$

$$SP_{\max}(x) = \frac{1-\varepsilon+m\cdot\varepsilon}{1+\varepsilon}.$$

The social position is maximal when the function $f(\varepsilon) = \frac{1-\varepsilon+m\cdot\varepsilon}{1+\varepsilon}$ reaches its maximum value. The domain of this function is ε = [0,1] and $m \geq 2$. This is the constraint derived from the formula that serves to calculate the social position of the member of the community.

First, the monotonic of the function $f(\varepsilon)$ is studied. This function is non-decreasing, which is proved below.

A function $f(x)$ is said to be non-decreasing in an interval I if $f(b) \geq f(a)$ for all $b>a$, where $a, b \in I$ [Jef88].

$$\varepsilon_2 - \varepsilon_1 > 0 \Rightarrow f(\varepsilon_2) - f(\varepsilon_1) \geq 0.$$

$$f(\varepsilon_2) - f(\varepsilon_1) = \frac{1-\varepsilon_2+\varepsilon_2\cdot m}{1+\varepsilon_2} - \frac{1-\varepsilon_1+\varepsilon_1\cdot m}{1+\varepsilon_1}.$$

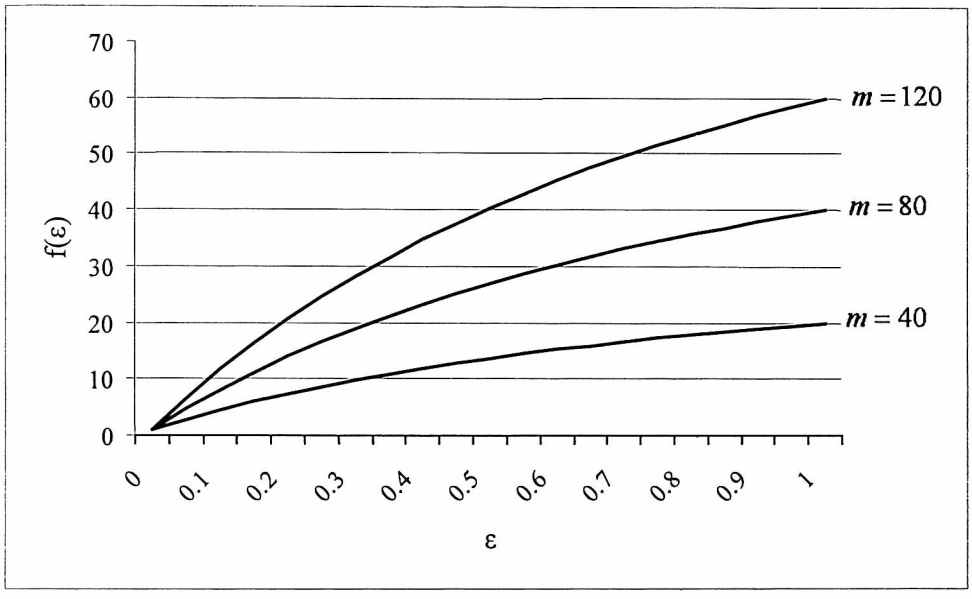

Fig. 7.13. The chart of the function $f(\varepsilon)$ for the social network that consists of m persons

$$f(\varepsilon_2)-f(\varepsilon_1)=\frac{(1-\varepsilon_2+\varepsilon_2\cdot m)\cdot(1+\varepsilon_1)-(1-\varepsilon_1+\varepsilon_1\cdot m)\cdot(1+\varepsilon_2)}{(1+\varepsilon_2)\cdot(1+\varepsilon_1)}.$$

$$f(\varepsilon_2)-f(\varepsilon_1)=$$

$$=\frac{1-\varepsilon_1\cdot\varepsilon_2-\varepsilon_2+\varepsilon_1+\varepsilon_2\cdot m+\varepsilon_1\cdot\varepsilon_2\cdot m}{(1+\varepsilon_2)\cdot(1+\varepsilon_1)}-$$

$$-\frac{-1-\varepsilon_2+\varepsilon_1+\varepsilon_1\cdot\varepsilon_2-\varepsilon_1\cdot m-\varepsilon_1\cdot\varepsilon_2\cdot m}{(1+\varepsilon_2)\cdot(1+\varepsilon_1)}.$$

$$f(\varepsilon_2)-f(\varepsilon_1)=\frac{-2\cdot\varepsilon_2+2\cdot\varepsilon_1+\varepsilon_2\cdot m-\varepsilon_1\cdot m}{(1+\varepsilon_2)\cdot(1+\varepsilon_1)}.$$

$$f(\varepsilon_2)-f(\varepsilon_1)=\frac{\overbrace{-(2-m)}^{<0\ \forall(m\geq 2)}\cdot\overbrace{(\varepsilon_2-\varepsilon_1)}^{>0}}{\underbrace{(1+\varepsilon_2)\cdot(1+\varepsilon_1)}_{>0\ \forall(\varepsilon>0)}}.$$

$$\forall(\varepsilon\in[0,1])\ f(\varepsilon_2)-f(\varepsilon_1)\geq 0.$$

This means that the function $f(\varepsilon)$ is non-decreasing, so it reaches the maximum value for $\varepsilon = 1$ and then $f(\varepsilon)=\frac{m}{2}$. This leads to the conclusion that $SP_{\max}(x)=\frac{m}{2}$ (Fig. 7.13). Q.E.D.

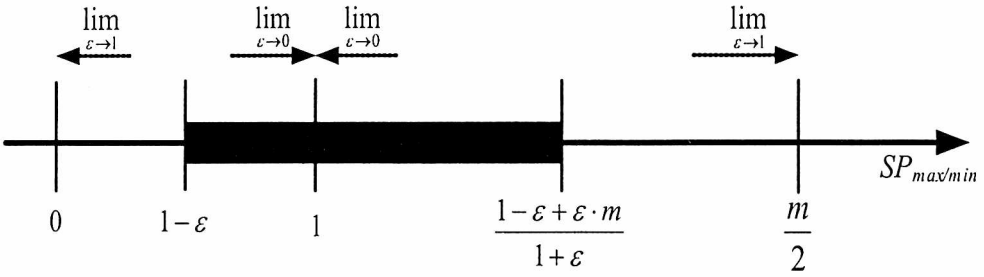

Fig. 7.14. The range of the social position values

The interval of social position depends on the number of members within the community m and the value of the coefficient ε (Fig. 7.14). In general, the limit value of social position is from the range $\left[1-\varepsilon, \frac{1-\varepsilon+\varepsilon \cdot m}{1+\varepsilon}\right]$, see also Lemma 7.3. The maximum value of the social position is reached for ε=1 and in such cases social position equals $\frac{m}{2}$, where m is the number of members within the community.

7.4. Other Centrality Measures

Although the definition of the virtual social network (Definition 7.1) differs from that presented by other researchers, the measures which are utilized in regular social network analysis [Was94] can also be easily applied in the virtual social network case. Hitherto prevailing approaches to social network analysis have provided many measures to determine the characteristics of a member within the network like centrality, prestige, reachability, connectivity [Car05, Han06, Was94]. All of them indicate the importance of a member in the network. The first two: centrality and prestige are analyzed and compared to the new measure proposed - social position in the experimental studies, see Sec. 7.5.

There are some particular approaches to evaluate one of the above mentioned measures - centrality: closeness centrality, betweenness centrality, and degree centrality [Fre79].

Degree centrality $DC(x)$ of member x takes into account the outdegree number of member x [Pro51, Sha54]. It is expressed by the normalized number of neighbours that are adjacent to the given person as follows:

$$DC(x) = \frac{o(x)}{m-1}, \tag{7.11}$$

where $o(x)$ - the number of the first level neighbours that are adjacent to x; m - the total number of members in the social network.

The closeness centrality pinpoints how close is a member to all the others within the social network [Bav50]. Its main idea is that the member takes the central position if they can quickly contact other members in the network. A similar idea was studied for hypertext systems [Bot92]. The closeness centrality $CC(x)$ of member x strongly depends on the geodesic distance, i.e. the shortest paths from member x to all other people in the social network [Sab66] and is calculated as follows:

$$CC(x) = \frac{m-1}{\sum_{i=1}^{m} d(x,y)}, \tag{7.12}$$

where $d(x, y)$ is the length of the shortest path from member x to y.

Betweenness centrality of member x pinpoints to what extent x is on the way between other members. It can be calculated only for undirected associations by dividing the number of the shortest geodesic distances (paths) from y to z by the number of the shortest geodesic distances from y to z that pass through member x. This calculation is repeated for all pairs of members y and z, excluding x. Betweenness centrality of member x is the sum of all the outcomes [Fre77, Fre80].

Another view of centrality, proposed by Borgatti, concentrates on the outcomes for nodes in the network [Bor05]. These outcomes can be, e.g. the speed and frequency of reception, which characterizes the traffic flow between nodes. Thus, the intensity of the traffic flow and the time that is required to pass information from one node to another should be investigated in order to evaluate the node centrality [Bor05]. The flow of information is also a basis to calculate the entropy that can be treated as the centrality measure. If the flow begins at node x and then an item is transferred from one node to another over the existing edges, then the probabilities that the flow stops at each node in the network are used to calculate the entropy [Tut07].

The eigenvector associated with the largest characteristic eigenvalue of the adjacency matrix has been proposed as a centrality measure by Bonacich [Bon72]. In this method, first the initial centrality of each member is established by utilization of one of the existing measures, e.g. degree, betweenness or closeness centrality. These necessary preliminary social positions are used as the input for the core part of the method and highly influence the outcome, i.e. different initial measures result in different final values, see Sec. 7.5.2. Having the initial values assigned, the eigenvector-like measure of centrality for the given member is calculated as the sum of the initial centrality values of all other members that are connected to this node. A shortcoming of this method is that members who are not chosen by others have centrality equal to zero. In consequence, these members contribute nothing to any member that is connected to them [Bon01]. As an extension of this method, called alpha-centrality, Bonacich and Lloyd propose an additional input status to ascribe each network member [Bon01]. This status is derived from member's general posi-

tion in the company or family rather than from their associations in the network. The numeric values of the status are added to the eigenvector-based centralities derived from member associations Note that unfortunately it is not always possible to establish such an additional status, especially for large social networks with thousands of members. Another modification of the eigenvector centrality takes into account not only the direct connections but also the indirect ones. Moreover, each indirect path is assigned with an appropriate weight [Bon87].

The second feature that characterizes a member in the social network is the member's prestige. Similarly to centrality, prestige can be calculated in various ways, e.g. proximity prestige, rank prestige, and degree prestige. The degree prestige is based on the indegree number so it takes into account the number of members that are adjacent to a particular member of the community [Was94]. In other words, more prominent people are those who received more nominations from members of the community [Ale63]. The degree prestige $DP(x)$ of member x can be described with the following formula:

$$DP(x) = \frac{i(x)}{m-1}, \tag{7.13}$$

where $i(x)$ is the number of members from the first level neighbourhood that are adjacent to x.

Proximity prestige $PP(x)$, in contrast to closeness centrality, reflects how close all members are within the social community to member x [Was94]. This measure depends on the geodesic distances of all members to x:

$$PP(x) = \frac{\frac{p(x)}{m-1}}{\frac{1}{p(x)}\sum_{i=1}^{p(x)} d(y,x)} = \frac{(p(x))^2}{(m-1)\cdot\sum_{i=1}^{p(x)} d(y,x)}, \tag{7.14}$$

where $p(x)$ - the number of members who can reach member x, i.e. there exists a path from these members to member x.

The rank prestige, which is also called status prestige [Was94], is measured based on the status of members in the network and depends not only on geodesic distance and the number of associations, but also on the prestige of members connected with the member [Kat53].

The measure of centrality and prestige can be utilized not only in the regular social networks but also in the web network, i.e. the network of web pages. Kumar *et al.* claim that the web can be seen as a social network [Kum02] and this enables similar node measures to be applied both to social networks and hypertext or web-based systems [Bot92].

Another concept, which is used to assess the status of the network member, is based on the formal theory of the opinion formation process. It consists of two

main components: the exogenous actors' initial opinions and the endogenous interpersonal influences [Fri98, Fri97]. The m actors' initial opinions denoted by the m-dimensional vector $y^{(0)}$ are described by the equation: $y^{(0)}=X\cdot b$, where: X is the $m\times k$ matrix of k exogenous variables; b is the $k\times 1$ vector of coefficients for the exogenous contributions. The transformation of these initial opinions is expressed in the following way [Fri98, Fri97]:

$$y^{(n+1)} = \alpha \cdot W \cdot y^{(n)} + (1 - \alpha) \cdot y^{(0)}, \tag{7.15}$$

where α – a scalar weight of the endogenous interpersonal opinions ($0\leq\alpha\leq 1$); W – an $m\times m$ matrix of endogenous interpersonal influences, n=1, 2, … .

Although Eq. (7.15) seems to be similar to the *SP* approach, compare Eq. (7.8), there are many crucial differences. First of all, the aim of the *SP* concept is to evaluate the static position of person within the social network and in consequence build the member ranking according to members' social positions. On the contrary, the model proposed by Friedkin analyzes how others influence individuals' opinions [Fri98]. The weights in the *SP* approach reflect how much of one's activity is passed to another user whereas Friedkin and Johnsen assigned weights based on the strength of the influence that one has on other people in the network [Fri97]. Contrary to *SP* (see the third condition for commitment function), in the social influence network theory the reflexive associations can have non-zero weights. Moreover, Friedkin and Johnsen require the weights of incoming relations to sum up to 1, including the reflexive ones. Contributions in *SP*, which fulfil a similar condition, Eq. (7.2) and (7.3), correspond to outgoing relations. Another significant difference between these two models is the role and assignment of initial values. In the social influence network theory the initial opinions depend on the user profile. In the *SP* approach, any non-negative values can be assigned as the initial social positions and *SP* remains convergent (Theorem 7.3, Sec. 7.3.2). These initial values $y^{(0)}$ are used for every iteration n+1, Eq. (7.15), whereas in *SP*, Eq. (7.8), only previous values are respected. Besides, contrary to *SP*, Eq. (7.8), coefficient α can have the separate value for each user [Fri97].

One of the most popular measures used for internet analysis is PageRank, which was introduced by Brin and Page to assess the value and importance of web pages [Ber05, Bri98, Bri06]. The PageRank value of a web page takes into consideration PageRanks of all other pages that link them to this particular one. Google uses this mechanism to rank the pages in their search engine. The main difference between PageRank and social position function is the existence and meaning of commitment function. In PageRank, all links have the same weight and importance whereas social position makes the quantitative distinction between the strengths of individual associations, see Sec. 7.3.1.

Xing and Ghorbani proposed a modified version of PageRank – Weighted PageRank (*WPR*) and carried out an investigation on only one query to the Google

search engine [Xin04]. They attached two additional weights to PageRank values, which together are in a sense, the equivalent of commitment function in Eq. (7.1). The main differences between these weights and the commitment function are their meaning, calculation and features. In *WPR* weights reflect additional, local structure properties round the node, in contradistinction to the strength of direct relationships from one node to another like in the case of commitment function. The *WPR* weights include quantities of incoming and outgoing links from all the nodes indirectly related to the one being considered. However, why do links outgoing from node y, e.g. to node u, and even links from node z to u, have direct influence on relationship from x to y while calculating $WPR(y)$, see Fig. 7.5. Similarly, to evaluate $WPR(x)$ using the link incoming from u, i.e. $WPR(u)$, we need to weaken the influence of $WPR(u)$ due to link from v to y, Fig 7.5. In other words, if u sends emails to x, then $WPR(x)$ is smaller when v sends emails to y. The interpretation of this facts is tricky in the case of social networks and human influences. Besides, *WPR* would not still respect any property of such communication, e.g. its intensity. This results from the fact that weights in *WPR* express the local structure property rather than weights assigned to links. Note that weights in social position reflect some external properties describing strength of relationships, e.g. communication intensity. In the case of *WPR* weights correspond to structure of the network. Hence, the *WPR* method operates on unweighted graph whereas social position is suitable for weighted ones. Besides, Xing and Ghorbani did not provide any requirements nor properties for their weighting functions. The convergence of the *WPR* iterative calculation has neither been proved.

Some authors use the term *social position* in the different context than in this section. Their social position is based on structural similarity of the first level associations, i.e. two members possess the same social position if they have the same neighbours [Fri98, Lor71, Was94].

7.5. Experiments

7.5.1. Ranking Creation and Comparison

The values of the social position function, Eq. (7.1), (7.5), and (7.8), can be utilized to create a ranking list of network members. First, the iterative way of calculation is used (see Sec. 7.2.4) to obtain values of social positions with the given precision. Next, the obtained values are utilized to order network members, i.e. individuals with the highest social position are placed at the top whereas members with the lowest social position occupy the last position in the ranking. Two members with the same social position are assigned the same, higher position in the ranking and the consecutive ranking position remains empty.

For example, for the set of social positions $\{SP(x_1)=0.9, SP(x_2)=1.3, SP(x_3)=2.1, SP(x_4)=0.9, SP(x_5)=0.4, SP(x_6)=0.9, SP(x_7)=0.5\}$, we have the following ordered list of individuals, each with the corresponding position in the ranking: $((x_3,1); (x_2,2); \{(x_1,3); (x_4,3); (x_6,3)\}; (x_7,6); (x_5,7))$. Note that three members x_1, x_4, x_6 occupy the same third position whereas the next member x_7 is in the 6th place.

The same procedure of ranking creation has been applied to rankings based on degree centrality *DC*, Eq. (7.11), closeness centrality *CC*, Eq. (7.12), degree prestige *DP*, Eq. (7.13), and proximity prestige *PP*, Eq. (7.14).

Once ranking lists have been created, a method to compare them is needed. For this purpose Kendall's coefficient of concordance was used to determine the similarity between two ranking lists, see Sec. 3.9.3, Eq. (3.8). When two rankings *A* and *B* have the same items in every position, Kendall's coefficient $\tau(A,B) = +1$, but when these rankings have all the same items in reverse order, $\tau(A,B) = -1$.

7.5.2. Classic Thurman Network

The Thurman office social network is a non-symmetrical network of 15 people who worked in one company (Fig. 7.15). Thurman spent 16 months observing the interactions among employees in the office of a large corporation [Thu79]. The adjacency matrix for the Thurman network is presented in Table 7.3, where non-zero values represent the existence of the connection between two users and their values correspond to the commitment in activity, i.e. strength of associations. For example, *President* (10) contacts *Amy* (4) whereas *Amy* does not communicate to *President*. These values have equalled one in the original adjacency matrix.

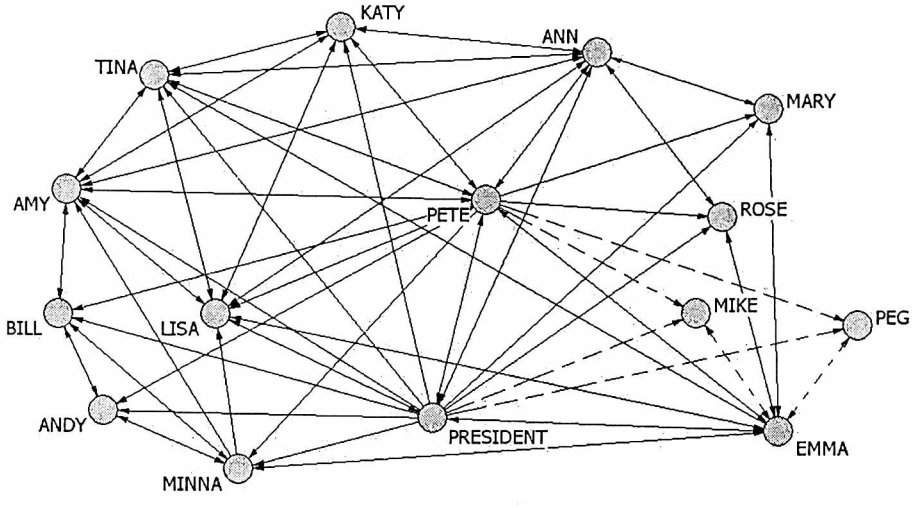

Fig. 7.15. Graph representation of the classic Thurman office social network

Table 7.3. Commitments in activity within the classic Thurman office social network

Member	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1. Emma			1/9		1/9	1/9		1/9		1/9		1/9	1/9	1/9	1/9
2. Ann			1/8	1/8	1/8	1/8	1/8			1/8		1/8	1/8		
3. Pete	1/14	1/14		1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14
4. Amy		1/6	1/6		1/6	1/6	1/6		1/6						
5. Lisa	1/7	1/7	1/7	1/7		1/7	1/7			1/7					
6. Tina		1/5	1/5	1/5	1/5		1/5								
7. Katy		1/5	1/5	1/5	1/5	1/5									
8. Minna	1/5			1/5	1/5				1/5		1/5				
9. Bill				1/3				1/3			1/3				
10. President	1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14	1/14		1/14	1/14	1/14	1/14	1/14
11. Andy			1/3					1/3	1/3						
12. Mary	1/2	1/2													
13. Rose	1/2	1/2													
14. Mike	1														
15. Peg	1														

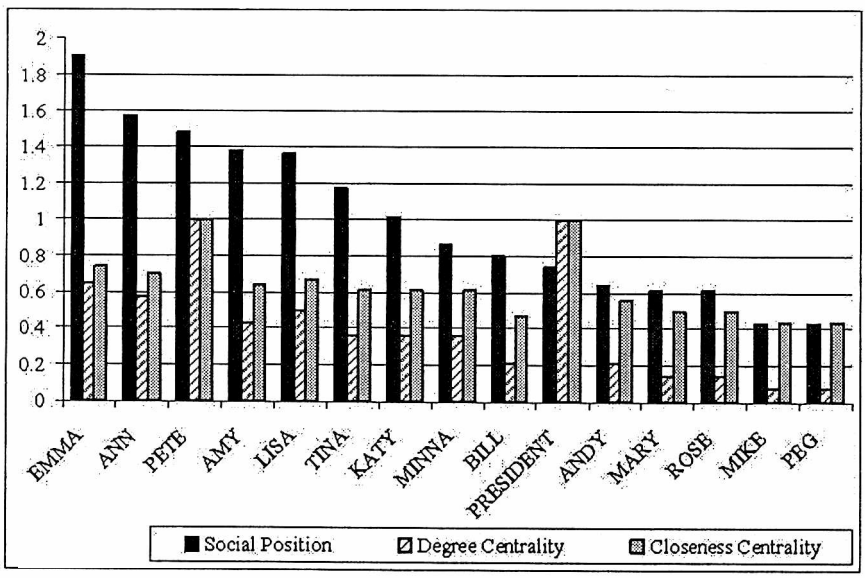

Fig. 7.16. The comparison of centrality measures (degree centrality *DC* and closeness centrality *CC*) to social position *SP* in the classic Thurman network

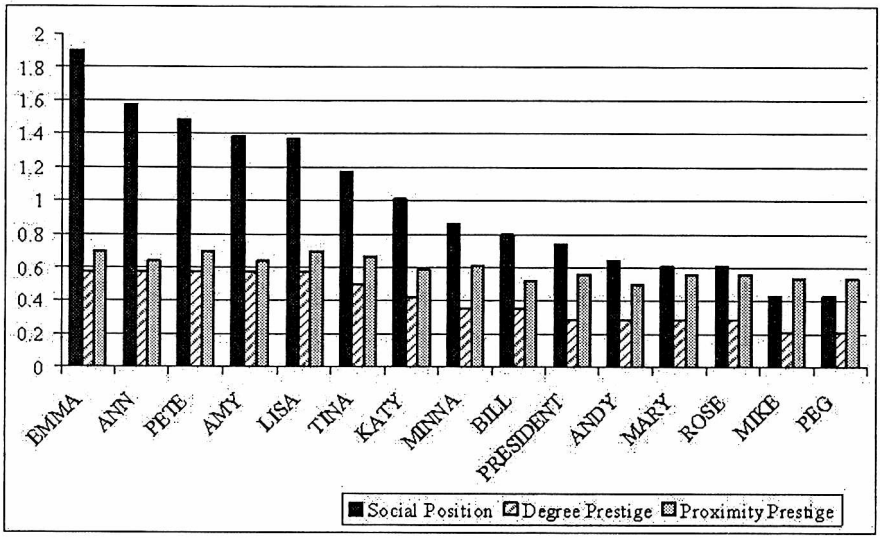

Fig. 7.17. The comparison of prestige measures (degree prestige *DP* and proximity prestige *PP*) to social position *SP* in the classic Thurman office network

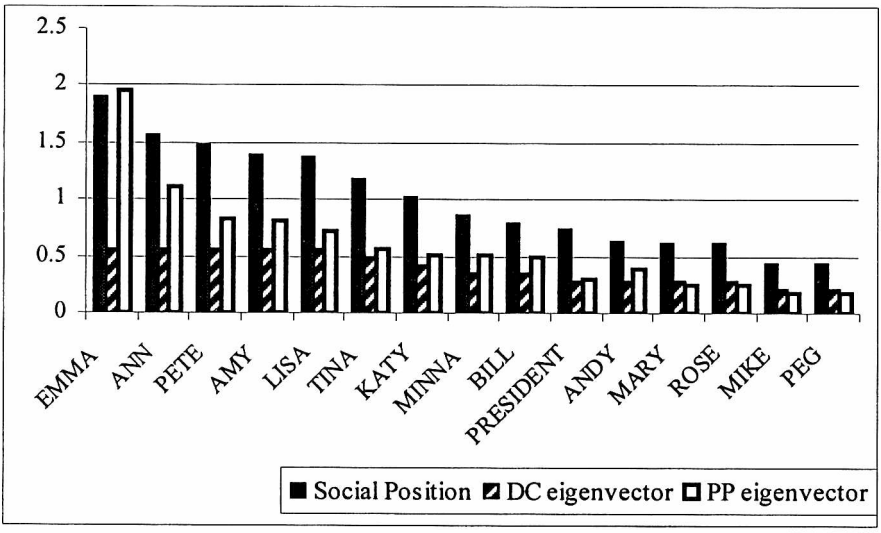

Fig. 7.18. The comparison of eigenvector measures (based on degree centrality *DC* and proximity prestige *PP*) with social position *SP* in the classic Thurman office network

In order to obtain the commitments in activity for each individual, value one - from the original matrix - is divided by the number of member's associations, e.g. *Emma* communicates with nine members, so her contribution of activity to each of her acquaintances equals $^1/_9$. The outcomes of these calculations are presented in Table 7.3.

The other measures have been calculated according to the appropriate formulas from Sec. 7.4: degree centrality (*DC*) - Eq. (7.11), closeness centrality (*CC*) - Eq. (7.12), degree prestige (*DP*) - Eq. (7.13), and proximity prestige (*PP*) - Eq. (7.14). They are compared with *SP* in Fig. 7.16 and 7.17. Centrality based on eigenvectors is calculated using two different initial, input centralities: degree centrality (*DC Eigenvector*) and proximity prestige (*PP Eigenvector*). After that the eigenvector measure is contrasted with *SP* (Fig. 7.18).

To prepare experiments some primary assumptions have to be made. The initial social positions $SP_0(x)=1$ are established for every member x in the network. The value of ε is 0.9 and the stop condition is: no difference in social position values to the precision to the fifth decimal place for all the members in two consecutive iterations, i.e. $\tau=0.00001$, Eq. (7.10).

Table 7.4. The positions in rankings for the analyzed measures in the Thurman office social network, for $\varepsilon=0.9$

Member	*SP*	*DC*	*CC*	*DP*	*PP*	*DC Eigenvector*	*PP Eigenvector*
Emma	1	3	3	1	1	1	1
Ann	2	4	4	1	5	1	2
Pete	3	1	1	1	1	1	3
Amy	4	6	6	1	5	1	4
Lisa	5	5	5	1	1	1	5
Tina	6	7	7	6	4	6	6
Katy	7	7	7	7	8	7	7
Minna	8	7	7	8	7	8	8
Bill	9	10	13	8	15	8	9
President	10	1	1	10	9	10	11
Andy	11	10	10	10	12	10	10
Mary	12	12	11	10	9	10	12
Rose	12	12	11	10	9	10	12
Mike	14	14	14	14	13	14	14
Peg	14	14	14	14	13	14	14

Based on the values obtained, seven separate rankings have been created. The positions of each member in every ranking are presented in Table 7.4. Note that

the order of people based on their social position, degree centrality, and closeness centrality varies a lot. On the other hand, the rankings of members based on their social position and degree prestige are quite similar, even though the distribution of social position is greater. However, social position provides a better opportunity to distinguish individuals within the network as opposed to both prestige measures. The information that *Emma, Ann, Pete, Amy,* and *Lisa* have the same, greatest degree prestige is insignificant since it results from the number of other members who are adjacent to these people. The social position measure $SP(x)$ takes into consideration not only the number of members who communicate to the evaluated person x but also their social positions and contribution in their activity directed to x. Based on these properties we can observe that *Emma* is the person with the highest social position in the network because *Mike* and *Peg* communicate only with *Emma* so they transfer their entire social positions to her. The prestige measures do not respect these features and this appears to be critical in assessing the importance of an individual in the social network.

Based on the comparison of the social position with both eigenvector measures we can observe that the rankings are very similar. However, in the case of *DC Eigenvector*, five people occupy the first place. Note that, in the eigenvector-like measures, only members that directly communicate to the given person influence the user position. On the contrary, the *SP* value of an individual depends on *SP*s of all members in the network due to recursive character of the social position measure. Moreover, the final eigenvector centrality varies depending on the method that was used to evaluate the initial centralities (see Fig. 7.18), whereas the initial values of *SP* – according to the experiments carried out – do not influence their final, limit values. Furthermore, the social position measure also respects the strength of each association in the form of commitment in activity, see Eq. (7.1). The commitment function is individualized for each pair of associations and reflects the real contribution in activity directed from one person to another. In the eigenvector approach, we have only row normalization of the adjacency matrix and the individualized values of personal initial centralities.

Table 7.5. Kendall's coefficient for each pair of rankings from Table 7.4

	SP	*DC*	*CC*	*DP*	*PP*	*DC Eigenvector*
DC	0.724	–	–	–	–	–
CC	0.676	0.895	–	–	–	–
DP	0.829	0.629	0.581	–	–	–
PP	0.657	0.571	0.619	0.657	–	–
DC Eigenvector	0.828	0.628	0.581	1	0.657	–
PP Eigenvector	0.962	0.709	0.652	0.828	0.681	0.828

For each pair of the rankings from Table 7.4, Kendall's coefficient, Eq. (7.16) was calculated and the results are presented in Table 7.5 and Fig. 7.19. Note that all rankings based on the neighbours directly connected to the given one (incoming associations), i.e. *DP*, *DC Eigenvector*, *PP Eigenvector*, and *SP* are in pairs very close; the range of Kendall's coefficient is from 0.828 up to 1 between *DP* and *DC Eigenvector*. Similarly, the rankings based on measures that take into account outgoing associations *DC* and *CC* are alike - Kendall's coefficient at the relatively high level of 0.895.

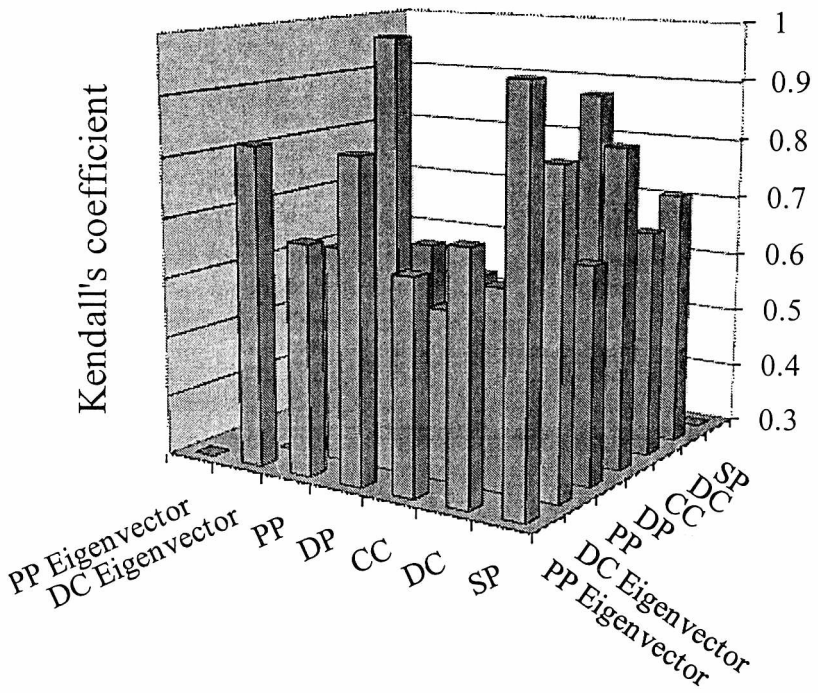

Fig. 7.19. The values of Kendall's coefficient for the pairs of rankings from Table 7.4

For that reason, the ranking based on social position (*SP*) is most similar to the rankings based on *PP Eigenvector*, *DC Eigenvector* and degree prestige (*DP*). On the contrary, the social position ranking is least similar to the ranking of the proximity prestige (*PP*) and closeness centrality (*CC*) measures (Table 7.4, Fig. 7.19).

The rankings based on *PP Eigenvector* and *DC Eigenvector* are more similar to other rankings rather than to each other. Kendall's coefficient for *PP Eigenvector* and *DC Eigenvector* equals 0.828 whereas for *DC Eigenvector* and *DP* is greater - equals 1 and for *PP Eigenvector* and *SP* is 0.962. This leads to the conclusion that

the initial centralities (*DC* and *PP*) highly influence the final values based on eigenvectors.

7.5.3. Flawed Thurman Network

The second experiment was conducted on a modified version of the Thurman network. Two members, *Mike* and *Peg*, and their associations were excluded from the original network (Fig. 7.15) to study how the previously calculated values of social position, centrality, and prestige would change if the least active people were passed over. The fact that *Mike* and *Peg* communicate only with *Emma* should be emphasized here (Fig. 7.15).

The input data for the calculation remains the same as for the original Thurman network. It means that the initial social positions equal 1 for every member, ε=0.9 and the stop condition isτ=0.00001, Eq. (7.10).

The commitments in activity for each member are also established in a similar way as in Sec. 7.5.2. Note that after exclusion of *Mike* and *Peg* some commitments have changed, e.g. *Emma* now communicates with seven members instead of nine so her contribution of activity to each of her acquaintances equals $^1/_7$.

Social position values have been compared to other measures: degree centrality (*DC*) - Eq. (7.11), closeness centrality (*CC*) - Eq. (7.12), degree prestige (*DP*) - Eq. (7.13), and proximity prestige (*PP*) - Eq. (7.14), see Fig. 7.20 and 7.21.

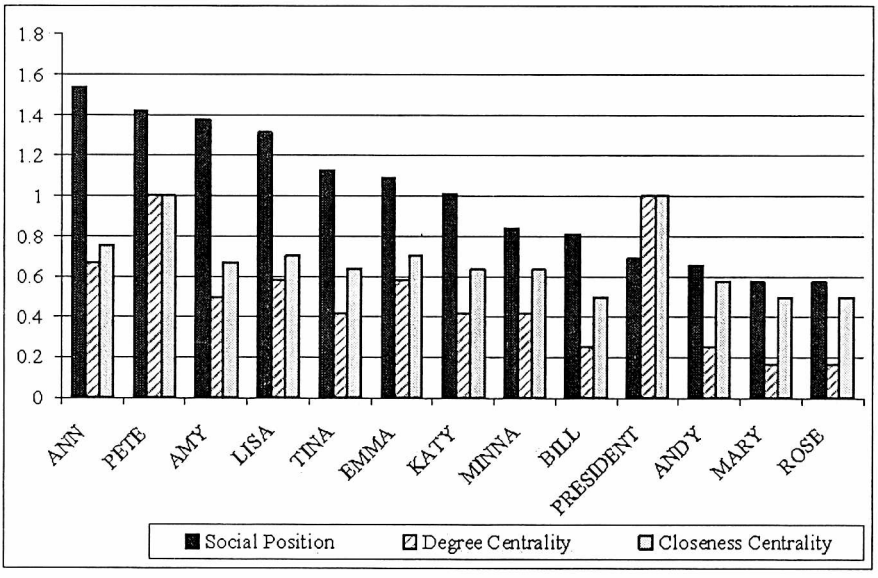

Fig. 7.20. The comparison of centrality measures (degree centrality *DC* and closeness centrality *CC*) to social position *SP* in the modified Thurman network

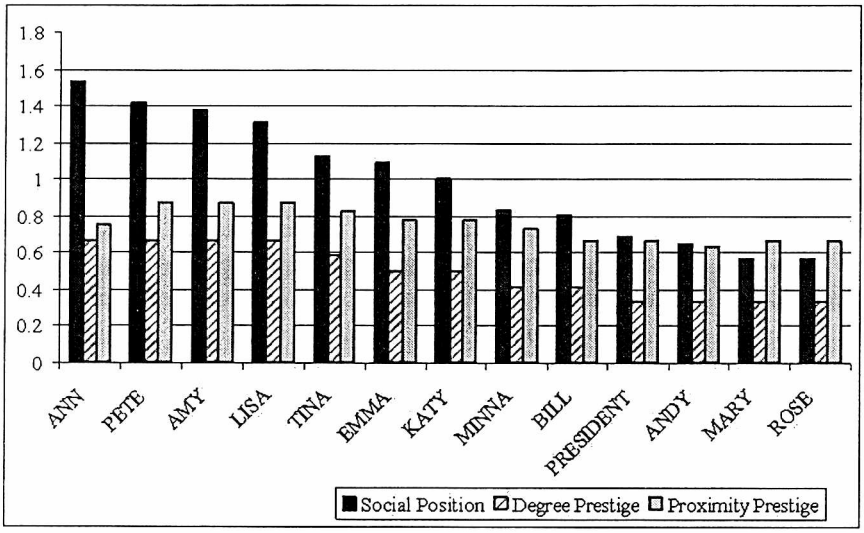

Fig. 7.21. The comparison of prestige measures (degree prestige *DP* and proximity prestige *PP*) to social position *SP* in the modified Thurman network

Table 7.6. The positions in rankings for the analyzed measures in the Thurman office social network, for ε=0.9. *SP* change column denotes the difference in social position between the modified and initial network

Member	*SP* rank	*SP* change	Gain in *SP*	*DC*	*CC*	*DP*	*PP*
Ann	1	-0.036 (-2.3%)	-	3	3	1	7
Pete	2	-0.061 (-4.2%)	-	1	1	1	1
Amy	3	-0.005 (-0.3%)	~0	6	6	1	1
Lisa	4	-0.053 (-3.9%)	-	4	4	1	1
Tina	5	-0.046 (-4%)	-	7	7	5	4
Emma	6	-0.809 (-42.6%)	- - -	4	4	6	5
Katy	7	-0.004 (-0.4%)	~0	7	7	6	5
Minna	8	-0.025 (-2.9%)	-	7	7	8	8
Bill	9	+0.015 (+1.9%)	+	10	11	8	9
President	10	-0.049 (-6.7%)	-	1	1	10	9
Andy	11	+0.015 (+2.4%)	+	10	10	10	13
Mary	12	-0.038 (-6.3%)	-	12	11	10	9
Rose	12	-0.038 (-6.3%)	-	12	11	10	9

In the next step, based on these values, five separate rankings have been created (Table 7.6).

As in the initial version of the Thurman office social network, the rankings of individuals based on their social position and degree prestige are quite similar,

although the distributions of values vary a lot. The rankings of degree centrality and closeness centrality are almost the same, but, especially in the first six positions, they are quite different than the social position ranking.

In general, the conclusions that can be drawn from the comparison between rankings based on different measures for the modified network are coherent with those for the original network, see Sec. 7.5.2.

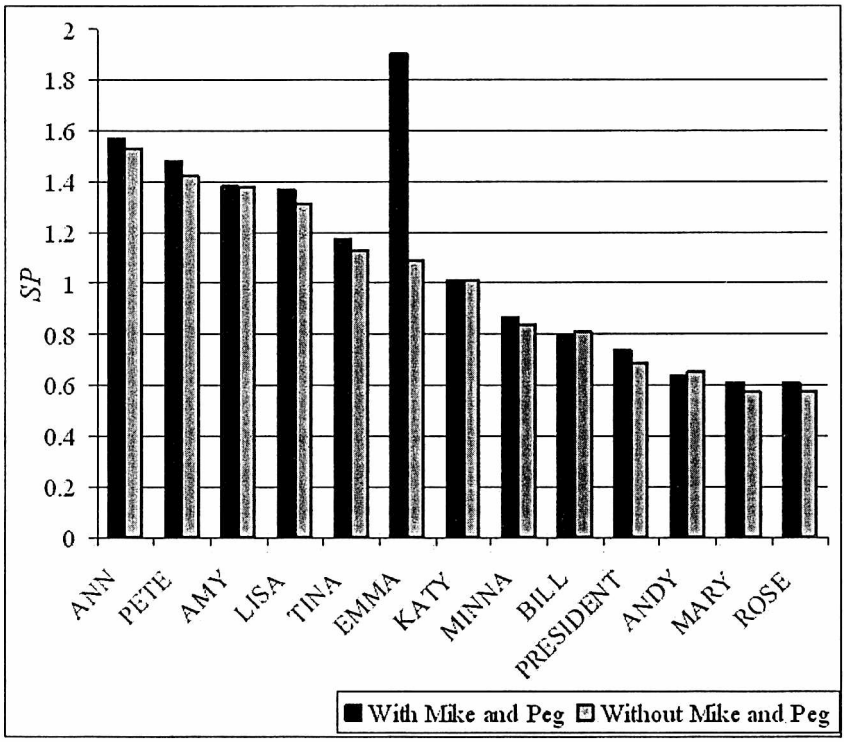

Fig. 7.22. The comparison of the classic and the modified Thurman network

The comparison of the social positions values for the initial and flawed version of the Thurman office network has revealed some interesting issues related to *Emma*'s social position (Fig. 7.22, Table 7.6). As we can observe, *Emma* who obtained the highest social position in the original network, here is in the sixth position and her social position before was twice as much as in the modified network (drop by 43%). This is caused by the removal of *Mike* and *Peg* from the network, who communicated only with *Emma* and transferred their entire social positions to her. The relative position of all other members of the community in the ranking does not change so much (<6.5%). However, all persons directly linked from *Emma* lost in their social position from 2.9% to 6.7%, whereas the others, not linked (*Amy, Katy*)

kept or even gained in their social position (*Bill, Andy*). This was because *Emma*'s social position that she passed to her acquaintances was much lower. The only exception to this rule was *Ann*, who lost 2.3%. This was caused by many other members that were incidental to her and who also lost much (Table 7.6).

Concluding the research on the modified Thurman office network, we can find that the removal of one individual x from the network results in the discernible decline in the social position of x's acquaintances to whom x passes his position. Moreover, this affects most of the acquaintances of x's acquaintances as well. The longer the paths from x to others, the weaker the impact of x's disappearance on their social position is. Although the social position of each individual depends directly only on their nearest neighbours, Eq. (7.1), any change in one part of the network influences the entire community.

7.5.4. Test Environment for Email-based Experiments

The large-scale experiments that illustrate the idea of social position assessment were carried out separately on two datasets: Enron employees' mailboxes and the Wrocław University of Technology (WUT) mail server logs. Enron Corporation was the biggest energy company in the USA. It employed around 21,000 people before its bankruptcy at the end of 2001. A number of other researches have been conducted on the Enron email dataset [Pri05, She05]. Some preliminary analyses on Enron dataset have been presented in [Kaz07g, Kaz08c]. The second dataset contains logs collected by the mail server of WUT and refers only to the emails incoming to the staff members as well as entire organizational units registered at the university.

First, the data has to be cleansed by removal of bad and unification of duplicated email addresses. Additionally, only emails from and to the Enron or WUT domain were preserved. Every email with more than one recipient was treated as fraction of the regular email. The strength of email communication $S(x \rightarrow y)$ from x to y was used for this purpose:

$$S(x \rightarrow y) = \sum_{i=0}^{EM(x \rightarrow y)} \frac{1}{n_j(x \rightarrow y)}, \tag{7.16}$$

where $EM(x \rightarrow y)$ – the set of all email messages sent by x to y; $n_j(x \rightarrow y)$ – the number of all recipients of the ith email sent from x to y.

Based on the strength of the email communication from one user to another the commitment $C(x \rightarrow y)$ from Eq. (7.6) can be redefined as follows:

$$C(x \rightarrow y) = \frac{S(x \rightarrow y)}{n(x)}, \tag{7.17}$$

where $n(x)$ – the total number of emails sent by user x.

The general statistics related to the processed datasets are presented in Table 7.7.

Table 7.7. The statistical information for the Enron and WUT datasets

Statistical data	Enron	WUT
No. of emails before cleansing	517,431	8,052,227
Period (after cleansing)	01.1999-07.2002	02.2006-09.2007
No. of emails after cleansing	411,869	8,052,227
No. of external emails (sender or recipient outside the Enron/WUT domain)	120,180	5,252,279
No. of internal emails (sender and recipient from the Enron/WUT domain)	311,438	2,799,948
No. of distinct, cleansed email addresses	74,878	165,634
No. of isolated users	9,390	0
No. of distinct, cleansed email addresses from the Enron/WUT domain (social network users) without isolated members, the set M in $VSN=(M, R)$, Definition 7.1	20,750	5,845
Emails per user	15	479
No. of network users within VSN with no activity	15,690 (76%)	26 (0.45%)
Commitments extracted from emails	201,580	149,344
No. of non-zero relationships, $C(a \rightarrow b)>0$	250,003	176,504
Associations per user	12.0	30.2
Percentage of all possible associations	0.0583%	0.517%

After data preparation the commitment function is evaluated for each pair of members. To evaluate association commitment function $C(x \rightarrow y)$ Eq. (7.17) was used. The initial social positions for all members were established to 1 and the stop condition $\tau=0.00001$ was applied separately for each user. The social positions without and with time coefficient were calculated for six different values of coefficient ε, i.e. $\varepsilon=0.01$, $\varepsilon=0.1$, $\varepsilon=0.3$, $\varepsilon=0.5$, $\varepsilon=0.7$, and $\varepsilon=0.9$.

7.5.5. Iterative Data Processing

The experiments conducted revealed that the number of iterations necessary to calculate the social positions for all users tightly depends on the value of the parameter ε, see Eq. (7.1): the greater the value of ε, the greater the number of iterations, Fig. 7.23. The number of iterations directly influences the processing time. Thus, much more time is required to fulfil the same stop condition $\tau=0.00001$ for greater values of coefficient ε, Fig. 7.23.

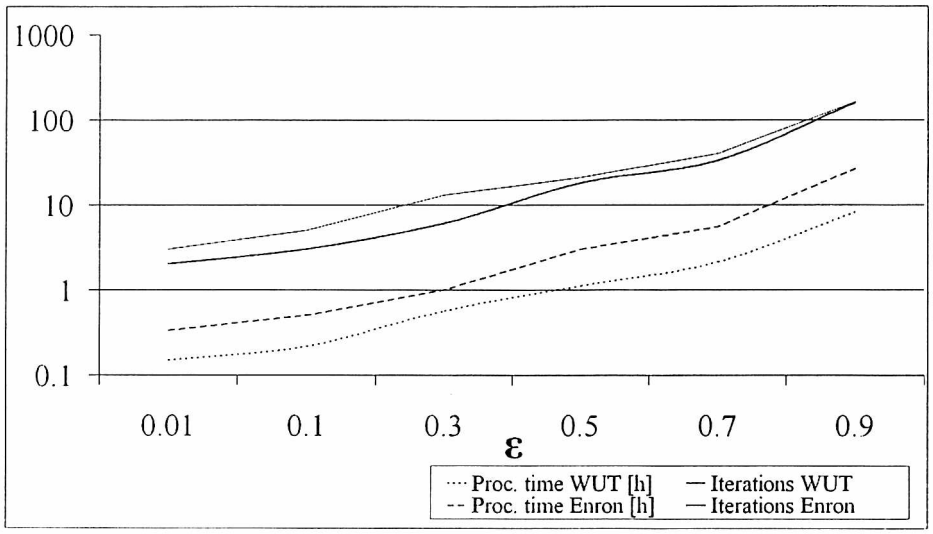

Fig. 7.23. The number of necessary iterations and processing time in relation to ε

Obviously, both the processing time and the number of iterations also depend on precision level τ. The smaller the value of τ, the more calculations are necessary.

Efficiency of calculations can be essential in the case of large social networks that contain many millions of nodes. The quantity of calculations can be reduced by applying either a greater τ or a smaller ε. However, in both cases, it would happen at the expense of precision, see Sec. 7.2.4.

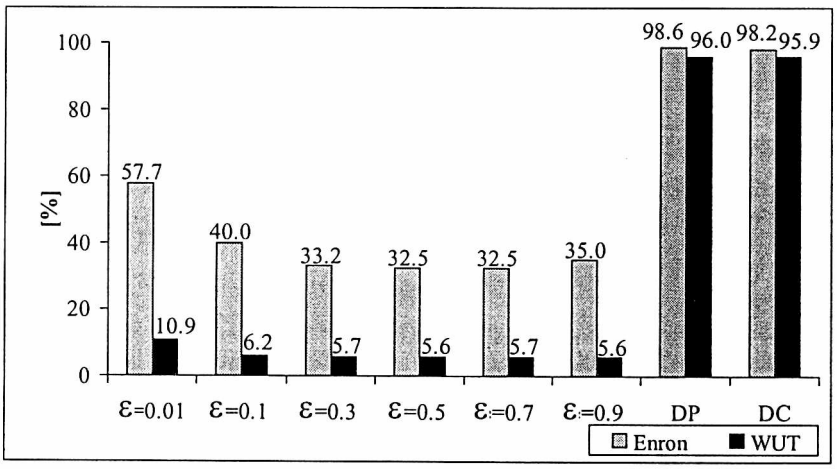

Fig. 7.24. Percentage of duplicates within the set of node measures, separately for social position *SP* with different values of ε, degree prestige (*DP*), and degree centrality (*DC*)

7.5.6. Diversity of Social Position Compared to Other Measures in Email-based Social Networks

Social position measure appears to be more diverse than the other measures. This can be visible especially while analyzing the number of nodes that possess the same centrality value, Fig. 7.24. Social positions are better for every value of ε, compared to the degree prestige (*DP*) and degree centrality (*DC*). Note that degree prestige function provides only 286 distinct values for Enron and 208 for WUT. In the case of the degree centrality, there are only 383 distinct values for Enron and 242 for WUT. For that reason, the percentage of duplicates exceeds 95% for degree measures whereas it is below 60% for social positions in Enron and below 11% for WUT, Fig. 7.24.

7.5.7. Ranking Comparison in Email-based Social Networks

To compare rankings created upon different measures in the email-based social networks, Kendall's coefficient of concordance was used, Eq. (3.8). It was calculated separately for each pair of user rankings based on the values of degree centrality (*DC*), degree prestige (*DP*), and social position for different ε, Table 7.8 and 7.9.

Table 7.8. Kendall's coefficients for the Enron email dataset

	ε=0.01	ε=0.1	ε=0.3	ε=0.5	ε=0.5	ε=0.9	*DC*
ε=0.1	0.9988						
ε=0.3	0.8727	0.8732					
ε=0.5	0.8623	0.8627	0.9850				
ε=0.7	0.8474	0.8478	0.9681	0.9822			
ε=0.9	0.8308	0.8311	0.9484	0.9620	0.9796		
DC	0.0041	0.0041	0.0084	0.0081	0.0077	0.0074	
DP	0.0052	0.0052	0.0081	0.0079	0.0077	0.0746	0.3517

Table 7.9. Kendall's coefficients for the WUT email dataset

	ε=0.01	ε=0.1	ε=0.3	ε=0.5	ε=0.5	ε=0.9	*DC*
ε=0.1	0.9782						
ε=0.3	0.9399	0.9612					
ε=0.5	0.9054	0.9262	0.9638				
ε=0.7	0.8691	0.8886	0.9237	0.9582			
ε=0.9	0.8197	0.8355	0.8652	0.8967	0.9366		
DC	-0.6874	-0.6946	-0.7083	-0.6931	-0.7353	-0.7497	
DP	-0.6655	-0.6716	-0.6829	-0.7215	-0.7027	-0.7099	0.7919

The similarity of rankings based on social position calculated for different ε provided an obvious correlation: the greater the difference in ε, the less similar the rankings. However, for any two values of ε, Kendall's coefficient was relatively high and always greater than 0.82. Hence, social position is a stable measure that depends on ε to a limited extent.

Simultaneously, *SP*-based rankings were different than both *DC*- and *DP*-based ones: Kendall's coefficient was from -0.75 (WUT) to only 0.07 (Enron). The closeness between *DC*- and *DP*-based rankings was rather high: Kendall's coefficient equalled 0.35 for Enron and as much as 0.79 for WUT. *DC*- and *DP*-based rankings are close to each other and differ from *SP*-based because both *DC* and *DP* provide a big number of duplicates and do not distinguish between users, see Sec. 7.5.6. This shows that *DC* and *DP* deliver similar, limited knowledge about users in the network whereas social position function is a diverse, meaningful measure.

7.5.8. Top Network Members in Email Communication

After analyzing the individual ranking position based on the social position measure, it appears that one of the highest positions in the Enron social network is occupied by Kenneth Lay: the 5th place for ε=0.01, the 2nd for ε=0.1 and ε=0.3, the 1st for ε=0.5 and ε=0.7, and finally the 4th place for ε=0.9. Kenneth Lay was the former Chairman of the Board and Chief Executive Officer, who was accused and sentenced for a broad range of financial crimes. Another Enron employee Vince Kaminski, who was risk-manager and as one of the first uncovered the frauds in Enron, takes the 9th place for ε=0.01, the 5th for ε=0.1, ε=0.3, and ε=0.5, the 3rd for ε=0.7, and finally the 1st place for ε=0.9.

Top email users in WUT are: 1 faculty library, Science Information Centre, 4 individuals - network administrators, 1 trade union, Promotion Department, 1 dean, and Ph.D. Office.

The lists of top 10 email users in both organizations are rather stable regardless of the ranking function. This means that top users can change their rank positions in relation to ε. However, the changes are rather insignificant and these users still remain on the top of the ranking lists.

7.6. Conclusions

Social position is a measure of the importance of a member in the social network that reflects the characteristic of the member's neighbourhood. Its value for a given individual respects both social positions of the nearest acquaintances as

well as their attention directed to the person considered. Thus, the social position measure gives opportunity to analyze virtual social networks with respect to the social behaviour of individuals, especially their mutual communication or common activities within internet services. In other words, social position makes use of associations extracted from communication data.

Although the social position assessment is easier to perform in virtual social networks, it can also be applied to evaluate the social status of individuals within the regular weighted social networks in which the weights reflect the strength of the relations.

Social position can be effectively estimated using the iterative way of calculation, i.e. SPIN algorithm, see Sec. 7.2.4. Its convergence as well as other features like the fixed total and average value, and the limit superior have been proved in a formal way, see Sec. 7.3.

The experiments have shown that the social position appears to be more meaningful than some other measures like centrality or prestige, which are commonly utilized to assess the position of individuals in the social network, see Sec. 7.5. Contrary to other measures, the social position of each individual indirectly takes into account the strength of all associations existing within the network, i.e. the structure of the entire network.

The social position value of the member depends on positions of all members in the network, whereas other measures either do not consider positions of others like prestige and centrality measures (see Sec. 7.4) or they respect only the first-level neighbours, e.g. eigenvector-like measures. Additionally, the final values obtained by utilization of the eigenvector measures tightly depend on the initial centralities, while the initial values of SP do not influence their final, limit values.

The social position function appears to be a powerful, stable and diverse measure, which can be used to select key users for project teams, find new potential employees, search for the best potential consumers for advertising campaigns or recommender systems [Kaz06b], and finally for use in target marketing [Yan06]. It is applicable especially to large datasets like email communication [Kaz08c, Kaz *b*].

8 Summary

Associations occur wherever two or more objects get into relationships. They can link sets, see Sec. 3, 4, 6, lists, see Sec. 5, or single pairs, see Sec. 7. Associations play an important role in the modelling of complex structures and many application areas like recommender systems, social networks or web content management systems.

There are many different types of associations but in this monograph only some of them were analyzed: indirect association rules, negative association rules, sequential patterns with the negative conclusions, dynamic user-to-cluster associations and associations between social network members.

Overall, associations can be either discovered or simply calculated from the source data. In the former case, associations are unknown in advance and need to be extracted from larger sets or source sequences. This especially refers to association rules and sequential patterns. Since they typically are discovered from larger datasets and operate on big numbers of objects, they usually require specialized, efficient algorithms. Nevertheless, there also exist obvious associations that can be simply calculated like relationships between humans extracted from the data about their mutual activities or communication, see Sec. 7.

All types of associations may entail further processing and analysis. In most applications, we have a vast amount of associations available. For that reason, their interpretation can be complicated and often requires filtering or merging mechanisms, see Sec. 4.4.2 and 5.5.2. Moreover, this filtering can be an integral component of the association extraction process, like for example considering only rules that match existing hyperlinks in the SPAWN algorithm, see Sec. 5.3.2.

The associations obtained can be utilized in a variety of ways. First of all, they can be the input for other more condensed measures or indicators. Examples of this can be complex association rules, which usually extend recommendation lists for individual web pages, see Sec. 3, Positive and Negative Recommendation functions, see Sec. 4 and 5, recommendation lists created upon user-to-cluster associations, see Sec. 6, social position centrality measure in social networks, see Sec. 7.

Only some selected application areas of different association types have been considered in this monograph; including recommender systems, content management systems (CMS) and social networks. In particular, in the range of recommender systems, recommendations of pages - next steps in the web sites, see Sec. 3 and 5.6, or assignment of web advertisements in the personalized way have been studied, see Sec. 6. Within the web content management systems, the application of positive and negative associations to hyperlink usability assessment was analyzed, see Sec. 4 and 5.5. Besides, associations between humans in the social network were applied to compute the importance of network nodes - social position function, see Sec. 7.

8.1. Main Contribution

Every single section of this monograph makes its own unique contribution to science. This in particular includes: new patterns in data mining, new algorithms for pattern extraction, new measures, analysis of their profile and application areas both for new and well-known patterns.

Novel patterns that differ from the others known in data mining, namely, indirect association rules and sequential patterns with negative conclusions were introduced in Sec. 3 and 5, respectively. The former are derived from previously mined regular association rules and reflect transitive relationships existing between objects, i.e. if there are two associations $X \rightarrow Y$ and $Y \rightarrow Z$ then there also exists a partial indirect association $X \rightarrow Z$ through Y, see Sec. 3.4.1. All partial rules from X to Z are aggregated into one complete indirect rule, see Sec. 3.4.2. Patterns in which Y is a set of items were considered in Sec. 3.4.3. Besides, combinations of indirect and direct association rules in the form of complex association rules were proposed in Sec. 3.4.5. They, in turn, were utilized in web-based recommender systems. Experiments proved that indirect association rules can effectively extend recommendation lists compared to similar lists based on only direct rules, see Sec. 3.9. To discover both indirect and complex association rules the IDARM* algorithm was proposed in Sec. 3.6.

Section 4 refers to application of already known patterns - positive association rules and first of all negative association rules to verify existing links between objects. It has been performed in the web environment to assess usability of hyperlinks between web pages. To achieve this both positive and negative association rules need to be merged and matched with hyperlinks. The matching is also carried out during the extraction process, i.e. inside the proposed PANAMA algorithm for mining both positive and negative association rules. The usability of both types of association rules in hyperlink assessment was confirmed by experiments with the participation of web content managers from Poland and the Netherlands.

Another type of patterns - sequential patterns with negative conclusions is proposed in Sec. 5. They are in a sense a fusion of regular sequential patterns and negative association rules. The left-hand side of a sequential pattern with the negative conclusion is a sequence which is negatively related with a set on the right-hand side. Hence, a pattern $q \rightarrow \sim X$ means that elements of set X hardly occur after the frequent sequence q. The SPAWN algorithm for mining sequential patterns with the negative conclusions based on maximum complements, which constitute the projection database, is proposed in Sec. 5.3.2. Two implementation areas that integrate sequential patterns with the negative conclusions together with positive and negative association rules were analyzed in Sec. 5.5 and 5.6.

Section 6 describes the unique concept for personalized, adaptive web advertising that integrates content and usage knowledge by means of dynamic associations, which link current user behaviour with the beforehand computed content and usage clusters. The associations can change over time due to evolving user interests. The connections between the clusters and the advertisers' web sites enable suggestion of advertisements that are relevant to recent user needs and simultaneously are most likely to be clicked by the user.

Section 7 provides a new centrality measure for the social network - social position that makes use of associations between network members - humans. The requirement set for the association weights - commitment function has been postulated in Sec. 7.2.2 and 7.2.3. The iterative SPIN algorithm for the evaluation of social positions is presented in Sec. 7.2.4. A significant contribution is formal analyses on social position profile, see Sec. 7.3. The features of social position function were studied in the experimental way using large email datasets, see Sec. 7.5. As a result, much greater diversity of social position values, compared to the other measures, was observed.

Concluding, the essential contribution of the monograph includes:

1. New methods:

a) for extraction of indirect knowledge in the form of indirect and complex association rules, see Sec. 3;

b) for extraction of negative knowledge, i.e. negative association rules, see Sec. 4, and sequential patterns with negative conclusions, see Sec. 5;

c) for extension of recommendation lists using indirect association rules, see Sec. 3;

d) for verification of the previously known associations, in particular, negative recommendations and verifications:

- hyperlink usability assessment based on positive and negative association rules, see Sec. 4, as well as positive and negative sequential patterns, see Sec. 5.5;
- verification of content-based recommendation lists based on positive and negative usage patterns, see Sec. 5.6;

e) for personalized and adaptive web advertising, see Sec. 6;

f) for evaluation of human importance in the social network, see Sec. 7.

2. New patterns in data mining: indirect association rules and complex association rules, see Sec. 3.4, sequential patterns with negative conclusions, see Sec. 5.3.

3. Four new algorithms: IDARM*, see Sec. 3.6, PANAMA, see Sec. 4.3.3, SPAWN, see Sec. 5.3.2, SPIN, see Sec. 7.2.4.

4. A number of new measures and functions that facilitate application of different associations in various domains:

a) measures for new patterns: partial indirect confidence, see Sec. 3.4.1, complete indirect confidence, see Sec. 3.4.2, complex confidence, see Sec. 3.4.5, support and confidence for sequential patterns with negative conclusions, see Sec. 5.3.1;

b) Positive and Negative Recommendation functions for the assessment of hyperlinks, see Sec. 4.4.2, 5.5.2 and their combinations - positive and negative verification functions, see Sec. 5.5.3;

c) ranking functions for recommender systems, see Sec. 5.6.2;

d) functions that reflect user behaviours, see Sec. 6.5.3, and advertising policies, see Sec. 6.5.4;

e) function for vector integration in web advertising, see Sec. 6.6.2;

f) social position function in the social network, see Sec. 7.2.1;

g) commitment function to measure strength of associations in the social network, see Sec. 7.2.2, 7.2.3, and 7.5.4.

5. Experimental and formal (in the case of social position) analyses of profile and usability of the methods, patterns and measures presented.

8.2. Prior Verification of the Concepts

Prior to the publication of this monograph, the concepts and ideas elaborated therein were verified by a number of anonymous reviewers, since these were already presented at international conferences and in scietific journals.

Indirect association rules studied in Sec. 3 as well as their application to recommender systems have been proposed as new patterns at first in [Kaz04a, Kaz04b] then developed in [Kaz04g, Kaz05b, Kaz05d, Kaz05e], and finally in [Kaz *a*]. The IDARM algorithm for mining indirect association rules was proposed in [Kaz05b] whereas its modified version - IDARM* was introduced in [Kaz *a*].

The novel idea of both negative and positive verification of hyperlinks based on association rules derived from web logs, see Sec. 4, has been proposed for the first time in [Kaz06e] and then examined in [Kaz08f]. The latter also contains the PANAMA algorithm for mining both positive and negative association rules with

respect to its application to hyperlink verification. Negative association rules were combined with sequential patterns with negative conclusions in [Kaz07c, Kaz08b], see Sec. 5. Positive and negative patterns including simplified, 2-page sequential patterns extracted from usage data have been utilized for the filtering of content-based recommendation list in [Kaz07c]. Some selected applicable components of the concept were patented [Kaz06f].

The new concept of sequential patterns with negative conclusions, see Sec. 5, for only 2-page patterns was introduced in [Kaz07c], then extended for general patterns in [Kaz08b]. Finally, the SPAWN algorithm for mining sequential patterns with negative conclusions, see Sec. 5.3.2, was published in [Kaz08a]. Besides, their application to recommender systems was proposed in [Kaz07c], see Sec. 5.5, as well as to hyperlink verification in [Kaz08b], see Sec. 5.6.

The new solutions presented in Sec. 6 have been at first developed as the ROSA system [Kaz03b, Kaz03c, Kaz04d], including some extensions [Kaz04e, Kaz05c, Kaz06a]. Next, the AdROSA system - personalized, adaptive web advertising method was proposed in [Kaz04c, Kaz05a], and finally presented in the most comprehensive way in [Kaz07a]. The survey on recommender systems that also contains relations between recommender systems and various data mining methods was published in [Kaz08g].

The results of the research on social network analysis and the application of associations (relationships) for evaluation of social position of individuals were published in a number of papers [Kaz06b, Kaz06c, Kaz07d, Kaz07f, Kaz08e, Kaz08h, Mus08a, Mus08d], especially in [Kaz07e, Kaz07g, Kaz08c, Kaz *b*, Kaz *c*]. Some elements of the concept were also patented [Kaz06d].

8.3. Possible Extensions

All the concepts presented in the monograph can be the subject of further research. In particular new application areas can be studied. This especially refers to negative patterns, which actually can help to verify any previously known knowledge (associations) obtained using other methods and data sources.

Indirect association rules can be considered with respect to some other measures different from indirect confidence, see Sec. 3, e.g. using measures enumerated in [Tan02c]. It also refers to their possible combinations.

Sequential patterns with negative conclusions can be further studied, in particular in the experimental way, in terms of their usability in various application domains.

Social position function can be engaged to extract key persons in the management of working teams as well as in viral marketing. However, it requires some additional analysis.

A very interesting extension of almost all concepts presented in the monograph is the dynamics of analyzed patterns and measures in the evolving environments. Additionally, some incremental approaches for pattern extraction could be very useful in the case of large and complex systems.

References

[Abe99] Abe N., Nakamura A., *Learning to Optimally Schedule Internet Banner Advertisements.* 16th International Conference on Machine Learning, ICML 1999, Morgan Kaufmann, 1999, 12–21.

[Ada03] Adamic L.A., Adar E., *Friends and Neighbors on the Web.* Social Networks, 25(3), 2003, 211–230.

[Ada04] Adams R., *Intelligent Advertising.* AI & Society 18(1), 2004, 68–81.

[Ado01] Adomavicius G., Tuzhilin A., *Using Data Mining Methods to Build Customer Profiles.* IEEE Computer, 34(2), 2001, 74–82.

[Ado05] Adomavicius G., Tuzhilin A., *Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions.* IEEE Transactions on Knowledge and Data Engineering, 17(6), 2005, 734–749.

[Aff03] *Affiliate Program Reviews at ClickQuick.com*, 2003, http://www.clickquick.com.

[Agg98] Aggarwal C.C., Wolf J.L., Yu P.S., *A Framework for the Optimizing of WWW Advertising.* Trends in Distributed Systems for Electronic Commerce, International IFIP/GI Working Conference, TREC'98, LNCS 1402, Springer, 1998, 1–10.

[Agr93] Agrawal R., Imieliński T., Swami A., *Mining association rules between sets of items in large databases.* The 1993 ACM SIGMOD International Conference on Management of Data, ACM Press, 1993, 207–216.

[Agr94] Agrawal R., Srikant R., *Fast Algorithms for Mining Association Rules.* The 20th International Conference on Very Large Databases, VLDB, 1994, 487–499.

[Agr95] Agrawal R., Srikant R., *Mining Sequential Patterns.* The Eleventh International Conference on Data Engineering ICDE, IEEE Computer Society, 1995, 3–14.

[Agr96a] Agrawal R., Mannila H., Srikant R., Toivonen H., Verkamo I., *Fast Discovery of Association Rules.* Chapter 12 in Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R. (eds.), *Advances in Knowledge Discovery and Data Mining.* AAAI Press/The MIT Press, Menlo_ark - Cambridge, 1996.

[Agr96b] Agrawal R., Shafer J.C., *Parallel Mining of Association Rules.* IEEE Transactions on Knowledge and Data Engineering, 8(6), 1996, 962–969.

[Ale63] Alexander C.N., *A method for processing sociometric data.* Sociometry, 26, 1963, 268–269.

[Ala05] Alataş B., Akin E., *An efficient genetic algorithm for automated mining of both positive and negative quantitative association rules.* Soft Computing – A Fusion of Foundations, Methodologies and Applications, 10(3), 2005, 230–237.

[Ami03] Amiri A., Menon S., *Efficient Scheduling of Internet Banner Advertisements.* ACM Transactions on Internet Technology, 3(4), 2003, 334–346.

[Ant04a] Antonie M.-L., Zaïane O.R., *An associative classifier based on positive and negative rules.* The 9th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, DMKD 2004, ACM Press, 2004, 64–69.

[Ant04b] Antonie M.-L., Zaïane O.R., *Mining Positive and Negative Association Rules: An Approach for Confined Rules.* PKDD 2004, LNCS 3202, Springer, 2004, 27–38.

[Ard00] Ardissono L., Goy A., *Tailoring the Interaction With Users in Web Stores.* User Modeling and User-Adapted Interaction, 10(4), Kluwer Academic Publishers, 2000, 251–303.

[Ayr02] Ayres J., Gehrke J.E., Yiu T., Flannick J., *Sequential Pattern Mining Using Bitmaps.* Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2002, ACM Press, 2002, 429–435.

[Bae03] Bae S.M., Park S.C., Ha S.H., *Fuzzy Web Ad Selector Based on Web Usage Mining.* IEEE Intelligent Systems, 18(6), 2003, 62–69.

[Bar07] Baraglia R., Silvestri F., *Dynamic personalization of web sites without user intervention.* Communication of the ACM, 50(2), 2007, 63–67.

[Bau97] Baudisch P., Leopold D., *User-configurable Advertising Profiles Applied to Web Page Banners.* First Berlin Economics Workshop, Berlin, Germany, 1997, http://patrickbaudisch.com/publications/1997-Baudisch-Berlin-UserConfigurableAdvertisingProfiles.pdf.

[Bav50] Bavelas A., *Communication patterns in task-oriented groups.* Journal of the Acoustical Society of America, 22, 1950, 271–282.

[Ber05] Berkhin P., *A Survey on PageRank Computing.* Internet Mathematics, 2(1), 2005, 73–120.

[Bil00] Billsus D., Pazzani M., *User Modeling for Adaptive News Access.* User Modeling and User-Adapted Interaction, 10(2–3), 2000, 147–180.

[Bil03] Bilchev G., Marston D., *Personalised Advertising – Exploiting the Distributed User Profile.* BT Technology Journal, 21(1), 2003, 84–90.

[Bol99] Boley D., Gini M., Gross R., Han E.H., Hastings K., Karypis G., Kumar V., Mobasher B., Moorey J., *Document Categorization and Query Generation on the World Wide Web Using WebACE.* Artificial Intelligence Review, 13(5–6), 1999, 365–391.

[Bon72] Bonacich P., *Factoring and weighting approaches to status scores and clique identification.* Journal of Mathematical Sociology, 2, 1972, 113–120.

[Bon87] Bonacich P., *Power and centrality: a family of measures.* American Journal of Sociology, 92, 1987, 1170–1182.

[Bon01] Bonacich P., Lloyd P., *Eigenvector-like Measures of Centrality for Asymmetric Relations.* Social Networks, 23(3), 2001, 191–201.

[Bor05] Borgatti S.P., *Centrality and network flow.* Social Networks, 27(1), 2005, 55–71.

[Bot92] Botafogo R.A., Rivlin E., Shneiderman B., *Structural analysis of hypertexts: identifying hierarchies and useful metrics.* ACM Transaction on Information Systems, 10(2), 1992, 142–180.

[Bou00] Boulicaut J.-F., Bykowski A., Jeudy B., *Towards the Tractable Discovery of Association Rules with Negations.* The Fourth International Conference on Flexible Query Answering Systems, FQAS 2000, Physica-Verlag, 425–434.

[Boy04] Boyd D.M., *Friendster and Publicly Articulated Social Networking.* CHI 2004, ACM Press, 2004, 1279–1282.

[Bri98] Brin S., Page L., *The Anatomy of a Large-Scale Hypertextual Web Search Engine.* WWW7, 1998, also Computer Networks and ISDN Systems, 30(1–7), 1998, 107–117.

[Bri06] Brinkmeier M., *PageRank Revisited.* ACM Transactions on Internet Technology, 6(3), 2006, 282–301.

[Buo01] 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.

[Car05] Carrington P.J., Scott J., Wasserman S., *Models and Methods in Social Network Analysis.* Cambridge University Press, New York, 2005.

[Chak99] Chakrabarti S., Dom B.E., Kumar S.R., Raghavan P., Rajagopalan S., Tomkins A., Gibson D., Kleinberg J., *Mining the Web's link structure.* Computer, 32(8), 1999, 60–67.

[Chan06] Chang L., Yang D., Tang S., Wang T., *Mining Compressed Sequential Patterns.* Second International Conference on Advanced Data Mining and Applications, ADMA 2006, LNCS 4093 Springer, 2006, 761–768.

[Chan01] Chang W.-K., Chuang M.-H., *Validating Hyperlinks by the Mobile-Agent Approach.* Tunghai Science, 3, 2001, 97–112.

[Chen03] Chen Z., Fu A. W.-C., Tong F.C.-H., *Optimal Algorithms for Finding User Access Sessions from Very Large Web Logs.* World Wide Web: Internet and Web Information Systems, 6(3), 2003, 259–279.

[Chen04] Cheng H., Yan X., Han J., *IncSpan: incremental mining of sequential patterns in large database.* The Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004, ACM Press, 2004, 527–532.

[Chen05] Chen L., Bhowmick S.S., Chia L.-T., *Mining Positive and Negative Association Rules from XML Query Patterns for Caching.* 10th International Conference on Database Systems for Advanced Applications, DASFAA 05, LNCS 3453, Springer, 2005, 736–747.

[Chen06] Chen L., Bhowmick S.S., Li J., *Mining Temporal Indirect Associations.* 10th Pacific–Asia Conference, PAKDD 2006, LNCS 3918, Springer Verlag, 2006, 425–434.

[Chen07] Chen Y., Guo J., Wang Y., Xiong Y., Zhu Y., *Incremental Mining of Sequential Patterns Using Prefix Tree.* 11th Pacific–Asia Conference, PAKDD 2007, LNCS 4426 Springer, 2007, 433–440.

[Cheu96] Cheung D.W.L., Han J., Ng V., Wong C.Y., *Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique.* Twelfth International Conference on Data Engineering, IEEE Computer Society, 1996, 106–114.

[Cheu97] Cheung D.W.L., Lee S.D., Kao B., *A General Incremental Technique for Maintaining Discovered Association Rules.* Fifth International Conference on Database Systems for Advanced Applications (DASFAA), Advanced Database Research and Development Series 6 World Scientific, 1997, 185–194.

[Chi03] Chickering D.M., Heckerman D., *Targeted Advertising with Inventory Management.* 2nd ACM Conference on Electronic Commerce, EC00 (2000), 145–149; Interfaces, 33(5), 2003, 71–77.

[Cho02] Cho Y.H., Kim J.K., Kim S.H., *A personalized recommender system based on web usage mining and decision tree induction.* Expert Systems with Application, 23(3), 2002, 329–342.

[Cho05] *ChoiceStream Personalization Survey: Consumer Trends and Perceptions.* ChoiceStream, 2005, http://www.choicestream.com/pdf/ChoiceStream_PersonalizationSurveyResults2005.pdf.

[Chu04] Chun J., Oh J.-Y., Kwon S., Kim D., *Simulating the Effectiveness of Using Association Rules for Recommendation Systems.* AsiaSim 2004, Revised Selected Papers, LNCS 3398, Springer Verlag, 2005, 306–314.

[Cli03] *Going Beyond the Banner,* ClickQuick.com, 2003, http://www.clickquick.com/articles/beyond.htm.

[Con05] Cong S., Han J., Padua D.A., *Parallel mining of closed sequential patterns.* The 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2005, ACM Press, 2005, 562–567.

[Coo99] Cooley R., Mobasher B., Srivastava J., *Data Preparation for Mining World Wide Web Browsing Patterns*. Knowledge and Information Systems, 1(1), 1999, 5–32.

[Cor06] Cornelis C., Yan P., Zhang X., Chen G., *Mining Positive and Negative Association Rules from Large Databases*. IEEE International Conference on Cybernetics and Intelligent Systems, CIS 2006, IEEE Computer Society, 2006, 613–618.

[Cow00] Cowderoy A.J.C., *Measures of size and complexity for web-site content*. 11th European Software Control and Metrics Conference (ESCOM), Munich, Germany, 2000, 423–431.

[Cul04] Culotta A., Bekkerman R., McCallum A., *Extracting social networks and contact information from email and the Web*. First Conference on Email and Anti-Spam, 2004.

[Cun01] Cunningham P., Bergmann R., Schmitt S., Traphöner R., Breen S., Smyth B., *WEB-SELL: Intelligent sales assistants for the World Wide Web*. KI - Künstliche Intelligenz, 1, 2001, 28–32.

[Dan01] Daniłowicz C., Baliński J., *Document ranking based upon Markov chains*. Information Processing and Management, 37(4), 2001, 623–637.

[Daw03] Dawande M., Kumar S., Sriskandarajah C., *Performance Bounds of Algorithms for Scheduling Advertisements on a Web Page*. Journal of Scheduling, 6(4), 2003, 373–394.

[Dir02] *Directive on privacy and electronic communications*. Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector, 2002, http://europa.eu.int/eur-lex/pri/en/oj/dat/2002/l_201/l_20120020731en00370047.pdf.

[Don05] Donoho S., *Link Analysis*. Chap. 19 in Maimon O., Rokach L. (eds.), *The Data Mining and Knowledge Discovery Handbook*. Springer, 2005.

[Don06] Dong X., Sun F., Han X., Hou R., *Study of Positive and Negative Association Rules Based on Multi-confidence and Chi-Squared Test*. The Second International Conference on Advanced Data Mining and Applications, ADMA 2006, LNCS 4093, Springer, 2006, 100–109.

[Don07] Dong X., Niu Z., Shi X., Zhang X., Zhu D., *Mining Both Positive and Negative Association Rules from Frequent and Infrequent Itemsets*. Third International Conference on Advanced Data Mining and Applications, ADMA 2007, LNCS 4632, Springer, 2007, 122–133.

[Dou04] *Online Advertising*. DoubleClick Inc., 2004, http: //www.doubleclick.com/us/products /online_advertising/.

[Dum00] Dumais S.T., Chen H., *Hierarchical classification of Web content*. The 23rd ACM SIGIR Conference, ACM Press, 2000, 256–263.

[Fag03] Fagin R., Kumar R., Sivakumar D., *Comparing Top k Lists*. SIAM Journal on Discrete Mathematics Vol. 17, No. 1, 2003, 134–160.

[Fla00] Flake G., Lawrence S., Giles C.L., *Efficient identification of web communities*. The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, 2000, 150–160.

[For01] Fortez I., Balcázar J.L., Bueno R.M., *Bounding Negative Information in Frequent Sets Algorithms*. 4th International Conference on Discovery Science, DS 2001, Washington, DC, USA, November 25–28, 2001, LNCS 2226, Springer, 2001, 50–58.

[Fre77] Freeman L.C., *A set of measures of centrality based on betweenness*. Sociometry, 40, 1977, 35–41.

[Fre79] Freeman L.C., *Centrality in social networks: Conceptual clarification*. Social Networks, 1(3), 1979, 215–239.

[Fre80] Freeman L.C., Roeder D., Mulholland R.R., *Centrality in Social Networks: II. Experimental Results*. Social Networks, 2(2), 1980, 119–141.

[Fri98] Friedkin N.E., *A Structural Theory of Social Influence*. Cambridge University Press, Cambridge, 1998.

[Fri97] Friedkin N.E., Johnsen E.C., *Social positions in influence networks*. Social Networks, 19(3), 1997, 209–222.

[Gan06] Gan M., Zhang M., Wang S., *Extended Negative Association Rules and the Corresponding Mining Algorithm*. 4th International Conference on Advances in Machine Learning and Cybernetics, ICMLC 2005, Revised Selected Papers, LNCS 3930, Springer, 2006, 159–168.

[Gar99] Garofalakis M.N., Rastogi R., Shim K., *SPIRIT: Sequential Pattern Mining with Regular Expression Constraints*. 25th International Conference on Very Large Data Bases, VLDB'99, Morgan Kaufmann, 1999, 223–234.

[Gar06] García-Hernández R.A., Trinidad J.F.M., Carrasco-Ochoa J.A., *A New Algorithm for Fast Discovery of Maximal Sequential Patterns in a Document Collection*. 7th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing 2006, LNCS 3878, Springer Verlag, 2006, 514–523.

[Gart97] Garton L., Haythorntwaite C., Wellman B., *Studying Online Social Networks*. Journal of Computer-Mediated Communication, 3(1), 1997, http://jcmc.indiana.edu/vol3/issue1/garton.html.

[Ger03] Géry M., Haddad M.H., *Evaluation of web usage mining approaches for user's next request prediction*. WIDM 2003, ACM Press, 2003, 74–81.

[Gey02] Geyer-Schulz A., Hahsler, M., *Comparing Two Recommender Algorithms with the Help of Recommendations by Peers*. WEBKDD 2002, Revised Papers, LNCS 2703, Springer Verlag, 2003, 137–158.

[Gib98] Gibson D., Kleinberg J., Raghavan P., *Inferring Web communities from link topology*. Hypertext 1998, The Ninth ACM Conference on Hypertext and Hypermedia, ACM Press, 1998, 225–234.

[Gol04] Golbeck J., Hendler J.A., *Accuracy of Metrics for Inferring Trust and Reputation in Semantic Web-Based Social Networks*. 14th International Conference Engineering Knowledge in the Age of the Semantic Web 2004, LNCS 3257, Springer Verlag, 2004, 116–131.

[Goo01] Goodrum A., McCain K.W., Lawrence S., Giles C.L., *Scholarly publishing in the Internet age: a citation analysis of computer science literature*. Information Processing and Management 37(5), 2001, 661–675.

[Goo08] *Google Advertising Programs*, 2008, http://www.google.com/ads/.

[Gwi01] Gwiazda K., Kazienko P., *XLink – the future of document linking*. Information Systems Architecture and Technology ISAT 2001. Conference proceedings, Wrocław University of Technology, 2001, 132–139.

[Ha02] Ha S.H., *Helping Online Customers Decide through Web Personalization*. IEEE Intelligent Systems, 17(6), 2002, 34–43.

[Haf00] Haffner E. G., Heuer A., Roth U., Engel T., Meinel C., *Advanced Studies on Link Proposals and Knowledge Retrieval of Hypertexts with CBR*. EC-Web 2000, LNCS 1875, Springer Verlag, 2000, 369–378.

[Hao01] Hao M.C., Hsu M., Dayal U., Wei S.F., Sprenger T., Holenstein T., *Market basket analysis visualization on a spherical surface*. SPIE Vol. 4302, Visual Data Exploration and Analysis VIII, 2001, 227–233, http://www.hpl.hp.com/techreports/2001/HPL-2001-3.pdf.

[Ham04] Hamano S., Sato M., *Mining Indirect Association Rules*. ICDM 2004, LNCS 3275, Springer Verlag, 2004, 106–116.

[Han00] Han J., Pei J., Yin Y., *Mining Frequent Patterns without Candidate Generation*. ACM SIGMOD International Conference on Management of Data, ACM Press, 2000, 1–12.

[Han01] Han J., Kamber M., *Data Mining Concepts and Techniques*. San Francisco, Morgan Kaufmann, 2000.

[Han04] Han J., Pei J., Yin Y., Mao R., *Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach*. Data Mining and Knowledge Discovery, 8(1), 2004, 53–87.

[Han06] Hanneman R., Riddle M., *Introduction to social network methods, online textbook*. available at http://faculty.ucr.edu/~hanneman/nettext/ (01.04.2006).

[Hen01] Henzinger M.R., *Hyperlink Analysis for the Web*. IEEE Internet Computing, 5(1), 2001, 45–50.

[Her00] Herlocker J.L., Konstan J.A., Riedl J., *Explaining Collaborative Filtering Recommendations*. CSCW 2000, ACM Press, 2000, 241–250.

[Ho06] Ho C.-C., Li H.-F., Kuo F.-F., Lee S.-Y., *Incremental Mining of Sequential Patterns over a Stream Sliding Window*. 6th IEEE International Conference on Data Mining, ICDM 2006, IEEE Computer Society, 2006, 677–681.

[Hoe05] Höppner F., *Association rules*. Chapter 16 in Maimon O., Rokach L. (eds.), *The Data mining and knowledge Discovery Handbook*. Springer, 2005.

[HTTP92] *HTTP Requests Fields*. W3C, http://www.w3.org/Protocols/HTTP/HTRQ_Headers.html, 1992.

[Hua08] Huang Y., Zhang L., Zhang P., *A Framework for Mining Sequential Patterns from Spatio-Temporal Event Data Sets*. IEEE Transaction on Knowledge and Data Engineering, 20(4), 2008, 433–448.

[Hun98] Hunt M., *Direct-to-Consumer Advertising of Prescription Drugs*. National Health Policy Forum, April 1998, http://nhpf.ags.com/pdfs_bp/BP_DTC_4-98.pdf.

[Iye05] Iyer G., Soberman D., Villas-Boas J.M., *The Targeting of Advertising*. Marketing Science, 24(3), 2005, 461–476.

[Jef88] Jeffreys H., Jeffreys B.S., *Increasing and Decreasing Functions*. §1.065 in *Methods of Mathematical Physics*, 3rd ed., Cambridge University Press, Cambridge, England, 1988, 22.

[Jus08a] Juszczyszyn K., Kazienko P., Musiał K., *Local Topology of Social Network Based on Motif Analysis*. 12th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, KES 2008, Springer Verlag, LNAI 5178, 2008, 97–105.

[Jus08b] Juszczyszyn K., Kazienko P., Musiał K., Gabryś B., *Temporal Changes in Connection Patterns of an Email-based Social Network*. The 2008 IEEE/WIC/ACM International Conference on Web Intelligence (WI'08) and Intelligent Agent Technology (IAT'08), 2nd Workshop on Collective Intelligence in Semantic Web and Social Networks (CISWSN 2008), IEEE Computer Society Press, WI-IAT Workshops 2008, 9–12.

[Kaj07] Kajdanowicz T., Kazienko P., Musiał K., *Discovering Multidimensional Social Communities in Web 2.0*. The 3rd European Symposium on Nature-inspired Smart Information Systems, St Julians, Malta, November 26–28, 2007.

[Kat53] Katz L., *A new status derived from sociometrics analysis*. Psychometrica, 18, 1953, 39–43.

[Kaz00] Kazienko P., *Grupowanie dokumentów hipertekstowych na podstawie drzewa maksymalnych przepływów. Praca doktorska* (*Clustering of hypertext documents based on flow equivalent trees. Ph.D Thesis*), in Polish. Department of Information Systems, Wrocław University of Technology, PRE 31, 2000, http://www.zsi.pwr.wroc.pl/~kazienko/pub/Ph.D.Thesis2000/PhD.zip.

[Kaz01] Kazienko P., *Strukturalne podobieństwo dokumentów hipertekstowych* (*Structural similarity of hypertext documents*), in Polish. Informacja Wiedza Gospodarka. Prace PTIN nr 4, Polskie Towarzystwo Informacji Naukowej, Warszawa 2001, 244–253.

[Kaz02a] Kazienko P., Gwiazda K., *XML na poważnie (Taking XML seriously)*, in Polish. Helion, Gliwice 2002.

[Kaz02b] Kazienko P., *Dylematy języka XML (XML dilemmas)*, in Polish. Tele.net Forum, 11/2002, 36-39.

[Kaz02c] Kazienko P., *Rodzina języków XML (The XML family)*, in Polish. Software 2.0, no. 6(90), czerwiec 2002, 22-27.

[Kaz02d] Kazienko P., Zgrzywa M., *Języki wyszukiwania w dokumentach XML (Retrieval language for XML documents)*, in Polish. MiSSI 2002. Multimedialne i Sieciowe Systemy Informacyjne. Conference proceedings, Wrocław, 2002, 391-400.

[Kaz02e] Kazienko P., Zgrzywa M., *The Evolution of the Information Retrieval Languages for XML Documents*. Information Systems Applications and Technology ISAT 2003 Seminar. Proc. of the 24th Int. Scientific School. Wrocław University of Technology, 2003, 92-99.

[Kaz03a] Kazienko P., *Co tam panie w XML-u? (What's going on in XML)*, in Polish. Software 2.0 czerwiec, nr 6(102) 2003, 26-30.

[Kaz03b] Kazienko P., Kiewra M., *ROSA - Multi-agent System for Web Services Personalization*. First Atlantic Web Intelligence Conference Proceedings, AWIC'03, LNAI 2663, Springer Verlag, 2003, 297-306.

[Kaz03c] Kazienko P., Kiewra M., *Link Recommendation Method Based on Web Content and Usage Mining*. New Trends in Intelligent Information Processing and Web Mining, IIS: IIPWM'03, Advances in Soft Computing, Springer Verlag, 2003, 529-534.

[Kaz03d] Kazienko P., Litwin M., *On Using Topic Maps for Knowledge Representation*. Information Systems Applications and Technology ISAT 2003 Seminar. Proc. of the 24th International Scientific School. Wrocław University of Technology, 2003, 100-107.

[Kaz04a] Kazienko P., *Multi-agent Web Recommendation Method Based on Indirect Association Rules*. 8th International Conference on Knowledge-Based Intelligent Information & Engineering Systems KES'2004, Springer Verlag, LNAI 3214, 2004, 1157-1164.

[Kaz04b] Kazienko P., *Product Recommendation in E-Commerce Using Direct and Indirect Confidence for Historical User Sessions*. 7th International Conference on Discovery Science DS'04, LNAI 3245, Springer Verlag, 2004, 255-269.

[Kaz04c] Kazienko P., Adamski M., *Personalized Web Advertising Method*. The Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Adaptive Hypermedia, AH 2004, LNCS 3137, Springer Verlag, 2004, 146-155.

[Kaz04d] Kazienko P., Kiewra M., *Personalized Recommendation of Web Pages*. Chapter 10, in: T. Nguyen (ed.), *Intelligent Technologies for Inconsistent Knowledge Processing*, Advanced Knowledge International, Adelaide, South Australia, 2004, 163-183.

[Kaz04e] Kazienko P., Kiewra M., *Integration of Relational Databases and Web Site Content for Product and Page Recommendation*. 8th International Database Engineering & Applications Symposium, IDEAS '04, IEEE Computer Society, 2004, 111-116.

[Kaz04f] Kazienko P., Sobecki J., *XML-based Learning Scenario Representation and Presentation in the Adaptive E-learning Environment*. The 17th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems, IEA/AIE 2004, LNAI 3029, Springer Verlag, 2004, 967-976.

[Kaz04g] Kazienko P., Matrejek M., *Parameters for Mining Indirect Associations in the Web Environment*. Multimedia and Network Information Systems Conference MiSSI 2004, Vol. II, 2004, Politechnika Wrocławska, 201-211.

[Kaz05a] Kazienko P., *Multi-agent System for Web Advertising*. 9th International Conference on Knowledge-Based Intelligent Information & Engineering Systems, KES'2005, LNAI 3682, Springer Verlag, 2005, 507-513.

[Kaz05b] Kazienko P., *IDARM - Mining of Indirect Association Rules.* New Trends in Intelligent Information Processing and Web Mining, IIS: IIPWM'05, Advances in Soft Computing, Springer Verlag, 2005, 77–86.

[Kaz05c] Kazienko P., Kołodziejski P., *WindOwls - Adaptive System for the Integration of Recommendation Methods in E-commerce.* Third Atlantic Web Intelligence Conference, AWIC 2005, LNAI 3528, Springer Verlag, 2005, 218–224.

[Kaz05d] Kazienko P., Kuźmińska K., *The Influence of Indirect Association Rules on Recommendation Ranking Lists.* 5th International Conference on Intelligent Systems Design and Applications, ISDA 2005, RAAWS 2005, IEEE Computer Society, 2005, 482–487.

[Kaz05e] Kazienko P., Matrejek M., *Adjustment of Indirect Association Rules for the Web.* 31st Conference on Current Trends in Theory and Practice of Computer Science, SOFSEM 2005, Springer Verlag, LNCS 3381, 2005, 211–220.

[Kaz06a] Kazienko P., Kołodziejski P., *Personalized Integration of Recommendation Methods for E-commerce.* International Journal of Computer Science & Applications, 3(3), August 2006, 12–26.

[Kaz06b] Kazienko P., Musiał K., *Recommendation Framework for Online Social Networks.* AWIC 2006, The 4th Atlantic Web Intelligence Conference, Studies in Computational Intelligence, Vol. 23, Springer Verlag, 2006, 111–120.

[Kaz06c] Kazienko P., Musiał K., *Social Capital in Online Social Networks.* 10th International Conference on Knowledge-Based Intelligent Information & Engineering Systems, KES 2006, LNAI 4252, Springer Verlag, 2006, 417–424.

[Kaz06d] Kazienko P., Musiał K., *Sposób i układ do wyznaczania pozycji społecznej osoby w sieci społecznej (A Method and System for Estimation of Individual's Social Position in the Social Network).* Polish patent application, P380794, 2006.

[Kaz06e] Kazienko P., Pilarczyk M., *Hyperlink Assessment Based on Web Usage Mining.* The 7th Conference on Hypertext and Hypermedia, HT'06, ACM Press, 2006, 85–88.

[Kaz06f] Kazienko P., Pilarczyk M., *Sposób i urządzenie do klasyfikowania odsyłaczy hipertekstowych i prezentowania stron internetowych wraz z odsyłaczami (A Method and Apparatus for Classification of Hyperlinks and Presentation of Web Pages with Hyperlinks).* Polish patent application P380330, 2006.

[Kaz07a] Kazienko P., Adamski M., *AdROSA - Adaptive Personalization of Web Advertising.* Information Sciences, 177(11), 2007, 2269–2295.

[Kaz07b] Kazienko P., *Expansion of Telecommunication Social Networks.* The Fourth International Conference on Cooperative Design, Visualization and Engineering, CDVE 2007, LNCS 4674, Springer Verlag, 2007, 404–412.

[Kaz07c] Kazienko P., *Filtering of Web Recommendation Lists Using Positive and Negative Usage Patterns.* The 11th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, KES2007, RAAWS 2007, Springer Verlag, LNAI 4694, Part III, 2007, 1016–1023.

[Kaz07d] Kazienko P., Musiał K., *On Utilizing Social Networks to Discover Representatives of Human Communities.* International Journal of Intelligent Information and Database Systems, Special Issue on Knowledge Dynamics in Semantic Web and Social Networks, 1(3/4), 2007, 293–310.

[Kaz07e] Kazienko P., Musiał K., *Assessment of Personal Importance Based on Social Networks.* The 6th Mexican International Conference on Artificial Intelligence, MICAI 2007, LNAI 4827, Springer Verlag, 2007, 529–539.

[Kaz07f] Kazienko P., Musiał K., Przybycin A., Zgrzywa A., *Concepts and Applications of Social Network Analysis in Telecommunication.* The 28th International Scientific School Infor-

mation Systems Architecture and Technology, ISAT'2007, Biblioteka Informatyki Szkół Wyższych, Politechnika Wrocławska, 2007, 71-77.

[Kaz07g] Kazienko P., Musiał K., Zgrzywa A., *Evaluation of Node Position Based on Mutual Interaction in Social Network of Internet Users.* Technologie przetwarzania danych, II Krajowa Konferencja Naukowa, TPD 2007, Wydawnictwo Politechniki Poznańskiej, 2007, 265-276.

[Kaz08a] Kazienko P., *Mining Sequential Patterns with Negative Conclusions.* 10th International Conference on Data Warehousing and Knowledge Discovery, DaWaK 2008, LNCS 5182, Springer, 2008, 423-432.

[Kaz08b] Kazienko P., *Usage-Based Positive and Negative Verification of User Interface Structure.* The Fourth International Conference on Autonomic and Autonomous Systems, ICAS 2008, IEEE Computer Society, 2008, 1-6.

[Kaz08c] Kazienko P., Musiał K., *Mining Personal Social Features in the Community of Email Users.* 34th Conference on Current Trends in Theory and Practice of Computer Science, SOFSEM 2008, LNCS 4910, Springer, 2008, 708-719.

[Kaz08d] Kazienko P., Musiał K., Juszczyszyn K., *Recommendation of Multimedia Objects based on Similarity of Ontologies.* 12th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, KES 2008, Springer Verlag, Part I, LNAI 5177, 2008, 194-201.

[Kaz08e] Kazienko P., Musiał K., Kajdanowicz T., *Profile of the Social Network in Photo Sharing Systems.* 14th Americas Conference on Information Systems, AMCIS 2008, Minitrack: Social Network Analysis in IS Research, August 14-17, 2008, Toronto, Canada, Association for Information Systems (AIS), ISBN: 978-0-615-23693-3.

[Kaz08f] Kazienko P., Pilarczyk M., *Hyperlink Recommendation Based on Positive and Negative Association Rules.* New Generation Computing 26(3), May 2008, 227-244.

[Kaz08g] Kazienko P., *Web-based Recommender Systems and User Needs - the Comprehensive View.* 6th International Conference on Multimedia & Network Information Systems, MiSSI 2008, Wrocław, Poland, September 18-19, 2008, IOS Press, 2008, 243-258.

[Kaz08h] Kazienko P., Kołodziejski P., *Adaptive System for the Integration of Recommendation Methods with Social Filtering Enhancement.* 6th International Conference on Multimedia & Network Information Systems, MiSSI 2008, Wrocław, Poland, September 18-19, 2008, IOS Press, 2008, 291-296.

[Kaz *a*] Kazienko P., *Mining Indirect Association Rules for Web Recommendation.* International Journal of Applied Mathematics and Computer Science, in press.

[Kaz *b*] Kazienko P., Musiał K., Zgrzywa A., *Evaluation of Node Position Based on Email Communication.* Control & Cybernetics, accepted.

[Kaz *c*] Kazienko P., Musiał K., *Social Position of Individuals in Virtual Social Networks.* Journal of Mathematical Sociology, under review.

[Ken48] Kendall M.G., *Rank correlation methods.* Charles Griffin & Company, Ltd., London, 1948.

[Kob02] Kobsa A., *Personalized Hypermedia and International Privacy.* Communications of the ACM, 45(5), 2002, 64-67.

[Kob03] Kobsa A., Schreck J., *Privacy Through Pseudonymity in User-Adaptive Systems.* ACM Transactions on Internet Technology, 3(2), 2003, 149-183.

[Koh07] Koh Y.S., Pears R., *Efficiently Finding Negative Association Rules Without Support Threshold.* 20th Australian Joint Conference on Artificial Intelligence, AI 2007, LNCS 4830, Springer, 2007, 710-714.

[Kop07] Kopel M., Kazienko P., *Application of Agent-based Personal Web of Trust to Local Document Ranking*. The 1st KES Symposium on Agent and Multi-Agent Systems - Technologies and Applications, KES AMSTA 2007, LNAI 4496, Springer Verlag, 2007, 288-297.

[Kru97] Krulwich B., *Lifestyle Finder: Intelligent User Profiling Using Large-Scale Demographic Data*. AI Magazine, 18(2) AAAI, 1997, 37-45.

[Kry05] Kryszkiewicz M., Cichoń K., *Support oriented discovery of generalized disjunction-free representation of frequent patterns with negation*. Advances in Knowledge Discovery and Data Mining, 9th Pacific-Asia Conference, PAKDD 2005, LNCS 3518, Springer, 2005, 672-682.

[Kum02] Kumar R., Raghavan P., Rajagopalan S., Tomkins A., *The Web and Social Networks*. IEEE Computer, 35(11), 2002, 32-36.

[Lai00] Lai H., Yang T.C., *A Group-based Inference Approach to Customized Marketing on the Web - Integrating Clustering and Association Rules Techniques*. 33rd Annual Hawaii International Conference on System Sciences (HICSS-33), Track 6: Internet and the Digital Economy, IEEE Computer Society, 2000.

[Lan99] Langheinrich M., Nakamura A., Abe N., Kamba T., Koseki Y., *Unintrusive Customization Techniques for Web Advertising*. Computer Networks, 31(11-16), 1999, 1259-1272.

[Lau00] Laur P.A., Masseglia F., Poncelet P., Teisseire M., *A General Architecture for Finding Structural Regularities on the Web*. AIMSA 2000, LNAI 1904, Springer Verlag, 2000, 179-188.

[Lau07] Laur P.-A., Symphor J.E., Nock R., Poncelet P., *Statistical supports for mining sequential patterns and improving the incremental update process on data streams*. Intelligent Data Analysis, 11(1), 2007, 29-47.

[Law99] Lawrence S., Giles C.L., Bollacker K., *Digital Libraries and Autonomous Citation Indexing*. IEEE Computer 32(6), 1999, 67-71.

[Law01] Lawrence R.D., Almasi G.S., Kotlyar V., Viveros M.S., Duri S.S., *Personalization of Supermarket Product Recommendations*. Data Mining & Knowledge Discovery, 5(1/2), 2001, 11-32.

[Lee01] Lee G., Lee K.L., Chen A.L.P., *Efficient Graph-Based Algorithms for Discovering and Maintaining Association Rules in Large Databases*. Knowledge and Information Systems, 3(3) Springer Verlag, 2001, 338-355.

[Lee02] Lee D., Choi H., *Collaborative Filtering System of Information on the Internet*. ICCS 2002, Part III, LNCS 2331, Springer Verlag, 2002, 1090-1099.

[Li07] Li H., Chen H., *GraSeq : A Novel Approximate Mining Approach of Sequential Patterns over Data Stream*. Advanced Data Mining and Applications, 3rd International Conference, ADMA 2007, LNCS 4632, Springer Verlag, 2007, 401-411.

[Lin02] Lin M.-Y., Lee S.-Y., *Fast discovery of sequential patterns by memory indexing*. 4th International Conference on Data Warehousing and Knowledge Discovery, DaWaK 2002, LNCS 2454, Springer, 2002 150-160.

[Lor71] Lorrain F., White H.C., *Structural Equivalence of Individuals in Social Networks*. Journal of Mathematical Sociology, 1, 1971, 49-80.

[Lu02] Lu M., Zhou Q., Fan L., Lu Y., Zhou L., *Recommendation of Web Pages Based on Concept Association*. WECWIS'02, IEEE Computer Society, 2002, 221-227.

[Lu03] Lu Z., Yao Y., Zhong N., *Web Log Mining*. Chapter 9 in Zhong N., Liu J., Yao Y. (eds.), *Web Intelligence*. Springer, Berlin, New York, 2003.

[Mad99] Madria S.K., Bhowmick S.S., Ng W.-K., Lim E.P., *Research Issues in Web Data Mining*. DaWaK'99, Springer Verlag, LNCS 1676, 1999, 303-312.

[Mai05] Maimon O., Rokach L. (eds.), *The Data Mining and Knowledge Discovery Handbook.* Springer, 2005.

[Mar02] Marchiori M. (ed.), *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification, W3C Recommendation 16 April 2002.* World Wide Web Consortium, 2002, http://www.w3.org/TR/P3P/.

[Mas03] Masseglia F., Poncelet P., Teisseire M., *Incremental mining of sequential patterns in large databases.* Data & Knowledge Engineering, 46(1), 2003, 97–121.

[McC98] McCandless M., *Web Advertising.* IEEE Intelligent Systems, 13(3), 1998, 8–9.

[McG01] McGovern G., Norton R., O'Dowd C., *Web Content Style Guide.* Financial Times Press, Prentice Hall, 2001.

[Meh05] Mehta A., Saberi A., Vazirani U.V., Vazirani V.V., *AdWords and Generalized On-line Matching.* 46th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2005, IEEE Computer Society, 2005, 264–273.

[Meh07] Mehta A., Saberi A., Vazirani U.V., Vazirani V.V., *AdWords and generalized online matching.* Journal of the ACM, 54(5), Art. 22, 2007.

[Mil02] Milo R., Shen-Orr S., Itzkovitz S., Kashtan N., Chklovskii D., Alon U., *Network motifs: simple building blocks of complex networks.* Science, 298, 2002, 824–827.

[Mob00a] Mobasher B., Cooley R., Srivastava J., *Automatic Personalization Based on Web Usage Mining.* Communications of the ACM, 43(8), 2000, 142–151.

[Mob00b] 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 Verlag, 2000, 156–176.

[Mob01] Mobasher B., Dai H., Luo T., Nakagawa M., *Effective personalization based on association rule discovery from web usage data.* WIDM 2001, ACM Press, 2001, 9–15.

[Mob02] Mobasher B., Dai H., Luo T., Nakagawa M., *Using sequential and non-sequential patterns in predictive Web usage mining tasks.* ICDM 2002, IEEE Computer Society, 2002, 669–672.

[Mon03] Montaner M., López B., de la Rosa J.L., *A Taxonomy of Recommender Agents on the Internet.* Artificial Intelligence Review, 19(4), Kluwer Academic Publishers, 2003, 285–330.

[Moo00] Mooney R.J., Roy L., *Content-based book recommending using learning for text categorization.* 5th ACM Conference on Digital Libraries, DL00, ACM Press, 2000, 195–204.

[Morz03] Morzy T., Zakrzewisz M., *Data mining.* Chapter 11 in Błażewicz J., Kubiak W., Morzy T., Rubinkiewicz M. (eds.), *Handbook on Data Management in Information Systems.* Springer Verlag, Berlin Heidelberg New York, 2003, 487–565.

[Morz06] Morzy M., *Efficient Mining of Dissociation Rules.* DaWaK 2006, LNCS 4081 Springer, 2006, 228–237.

[Mus08a] Musiał K., Kazienko P., Kajdanowicz T., *Multirelational Social Networks in Multimedia Sharing Systems.* Chapter 18 in N.T.Nguyen, G.Kołaczek, B.Gabryś (eds.) *Knowledge Processing and Reasoning for Information Society,* Academic Publishing House EXIT, Warszawa, 2008, 275–292.

[Mus08b] Musiał K., Juszczyszyn K., Kazienko P., *Ontology-based Recommendation in Multimedia Sharing Systems.* System Science, 34(1), 2008, 97–106.

[Mus08c] Musiał K. Juszczyszyn K., Gabryś B., Kazienko P.: *Patterns of Interactions in Complex Social Networks based on Coloured Motifs Analysis.* 15th International Conference on Neural Information Processing of the Asia-Pacific Neural Network Assembly, ICONIP 2008, November 25–28, 2008, Auckland, New Zealand, KEDRI, 450–451.

[Mus08d] Musiał K., Kazienko P., Kajdanowicz T., *Social Recommendations within the Multimedia Sharing Systems*. The First World Summit on the Knowledge Society, WSKS'08, September 24-28, 2008, Athens, Greece, Lecture Notes in Computer Science LNCS 5288, Springer, 2008, 364-372. Best Paper Award.

[Nak02] Nakamura A., *Improvements in Practical Aspects of Optimally Scheduling Web Advertising*. 11th Int. WWW Conference, WWW2002, ACM Press, 2002, 536-541.

[Nak03] Nakagawa M., Mobasher B., *Impact of Site Characteristics on Recommendation Models Based On Association Rules and Sequential Patterns*. IJCAI'03 Workshop on Intelligent Techniques for Web Personalization, 2003, http://maya.cs.depaul.edu/~mobasher/papers/NM03a.pdf.

[Nga03] Ngai E.W.T., *Selection of Web Sites for Online Advertising Using the AHP*. Information & Management, 40(4) Elsevier, 2003, 233-242.

[Nov00] Novak T.P., Hoffman D.L., *Advertising and Pricing Models for the Web*. In: D. Hurley, B. Kahin, H. Varian, (eds.), *Internet Publishing and Beyond: The Economics of Digital Information and Intellectual Property*, MIT Press, Cambridge, 2000.

[Pal05] Pal S., Bagchi A., *Association against dissociation: some pragmatic considerations for frequent itemset generation under fixed and variable thresholds*. SIGKDD Explorations, 7(2), 2005, 151-159.

[Paz97] Pazzani M., Billsus D., *Learning and revising user profiles: The identification of interesting web sites*. Machine Learning, 27, Springer Verlag, 1997, 313-331.

[Paz99] Pazzani M., *A Framework for Collaborative, Content-Based and Demographic Filtering*. Artificial Intelligence Review, 13(5-6) Kluwer Academic Publishers, 1999, 393-408.

[Pei01] Pei J., Han J., Mortazavi-Asl B., Pinto H., Chen Q., Dayal U., Hsu M., *PrefixSpan: Mining Sequential Patterns by Prefix-Projected Growth*. The 17th International Conference on Data Engineering, IEEE Computer Society, 2001, 215-224.

[Pei04] Pei J., Han J., Mortazavi-Asl B., Wang J., Pinto H., Chen Q., Dayal U., Hsu M., *Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach*. IEEE Transaction on Knowledge and Data Engineering, 16(11), 2004, 1424-1440.

[Per02] Perner P., Fiss G., *Intelligent E-marketing with Web Mining, Personalization, and User-Adapted Interfaces*. Advances in Data Mining: Applications in E-Commerce, Medicine, and Knowledge Management, LNAI 2394, Springer Verlag, 2002, 37-52.

[Pie03] Pierrakos D., Paliouras G., Papatheodorou C., Spyropoulos C.D., *Web Usage Mining as a Tool for Personalization: A Survey*. User Modeling and User-Adapted Interaction, 13(4), 2003, 311-372.

[Pla06] Plantevit M., Laurent A., Teisseire M., *HYPE: mining hierarchical sequential patterns*. ACM 9th International Workshop on Data Warehousing and OLAP, DOLAP 2006, ACM Press, 2006, 19-26.

[Pri05] Priebey C.E., Conroy J.M., Marchette D.J., Park Y. *Scan Statistics on Enron Graphs*. Computational & Mathematical Organization Theory, 3, 2005, 229-247.

[Pok04] Pokrywka P., Kazienko P., *Serwis porównujący ceny z wielu sklepów internetowych (WWW service with comparison of prices from many e-commerce)*, in Polish. Software 2.0, No. 6, 2004, 70-76.

[Pro51] Proctor C.H., Loomis C.P., *Analysis of sociometric data, in: Research Methods in Social Relations*. In: M. Jahoda, M. Deutch, S.W. Cok (eds.), Dryden Press, NewYork, 1951, 561-586.

[Puj02] Pujol J.M., Sangüesa R., Delgado J., *Extracting reputation in multi agent systems by means of social network topology*. AAMAS 2002, The First International Joint Conference on Autonomous Agents & Multiagent Systems, ACM Press, 2002, 467-474.

[Ran04] Rana O.F., Hinze A., *Trust and reputation in dynamic scientific communities.* IEEE Distributed Systems Online, 5(1), 2004, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1270714.

[Ras92] Rasmussen E., *Clustering Algorithms.* Chapter 16 in: W. Frakes, R. Baeza-Yates (eds.), *Information retrieval: data, structures & algorithms.* Englewood Cliffs, NJ, Prentice Hall, 1992, 419–442.

[Rea07] 24/7 Real Media, Inc., http://www.realmedia.com/.

[Ren06] Ren J.-D., Zhou X.-L., *An Efficient Algorithm for Incremental Mining of Sequential Patterns.* Advances in Machine Learning and Cybernetics, 4th International Conference, ICMLC 2005, LNCS 3930 Springer Verlag, 2006, 179–188.

[Rod02] Rodriguez M.G., *Automatic data-gathering agents for remote navigability testing.* IEEE Software, 19(6), 2002, 78–85.

[Rod04] Rodgers Z., *Volume Up, Click-Throughs Down in Q4 '03 Serving Report.* Jupitermedia Corporation, February 5, 2004, http://www.clickz.com/stats/markets/advertising/article.php/3309271.

[Rus06] Rusmevichientong P., Williamson D.P., *An adaptive algorithm for selecting profitable keywords for search-based advertising services.* The 7th ACM Conference on Electronic Commerce, ACM-EC 2006, ACM Press, 2006, 260–269.

[Sab66] Sabidussi G., *The centrality index of a graph.* Psychmetrica, 31(4), 1966, 581–603.

[Sal89] Salton G., *Automatic Text Processing. The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley, Reading, MA, 1989.

[Sav98] Savasere A., Omiecinski E., Navathe S.B., *Mining for strong negative associations in a large database of customer transactions.* 14th ICDE, IEEE Computer Society, 1998, 494–502.

[Sch01] Schafer J.B., Konstan J.A., Riedl J., *E-Commerce Recommendation Applications.* Data Mining and Knowledge Discovery, 5(1/2) Springer Verlag, 2001, 115–153.

[Sch02] Schein A.I., Popescul A., Ungar L.H., Pennock D.M., *Methods and Metrics for Cold-Start Recommendations.* 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval SIGIR 2002, ACM Press, 2002, 253–260.

[Sha54] Shaw M.E., *Group structure and the behaviour of individuals in small groups.* Journal of Psychology, 38, 1954, 139–149.

[Sha05] Sharma L.K., Vys O.P., Tiwary U.S., Vyas R., *A Novel Approach of Multilevel Positive and Negative Association Rule Mining for Spatial Databases.* 5th Industrial Conference on Data Mining ICDM´2005, LNCS 3587, Springer, 2005, 620–629.

[She05] Shetty J., Adibi J., *Discovering Important Nodes through Graph Entropy The Case of Enron Email Database,* LinkKDD '05, 3rd International Workshop on Link Discovery, ACM Press, 2005, 74–81.

[Spe87] Spearman C., *The proof and measurement of association between two things.* The American Journal of Psychology 15, 1904, 72–101, also 100(3/4) Special Centennial Issue, 1987, 441–471.

[Spi01] Spiliopoulou M., Pohle C., *Data Mining for Measuring and Improving the Success of Web Sites.* Data Mining and Knowledge Discovery, 5(1/2), 2001, 85–114.

[Sri96] Srikant, R. Agrawal, R., *Mining sequential patterns: Generalizations and performance improvements.* 5th International Conference on Extending Database Technology, EDBT, Vol. 1057. Springer-Verlag, 1996, 3–17.

[Sri01] Srikant R., Yang Y., *Mining web logs to improve website organization.* 10th International World Wide Web Conference, WWW 10, ACM Press, 2001, 430–437.

[Sul97] Sullivan T., *Reading Reader Reaction: A Proposal for Inferential Analysis of Web Server Log Files.* 3rd Conference on Human Factors and the Web, US West Communications, 1997.

[Tan00] Tan P.-N., Kumar V., Srivastava J., *Indirect Association: Mining Higher Order Dependencies in Data.* PKDD 2000, Springer Verlag, LNCS 1910, 2000, 632–637.

[Tan02a] Tan P.-N., Kumar V., *Discovery of Web Robot Sessions Based on their Navigational Patterns.* Data Mining and Knowledge Discovery, 6(1), 2002, 9–35.

[Tan02b] Tan P.-N., Kumar V., *Mining Indirect Associations in Web Data.* WEBKDD 2001. Springer Verlag, LNCS 2356, 2002, 145–166.

[Tan02c] Tan P.-N., Kumar V., Srivastava J., *Selecting the Right Interestingness Measure for Association Patterns.* ACM SIGKDD, ACM Press, 2002, 32–41.

[Tan03] Tan P.-N., Kumar V., *Discovery of Indirect Associations from Web Usage Data.* Chapter 7 in N. Zhong, J. Liu, Y.Y. Yao (eds.), *Web Intelligence.* Springer Verlag, 2003, 128–152.

[Tan06] Tan P.-N., Steinbach M., Kumar V., *Introduction to Data Mining,* Pearson Addison-Wesley, Pearson International ed., 2006.

[Tel97] *Teledienstedatenschutzgesetz (German Teleservices Data Protection Act),* 22 Juli 1997, http://www.datenschutz-berlin.de/recht/de/rv/tk_med/tddsg.htm.

[Tel04] Teltzrow M., Kobsa A., *Impacts of User Privacy Preferences on Personalized Systems: a Comparative Study.* In: C.-M. Karat, J. Blom and J. Karat (eds.), *Designing Personalized User Experiences for eCommerce.* Dordrecht, Netherlands: Kluwer Academic Publishers, 2004, 315–332.

[Ten02] Teng W., Hsieh M., Chen M., *On the mining of substitution rules for statistically dependent items.* ICDM 2002, IEEE Computer Society, 2002, 442–449.

[Ter97] Terveen L., Hill W., Amento B., McDonald D., Creter J., *PHOAKS: A system for sharing recommendations.* Communications of the ACM, 40(3), 1997, 59–62.

[Thu79] Thurman B., *In the office: Networks and coalitions.* Social Networks, 2, 1979, 47–63.

[Tom00] Tomlin J., *An Entropy Approach to Unintrusive Targeted Advertising on the Web.* Computer Networks, 33(1–6) Elsevier, 2000, 767–774.

[Tut07] Tutzaue F., *Entropy as a measure of centrality in networks characterized by path-transfer flow.* Social Networks, 29(2), 2007, 249–265.

[Val06] Valverde S., Theraulaz G., Gautrais J., Fourcassie V., Sole R.V., *Self-organization patterns in wasp and open source communities.* IEEE Intelligent Systems, 21(2), 2006, 36–40.

[Wan03] Wan Q., An A., *Efficient Mining of Indirect Associations Using HI-Mine.* Advances in Artificial Intelligence: 16th Conference of the Canadian Society for Computational Studies of Intelligence, AI 2003, Springer Verlag, LNCS 2671, 2003, 206–221.

[Wan06a] Wan Q., An A., *An efficient approach to mining indirect associations.* Journal of Intelligent Information Systems, 27(2), 2006, 135–158.

[Wan06b] Wan Q., An A., *Efficient Indirect Association Discovery Using Compact Transaction Databases.* IEEE International Conference on Granular Computing, GrC'06, IEEE Press, 2006. http://www.cse.yorku.ca/~aan/research/paper/grc06-final.pdf.

[Wang02] Wang D., Bao Y., Yu G., Wang G., *Using Page Classification and Association Rule Mining for Personalized Recommendation in Distance Learning.* ICWL 2002, LNCS 2436, Springer Verlag, 2002, 363–376.

[Wang04] Wang F.-H., Shao H.-M., *Effective personalized recommendation based on time-framed navigation clustering and association mining.* Expert Systems with Applications, 27, 2004, 365–377.

[Was94] Wasserman S., Faust K., *Social network analysis: Methods and applications.* Cambridge University Press, New York, 1994.

[Wei96] Weiss R., Velez B., Sheldon M.A., Namprempr, C., Szilagyi P., Duda A., Gifford D.K., *HyPursuit: A Hierarchical Network Search Engine that Exploits Content-Link Hypertext Clustering.* 7th ACM Conference on Hypertext, Hypertext'96, ACM Press, 1996, 180–193.

[Wel01] Wellman B., *Computer Networks as Social Networks*. Science, 293(5537), 2001, 2031–2035.

[Wel07] Welch E.W., Pandey S., *Multiple Measures of Website Effectiveness and their Association with Service Quality in Health and Human Service Agencies*. HICSS'07, IEEE Computer Society, 2007, p. 107c.

[Wey88] Weyuker E. J., *Evaluating software complexity measures*. IEEE Transactions on Software Engineering, 14(9), 1988, 1357–1365.

[Wu04] Wu X., Zhang C., Zhang S., *Efficient Mining of Both Positive and Negative Association Rules*. ACM Transaction on Information Systems, 22(3), 2004, 381–405.

[Xin04] Xing W., Ghorbani A.A., *Weighted PageRank Algorithm*. 2nd Annual Conference on Communication Networks and Services Research, CNSR 2004, IEEE Computer Society, 2004, 305–314.

[Xio04] Xiong H., S., Tan P.-N., Kuma, V., *Exploiting a support-based upper bound of Pearson's correlation coefficient for efficiently identifying strongly correlated pairs*. KDD 2004, ACM Press, 2004, 334–343.

[Yag00] Yager R.R., *Targeted E-commerce Marketing Using Fuzzy Intelligent Agents*. IEEE Intelligent Systems, 15(6), 2000, 42–45.

[Yan03] Yang H., Parthasarathy S., *On the Use of Constrained Associations for Web Log Mining*. WEBKDD 2002, MiningWeb Data for Discovering Usage Patterns and Profiles, Springer Verlag, LNCS 2703, 2003, 100–118.

[Yan04] Yan P., Chen G., Cornelis C., De Cock M., Kerre E., *Mining Positive and Negative Fuzzy Association Rules*. 8th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, KES 2004, LNAI 3213, Springer, 2004, 270–276.

[Yan06] Yang W.S., Dia J.B., Cheng H.C., Lin H.T., *Mining Social Networks for Targeted Advertising*, 39th Hawaii International Conference on Systems Science, HICSS 2006, IEEE Computer Society, 2006.

[Yao02] 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.

[Yen96] Yen S.J., Chen A.L.P., *An Efficient Approach to Discovering Knowledge from Large Databases*. The Fourth International Conference on Parallel and Distributed Information Systems, PDIS'96, IEEE Computer Society, 1996, 8–18.

[Yih06] Yih W.-T., Goodman J., Carvalho V.R., *Finding advertising keywords on web pages*. WWW 2006, ACM, 2006, 213–222.

[Yua02] Yuan X., Buckles B.P., Yuan Z., Zhang J., *Mining Negative Association Rules*. ISCC'02, IEEE Computer Society, 2002, 623–628.

[Zak97a] Zaki M.J., Parathasarathy S., Li W., *A Localized Algorithm for Parallel Association Mining*. 9th Annual ACM Symposium on Parallel Algorithms and Architectures, SPAA'97, 1997, 321–330.

[Zak97b] Zaki M., Parthasarathy S., Ogihara M., Li W., *New Algorithms for Fast Discovery of Association Rules*. KDD'97, AAAI Press, 1997, 283–296.

[Zak00] Zaki M.J., *Scalable algorithms for association mining*. IEEE Transactions on Knowledge and Data Engineering, 12(3), 2000, 372–390.

[Zak01] Zaki M. J., *SPADE: An efficient algorithm for mining frequent sequences*. Machine Learning 42, 1/2, 2001, 31–60.

[Zha02] Zhang M., Kao B., Cheung D.W., Yip C.L., *Efficient algorithms for incremental update of frequent sequences*. 6th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, PAKDD 2002, LNCS 2336 Springer 2002, 186–197.

[Zho04] Zhong N., Yao Y.Y., Liu C., Ou C., Huang J., *Data Mining for Targeted Marketing*. Chapter 6 in N. Zhong, J. Liu (eds.), *Intelligent Technologies for Information Analysis*, Springer Verlag, 2004, 109–131.

[Zhu06] Zhu W., Chen C., Allen R.B., *Visualizing an enterprise social network from email*. 6th ACM/IEEE-CS joint Conference on Digital Libraries, ACM Press, 2006, 383.

[Zhu07] Zhu T., Bai S., *A Parallel Mining Algorithm for Closed Sequential Patterns*. 21st International Conference on Advanced Information Networking and Applications, AINA 2007, Vol. 1, IEEE Computer Society, 2007, 392–395.

Associations describe relations[illegible] between objects and can be extracted from large datasets using data mining techniques. Some selected kinds of associations like positive and negative association rules, sequential patterns, dynamic associations and associations between humans in social networks are studied in the monograph. Two new patterns are introduced, namely indirect association rules and sequential patterns with negative conclusions.

There are also some application areas discussed: association rules and sequential patterns utilized to recommender systems and hyperlink assessment, dynamic associations exploited in web advertising and social network analysis.

Wydawnictwa Politechniki Wrocławskiej
są do nabycia w księgarni „Tech"
plac Grunwaldzki 13, 50-377 Wrocław
budynek D-1 PWr., tel. 071 320 29 35
Prowadzimy sprzedaż wysyłkową

ISBN 978-83-7493-444-2